Large-scale dynamical model approximation and its applications

Charles Poussot-Vassal

▶ **To cite this version:**

Charles Poussot-Vassal. Large-scale dynamical model approximation and its applications. Automatic. INP DE TOULOUSE, 2019. tel-02188802

HAL Id: tel-02188802

https://theses.hal.science/tel-02188802v1

Submitted on 18 Jul 2019

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

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

THÈSE

En vue de l'obtention de l'

HABILITATION A DIRIGER DES RECHERCHES

Délivrée par : *l'Institut National Polytechnique de Toulouse (INP Toulouse)*

Présentée et soutenue le *JJ/MM/2019* **par :**

CHARLES POUSSOT-VASSAL

Large-scale dynamical model approximation and its applications

JURY

ATHANASIOS ANTOULAS	Professeur, RICE University	Rapporteur
PIERRE APKARIAN	Directeur de Recherche, ONERA Toulouse	Examinateur
GERMAIN GARCIA	Professeur, INSA Toulouse et LAAS CNRS	Examinateur
SARA GRUNDEL	Docteur, Max Planck Institute	Examinateur
MARTINE OLIVI	Chargée de Recherche, INRIA	Examinateur
CHRISTOPHE PRIEUR	Directeur de Recherche, GIPSA-lab CNRS	Rapporteur
KAREN WILLCOX	Professeur, University of Texas	Rapporteur

Abstract

Comprendre est le commencement
d'approuver.

Bachus Spinoza

Linear dynamical models play an important role in many engineering fields, including simulation, analysis, optimization and control of complex systems and processes. This role is emphasized when critical systems are under consideration, and for which a deep attention and understanding are needed. Among others, this attention may be motivated by industrial, economical, societal and strategical reasons. Indeed, for these cases, digital-based solutions involving dedicated computer-based softwares are being developed and largely preferred by engineers and researchers to reduce development costs and time, to improve and to better understand the systems under consideration. These systems being largely grounded on accurate complex and large-scale dynamical models, not well adapted to standard numerical tool and computationally demanding, their approximation by an (accurate) low complexity dynamical model is then a cornerstone for further advanced developments.

This manuscript addresses this last point, namely, the linear large-scale (and infinite dimensional) dynamical model approximation. Moreover, many research and industrial applications are detailed, illustrating the wide application spectrum of this research field. More specifically, the interpolatory framework, tailored to a large variety of dynamical model structures and classes, is the main tool invoked. Moreover, as side effect of the main purpose of dynamical model approximation, extensions to the approximation of the input-output stability regions of a class of meromorphic functions is also presented, highlighting the effectiveness and versatility of the methods developed within this research field.

Manuscript reading guidelines

The present manuscript is composed of seven chapters. It aims at presenting some of the contributions within the linear **large-scale dynamical model approximation and its applications**. Attention is given to present these results in a hopefully didactical manner, with as much as possible **Examples** (illustrations of principles on academic problems) and **Use-cases** (illustrations of principles involving industrial applications, usually treated within projects). More specifically, the following chapters are detailed:

Chapter 1 introduces the linear dynamical systems definitions and notations involved along the manuscript. Moreover, computational aspects related to linear algebra operations, largely used in model approximation, are discussed.

Chapter 2 gives a glimpse of the model reduction problem and state of the art. It allows also introducing some notations. As this subject is largely treated in the literature, author do not claim at being exhaustive. Still, it may provide a compact view and potential first access to beginners.

Chapter 3 details the first contribution of my activities in model approximation: the frequency-limited model approximation (this chapter is mainly inspired from Vuillemin, 2014).

Chapter 4 details the second contribution of my activities in model approximation: the approximation by an input-output delay structured model (this chapter is mainly inspired from Pontes, 2017).

Chapter 5 is attached to present a new result, aiming at estimating the stability of a finite energy meromorphic function, by mean of rational approximation. This chapter presents an early stage result where proofs are still incomplete. However, through numerical examples and some conjectures, author believes that it still might be relevant for future researches.

Chapter 6 summarises the manuscript main results, both from the theoretical part and applicative one. Moreover, a brief description of the **MOR** Toolbox, a software dedicated to large-scale problems, is given.

Chapter 7 finally provides some ideas for further research activities in the forthcoming years, within the continuation of dynamical model approximation and its applications.

Obviously, expert reader in linear systems theory may skip Chapter 1, and those expert in linear model reduction may also skip Chapter 2.

Contents

Charles Poussot-Vassal

Researcher in systems and control theory, ONERA

12/08/1982
@ charles.poussot-vassal@onera.fr
charles.poussot-vassal@mordigitalsystems.fr
http://sites.google.com/site/charlespoussotvassal/

Current activity

since 2019 — Co-funder of **MOR Digital Systems** (Toulouse, France).
▷ *Software solutions for dynamical model approximation, `http://mordigitalsystems.fr/`.*

since May 2009 — Researcher (full time), **ONERA-DTIS** (Toulouse, France).
▷ *Topics: dynamical model theory and approximation, control theory, linear algebra.*
▷ *Main projects: Clean Sky SFWA (2009-2016), Clean Sky 2 Airframe (2017-2024), INTACOO.*
▷ *Referred publications: 11/50/1/7 (journals / conferences / book / chapters).*
▷ *Academic coll.: DLR (Germany), Virginia Tech (USA), Pol. di Milano (Italy), INRIA Nice (France).*
▷ *Industrial coll.: Airbus, Dassault-Aviation, EDF.*
▷ *Supervision: 4 Ph.D. (2 defended), 10 M.Sc.*
▷ *Teaching: lecture and labs in control theory at INSA Toulouse, ISAE and ENAC.*

Professional experiences

2009 (6 months) — Researcher (assistant), **Politecnico di Milano** (Milan, Italy).
▷ *Modeling and control of semi-active suspension systems (book publication, Elsevier).*

2005-2008 (3 years) — Researcher (Ph.D.), **GIPSA-lab/CNRS** Control dpt. (Grenoble, France).
▷ *Study and control of the automotive vehicles dynamics (suspensions, brake, steering wheel, tires).*

2005 (5 months) — Research engineer, **INRIA** (Montbonnot, France).
▷ *Friction compensation on a bipedal robot.*

2004 (7 months) — Research engineer, **ALCATEL Space** (Valence, France).
▷ *Modeling, control and simulation of a brushless motor for braking systems.*

Skills

Languages	**Italian**: bilingual (International Baccalauréat), **English**: frequently used in professional context (TOEIC: 800, ERASMUS exchange), **Spanish**: basic.
Engineering	Dynamical systems approximation and control theory, linear algebra, numerical simulation, digital implementation, signal processing, filtering.
Management	Management of research projects, European project proposal and tracking, technological intelligence, planning, budget.
Computer sciences	Matlab-Simulink, Scilab, LaTeX, Office suite.

Education

Ph.D. (2005-2008) — **Grenoble INP** Institut Polytechnique de Grenoble (Grenoble, France).
▷ *Ph.D. in systems and control theory.*
▷ *Subject: Multivariable robust linear parameter varying control of vehicles (Ministry grant).*

M.Sc. (2005) — **LTH** Lund Institute of Technology (Lund, Sweden).
▷ ***M.Sc. (with honors)** in control theory, embedded systems, numerical analysis.*

Engineer (2000-2005) — **INPG-ESISAR** Institut Polytechnique de Grenoble (Valence, France).
▷ ***Engineer (with honors)** in control theory, electronics and embedded systems.*

Extra activities & scientist

Community	Reviewer for control conferences and journals (IFAC, IEEE CSS, Elsevier), GT MOSAR officer.
Sports	Skiing (competition level), Tennis, Volleyball.
Others	First aid qualification, Driving licence.

Part I

Dynamical model approximation and its applications

Chapter 1

Generalities in linear dynamical models

The truth is out there.

X-files.

Contents

This chapter provides generalities and notations on linear dynamical systems. We start in Section 1.1, with the signals and dynamical systems definitions. Then, in Section 1.2, we provide details about the specific case of finite dimensional models and introduce the state-space form, as well as some associated tools largely used to analyse dynamical systems. Then, as it is an underlying (hidden) theme in the model approximation field, some discussions on linear algebra and computational aspects are given in Section 1.3. The chapter is closed in Section 1.4, opening the path to model approximation and its applications.

1.1 Preliminary in signals, systems and norms

A quick reminder about **LTI** models theory is given here mainly to introduce the basic notions and notations. More specifically, *(i)* the representation of a **LTI** model as a convolution with the impulse response, *(ii)* the $\mathcal{H}_2$, $\mathcal{H}_\infty$, $\mathcal{L}_2$ and $\mathcal{L}_\infty$ spaces and norms, and finally, *(iii)* the inequalities involving systems and signals are recalled. More detailed definitions can be found in many complete monographs such as Zhou and Doyle (1997); Gu et al. (2003); Antoulas (2005); Michiels and Niculescu (2007), which provide a much more precise definition set for linear dynamical systems.

1.1.1 Signals and norms

We define a signal as a Lebesgue measurable function $\mathbf{f}$ that maps the real numbers $\mathbb{R}$ to $\mathbb{C}^n$. The set of signals is defined as

$$\mathcal{S}_n = \{\mathbf{f} : \mathbb{R} \to \mathbb{C}^n \mid \mathbf{f} \text{ measurable}\}.$$

Let us denote $L_2^n(-\infty, \infty)$, a subspace of $\mathcal{S}_n$, as the set of functions with finite energy given as:

$$L_2^n(-\infty, \infty) = \left\{ \mathbf{f} \in \mathcal{S}_n \mid \int_{-\infty}^{\infty} ||\mathbf{f}(t)||_2^2 dt < \infty \right\}.$$

The $L_2^n(-\infty,\infty)$ is a Hilbert space equipped with the inner product and norm respectively defined as:

$$\langle \mathbf{f},\mathbf{g}\rangle_{L_2} = \int_{-\infty}^{\infty} \mathbf{g}^H(t)\mathbf{f}(t)dt \text{ and } \langle \mathbf{f},\mathbf{f}\rangle_{L_2}^{\frac{1}{2}} = \left(\int_{-\infty}^{\infty} ||\mathbf{f}(t)||_2^2 dt\right)^{\frac{1}{2}}.$$

In many engineering applications, the signal L_2-norm is often known as its energy, or the **RMS** for Root Mean Square.

1.1.2 Systems and norms

A dynamical model (or system) is a mathematical equation representing a physical process that evolves in time. In this monograph, we are particularly interested in continuous **LTI** models[1]. Hereafter, we provide some definitions, but for a more general axiomatic description of system / model, reader can refer to the first chapters of the books of Zhou and Doyle (1997) or Antoulas (2005).

Time-domain LTI models representation

A continuous **LTI** model $\mathbf{H}$ is an "input-output" map associating to an input signal $\mathbf{u} \in \mathcal{D}(\mathcal{S}_{n_u})$, a subset of $\mathcal{S}_{n_u}$, an output one $\mathbf{y} \in \mathcal{S}_{n_y}$ by means of the convolution operation, defined as

$$\begin{array}{rccl} \mathbf{H}: & \mathcal{D}(\mathcal{S}_{n_u}) & \mapsto & \mathcal{S}_{n_y} \\ & \mathbf{u}(t) & \mapsto & \mathbf{y}(t) = \displaystyle\int_{-\infty}^{\infty} \mathbf{h}(t-\tau)\mathbf{u}(\tau)d\tau = \mathbf{h}(t) * \mathbf{u}(t), \end{array}$$

where $\mathbf{h}(t)$ is the impulse response of the system. If $n_u > 1$ or $n_y > 1$, the system is said to be **MIMO** and **SISO** if $n_u = n_y = 1$. It is (strictly) causal if and only if $\mathbf{h}(t) = 0$ for $(t \leq 0)\ t < 0$. In these cases, its convolution thus reads:

$$\mathbf{y}(t) = \int_0^{\infty} \mathbf{h}(t-\tau)\mathbf{u}(\tau)d\tau = \int_0^{\infty} \mathbf{h}(\tau)\mathbf{u}(t-\tau)d\tau.$$

In this work, causal models will be mainly considered[2].

Frequency-domain LTI models representation

The unilateral Laplace transform $\mathcal{L}(\cdot)$ of the impulse response $\mathbf{h}$ of a **LTI** model is defined as:

$$\mathbf{H}(s) = \mathcal{L}(\mathbf{h}) = \int_0^{\infty} \mathbf{h}(\tau)e^{-s\tau}d\tau,$$

where $s \in \mathbb{C}$ denotes the Laplace variable. Then, by taking the Laplace transform of the causal convolution product above defined, one obtains

$$\mathbf{y}(s) = \mathbf{H}(s)\mathbf{u}(s),$$

where $\mathbf{u}(s)$ and $\mathbf{y}(s)$ are the Laplace transform of $\mathbf{u}(t)$ and $\mathbf{y}(t)$. The $n_y \times n_u$ complex-valued matrix function $\mathbf{H}(s)$ is the transfer function of the **LTI** model. A model is said to be real (as treated in this work) if its impulse response matrix $\mathbf{h}(t)$ is a real-valued matrix function. As a consequence, the complex-valued transfer function $\mathbf{H}(s)$ satisfies $\overline{\mathbf{H}(s)} = \mathbf{H}(\overline{s})$, for all $s \in \mathbb{C}$. Moreover, a transfer function is said to be proper if $\mathbf{H}(\infty) < \infty$ and strictly proper if $\mathbf{H}(\infty) = 0$. An **LTI** system $\mathbf{H}$ is said to be stable if and only if its transfer function is bounded and analytic on $\mathbb{C}_+$, *i.e.* it has no singularities on the closed right half-plane. Conversely, it is said to be anti-stable if and only if its transfer function is bounded and analytic on $\mathbb{C}_-$. The family of stable models can be regarded as a functional space of analytic meromorphic functions[3] on the right half-plane and is therefore a Hardy space. These specific spaces are slightly more detailed in what follows (see also Hoffman (1962) or Chapter 2 of Pontes (2017) for more details).

[1] Parametric **LTI** models, or **p-LTI** models, will also be addressed in this manuscript through specific examples and use-cases, and thus more specifically presented there.

[2] Some non-causal models can be considered, but mainly for some theoretical illustrations rather than practical applications.

[3] A function is a holomorphic function over domain $D\backslash A$ where A is a denumerable set of isolated points, being the poles.

Remark 1.1 (Meromorphic functions) *The complex-valued transfer function* $\mathbf{H}$ *is a meromorphic function. Given two meromorphic functions* $\mathbf{H}_1$ *and* $\mathbf{H}_2$*, their sum and product remains meromorphic.*

1.1.3 The (complex domain) $\mathcal{H}_p$ Hardy spaces and norms

Hardy spaces are generally referred to as spaces of functions of a complex variable $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$, analytic over a given region (*i.e.* complex differentiable at every point of this region) and on which a measure $||\mathbf{H}||_{\mathcal{H}_p}$ is finite (*i.e.* $||\mathbf{H}||_{\mathcal{H}_p} < \infty$). As the complex variable can be interpreted as a frequency one in many engineering fields (*e.g.* mechanical, electrical, ...), they are considered as frequency-domain spaces. In control theory, these spaces are of particular interest for many reasons: *(i)* stability properties of **LTI** dynamical systems are related to the fact that the transfer function belongs to some Hardy spaces (*e.g.* for continuous time systems, the Hardy space is defined in the open right half complex plane), *(ii)* performance characteristics of **LTI** systems are defined through Hardy space norms and *(iii)* the distance between two **LTI** models $\mathbf{H}_1$ and $\mathbf{H}_2$ can be measured by considering the norm of the difference of their transfer function, *i.e.* $||\mathbf{H}_1 - \mathbf{H}_2||_{\mathcal{H}_p}$. The latter item is of particular interest in the model approximation field since it provides a way to quantify the error between two dynamical models, which can then turn to be an optimisation criteria (see Chapter 2). After providing general definitions of Hardy spaces, two specific control-oriented ones are recalled (see also Hoffman (1962); Partington (1997, 2004); Antoulas (2005)).

Definition 1.1: $\mathcal{H}_p$ spaces

Let $\mathbb{C}_+ = \{x, y \in \mathbb{R} : x + \imath y \in \mathbb{C}, x > 0\}$, $\mathbb{C}_- = \{x, y \in \mathbb{R} : x + \imath y \in \mathbb{C}, x < 0\}$ and $\partial\mathbb{C} = \{x, y \in \mathbb{R} : x + \imath y \in \mathbb{C}, x = 0\}$. The $\mathcal{H}_p$ spaces are defined as

$$\mathcal{H}_p^{n_y \times n_u} \triangleq \mathcal{H}_p^{n_y \times n_u}(\mathbb{C}_+) = \left\{\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u} \mid ||\mathbf{H}||_{\mathcal{H}_p} < \infty\right\},$$

where $\mathbf{H}$ denotes the $n_y \times n_u$ complex-valued functions analytic in $\mathbb{C}_+$ and for which the p-norm is defined as

$$\begin{aligned} ||\mathbf{H}||_{\mathcal{H}_p} &\triangleq \sup_{x>0} \left(\int_{-\infty}^{+\infty} ||\mathbf{H}(x + \imath y)||_{S,p}^p dy \right)^{\frac{1}{p}} && \text{for } p \in [1\ \infty) \\ &\triangleq \sup_{x>0} ||\mathbf{H}(x + \imath y)||_{S,p} && \text{for } p = \infty, \end{aligned} \tag{1.1}$$

where $||\mathbf{H}(s_0)||_{S,p}$ is the Schatten p-norm of $\mathbf{H}$ evaluated at $s = s_0$. The $\mathcal{H}_p^{n_y \times n_u}(\mathbb{C}_-)$ is similarly defined.

Before presenting special cases on Hardy spaces, let us remind the following maximum modulus principle, which provides a way to simplify the search space of the norm formula (1.1).

Theorem 1.1: Maximum modulus principle

Given $\mathbb{X} \subseteq \mathbb{C}$ be a bounded domain, and let $\mathbf{H} : \mathbb{X} \to \mathbb{C}^{n_y \times n_u}$ be a function continuous on the closed set $\overline{\mathbb{X}}$ and analytic (holomorphic) on $\mathbb{X}$. Then, the maximum value of $\mathbf{H}$ on $\overline{\mathbb{X}}$ (which always exists) occurs on the boundary $\partial\mathbb{X} : \overline{\mathbb{X}} \backslash \mathbb{X}$. In other words,

$$\max_{\overline{\mathbb{X}}} ||\mathbf{H}|| = \max_{\partial\mathbb{X}} ||\mathbf{H}||.$$

Then, the search of the suprema in the above formulas (1.1), can be simplified by making use of the Maximum modulus principle, stated in Theorem 1.1, which states that a complex function $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ continuous inside a domain $\mathbb{X} \in \mathbb{C}$ and on its boundary $\partial\mathbb{X}$ (*e.g.* $\partial\mathbb{C}$), and analytic inside $\mathbb{X}$

(*e.g.* $\mathbb{C}_+$), attains its maximum on the boundary $\partial\mathbb{X}$ (*e.g.* $\partial\mathbb{C}$) of $\mathbb{X}$. Thus, (1.1) becomes

$$\begin{array}{rcll} ||\mathbf{H}||_{\mathcal{H}_p} & = & \left(\int_{-\infty}^{+\infty} ||\mathbf{H}(\imath y)||_{S,p}^p dy\right)^{\frac{1}{p}} & \text{for } p \in [1\ \infty) \\ & = & \sup\limits_{y\in\mathbb{R}} ||\mathbf{H}(\imath y)||_{S,p} & \text{for } p = \infty. \end{array} \tag{1.2}$$

Following (1.2), two special cases can be derived[4]:

$$||\mathbf{H}||_{\mathcal{H}_2} = \left(\int_{-\infty}^{+\infty} ||\mathbf{H}(\imath y)||_{S,2}^2 dy\right)^{\frac{1}{2}} = \left(\int_{-\infty}^{+\infty} ||\mathbf{H}(\imath y)||_F^2 dy\right)^{\frac{1}{2}} = \left(\int_{-\infty}^{+\infty} \mathbf{tr}\big(\overline{\mathbf{H}(\imath y)}\mathbf{H}^T(\imath y)\big) dy\right)^{\frac{1}{2}},$$

where $||\mathbf{H}(s)||_F^2 = \mathbf{tr}\big(\mathbf{H}(-s)\mathbf{H}^T(s)\big)$ denotes the Frobenius norm of a complex matrix $\mathbf{H}(s)$, and

$$||\mathbf{H}||_{\mathcal{H}_\infty} = \sup_{y\in\mathbb{R}} ||\mathbf{H}(\imath y)||_{S,\infty} = \sup_{y\in\mathbb{R}} ||\mathbf{H}(\imath y)||_2 = \sup_{y\in\mathbb{R}} \sigma_{max}\big(\mathbf{H}(\imath y)\big),$$

where $\sigma_{max}(\cdot)$ denotes the maximum singular value operator of a complex matrix.

Note that the $\mathcal{H}_2$-norm, as treated in the control literature is often weighted by the scaling factor $1/2\pi$ in order to link the original Hardy $\mathcal{H}_2$-norm with the impulse response of a dynamical transfer function, through the Parseval's Theorem. Now $\mathcal{H}_p$ spaces have been recalled, let us move to the $\mathcal{L}_p$ ones which will have an importance in the stability definition and characterisation in the rest of the monograph.

1.1.4 The $\mathcal{L}_p$ spaces

If $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ has no singularities on the imaginary axis but is not necessarily analytic neither on the right nor left half of the complex plane, $\mathcal{H}_p$-norms are not defined any more. Instead, the frequency domain $\mathcal{L}_p$ spaces are used.

Definition 1.2: $\mathcal{L}_p$ spaces

Let $\imath\mathbb{R} = \{x, y \in \mathbb{R} : x + \imath y \in \mathbb{C}, x = 0\}$. The $\mathcal{L}_p$ spaces are defined as

$$\mathcal{L}_p^{n_y\times n_u} \triangleq \mathcal{L}_p^{n_y\times n_u}(\imath\mathbb{R}) = \big\{\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y\times n_u} \mid ||\mathbf{H}||_{\mathcal{L}_p} < \infty\big\},$$

where $\mathbf{H}$ denotes the $n_y \times n_u$ complex-valued functions which has no singularities on the imaginary axis but is not necessarily analytic either on the right or left half of the complex plane. The frequency domain $\mathcal{L}_p$-norms of $\mathbf{H}$ are defined as follows:

$$\begin{array}{rcll} ||\mathbf{H}||_{\mathcal{L}_p} & \triangleq & \sup\limits_{x\neq 0}\left(\int_{-\infty}^{+\infty} ||\mathbf{H}(\imath y)||_{S,p}^p dy\right)^{\frac{1}{p}} & \text{for } p \in [1\ \infty) \\ & \triangleq & \sup\limits_{x\neq 0} ||\mathbf{H}(\imath y)||_{S,p} & \text{for } p = \infty, \end{array}$$

where $||\mathbf{H}(s_0)||_{S,p}$ is the Schatten p-norm of $\mathbf{H}$ evaluated at $s = s_0$.

As for the Hardy spaces, following Definition 1.2 and Theorem 1.1, two special norms can be derived:

$$||\mathbf{H}||_{\mathcal{L}_2} = \left(\int_{-\infty}^{+\infty} ||\mathbf{H}(\imath y)||_{S,2}^2 dy\right)^{\frac{1}{2}} = \left(\int_{-\infty}^{+\infty} ||\mathbf{H}(\imath y)||_F^2 dy\right)^{\frac{1}{2}} = \left(\int_{-\infty}^{+\infty} \mathbf{tr}\big(\overline{\mathbf{H}(\imath y)}\mathbf{H}^T(\imath y)\big) dy\right)^{\frac{1}{2}},$$

and

$$||\mathbf{H}||_{\mathcal{L}_\infty} = \sup_{y\in\mathbb{R}} ||\mathbf{H}(\imath y)||_{S,\infty} = \sup_{y\in\mathbb{R}} ||\mathbf{H}(\imath y)||_2 = \sup_{y\in\mathbb{R}} \sigma_{max}\big(\mathbf{H}(\imath y)\big).$$

Note that $\mathcal{H}_p$ may be viewed as a subspace of $\mathcal{L}_p$. Indeed, the limit ($x \to 0$) of a $\mathcal{H}_p$ function is in $\mathcal{L}_p$. Moreover, $\mathcal{H}_2$ is the image of $\mathcal{L}_2([0, \infty))$ through the Laplace transform, which implies causality and $L_2 - L_\infty$ stability.

[4] And by noticing that the Schatten 2-norm and ∞-norm are the Frobenius and Spectral norms, respectively.

1.1.5 The (complex domain) Hardy spaces and norms in control theory

Given the above general formulations of the $\mathcal{L}_p$ and $\mathcal{H}_p$ spaces, let us now define more formally the $\mathcal{L}_2$, $\mathcal{H}_2$, $\mathcal{L}_\infty$ and $\mathcal{H}_\infty$ spaces which play a central role in many control engineering problems, and especially in the linear model approximation problem.

Definition 1.3: $\mathcal{L}_2(\imath\mathbb{R})$ space

The $\mathcal{L}_2(\imath\mathbb{R})$ (or shortly $\mathcal{L}_2$) space is the vector-space of matrix-valued functions $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ defined on the imaginary axis and which satisfies

$$\int_{\mathbb{R}} \mathbf{tr}\left(\overline{\mathbf{H}(\imath\omega)}\mathbf{H}(\imath\omega)^T\right) d\omega = \int_{\mathbb{R}} ||\mathbf{H}(\imath\omega)||_F^2 d\omega < \infty.$$

This space is a Hilbert space equipped with the inner product defined as

$$\langle \mathbf{H}, \mathbf{G} \rangle_{\mathcal{L}_2} \triangleq \frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathbf{tr}\left(\overline{\mathbf{H}(\imath\omega)}\mathbf{G}(\imath\omega)^T\right) d\omega = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \langle \mathbf{H}(\imath\omega), \mathbf{G}(\imath\omega) \rangle_F d\omega,$$

with corresponding norm $||\mathbf{H}||_{\mathcal{L}_2} \triangleq \langle \mathbf{H}, \mathbf{H} \rangle_{\mathcal{L}_2}^{\frac{1}{2}}$.

Definition 1.4: $\mathcal{H}_2(\mathbb{C}_+)$ space

The $\mathcal{H}_2(\mathbb{C}_+)$ (or shortly $\mathcal{H}_2$) space is the vector-space of matrix-valued functions $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ analytic in the open right half plane $\mathbb{C}_+$ and which satisfies

$$\int_{\mathbb{R}} \mathbf{tr}\left(\overline{\mathbf{H}(\imath\omega)}\mathbf{H}(\imath\omega)^T\right) d\omega = \int_{\mathbb{R}} ||\mathbf{H}(\imath\omega)||_F^2 d\omega < \infty.$$

This space is an Hardy space equipped with the inner product defined as

$$\langle \mathbf{H}, \mathbf{G} \rangle_{\mathcal{H}_2} \triangleq \frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathbf{tr}\left(\overline{\mathbf{H}(\imath\omega)}\mathbf{G}(\imath\omega)^T\right) d\omega = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \langle \mathbf{H}(\imath\omega), \mathbf{G}(\imath\omega) \rangle_F d\omega,$$

with corresponding norm $||\mathbf{H}||_{\mathcal{H}_2} \triangleq \langle \mathbf{H}, \mathbf{H} \rangle_{\mathcal{H}_2}^{\frac{1}{2}}$.

The $\mathcal{H}_2(\mathbb{C}_+)$ space is the vector-space of transfer functions whose impulse responses $\mathbf{h}(t)$ are stable and have finite energy. By analogously denoting $\mathcal{H}_2(\mathbb{C}_-)$, the space where $\mathbf{H} \in \mathcal{H}_2(\mathbb{C}_-)$ if and only if $\mathbf{H}(-s) \in \mathcal{H}_2(\mathbb{C}_+)$). For rational functions, one can say that this space is the function-space of the **LTI** models $\mathbf{H}$ whose all the singularities are in $\mathbb{C}_+$, *i.e.* are all unstable. Therefore, the $\mathcal{H}_2(\mathbb{C}_-)$ space corresponds to the space of the anti-stable models. The $\mathcal{L}_2(\imath\mathbb{R})$ space is the function-space of models whose transfer functions are square-integrable over the imaginary axis. In addition, it is the Laplace transform image of the signal space $L_2^n(-\infty, \infty)$, *i.e.* $\mathcal{L}\big(L_2^n(-\infty, \infty)\big) = \mathcal{L}_2(\imath\mathbb{R})$. Finally, it is interesting to note the following direct relation:

$$\mathcal{L}_2(\imath\mathbb{R}) = \mathcal{H}_2(\mathbb{C}_+) \oplus \mathcal{H}_2(\mathbb{C}_-).$$

Consequently, given $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})$ and its decomposition into stable $\mathbf{H}_s \in \mathcal{H}_2(\mathbb{C}_+)$ and anti-stable $\mathbf{H}_a \in \mathcal{H}_2(\mathbb{C}_-)$ transfers such that $\mathbf{H} = \mathbf{H}_s + \mathbf{H}_a$, the following holds true:

$$\langle \mathbf{H}_s, \mathbf{H}_a \rangle_{\mathcal{L}_2(\imath\mathbb{R})} = 0.$$

Now, let us move to the $\mathcal{L}_\infty$ and $\mathcal{H}_\infty$ spaces and its well known $\mathcal{H}_\infty$-norm, largely used in the context of robust control (see *e.g.* Francis and Doyle (1987); Doyle et al. (1989); Gahinet and Apkarian (1994); Chilali and Gahinet (1996); Scherer et al. (1997); Zhou and Doyle (1997); Apkarian and Noll (2006)).

Definition 1.5: $\mathcal{L}_\infty(\mathbb{C}_+)$ space

The $\mathcal{L}_\infty(\mathbb{C}_+)$ (or shortly $\mathcal{L}_\infty$) space is the vector-space of matrix-valued functions $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ defined over $\mathbb{C}_+$ and which satisfies

$$\sup_{\omega} ||\mathbf{H}(\imath\omega)||_2 < \infty.$$

This space is a Hilbert space equipped with the induced-norm

$$||\mathbf{H}||_{\mathcal{L}_\infty} = \max_{\omega \in \mathbb{R}} \sigma\left(\mathbf{H}(\imath\omega)\right).$$

Definition 1.6: $\mathcal{H}_\infty(\mathbb{C}_+)$ space

The $\mathcal{H}_\infty(\mathbb{C}_+)$ (or shortly $\mathcal{H}_\infty$) space is the vector-space of matrix-valued functions $\mathbf{H} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ analytic over the right half plane $\mathbb{C}_+$ and which satisfies

$$\sup_{\Re(s)>0} ||\mathbf{H}(s)||_2 < \infty.$$

This space is an Hardy space equipped with the induced-norm

$$||\mathbf{H}||_{\mathcal{H}_\infty} = \max_{\omega \in \mathbb{R}} \sigma\left(\mathbf{H}(\imath\omega)\right). \tag{1.3}$$

Based on the above definitions, one may note that both $\mathcal{L}_\infty$ and $\mathcal{H}_\infty$-norms are similarly defined, although the $\mathcal{H}_\infty$ one assumes that the transfer function $\mathbf{H}$ is bounded and analytic over $\mathbb{C}_+$, meaning that the model does not have any singularity on the right hand side. It is to be noticed that a slight abuse of language is usually done in the control community since one usually computes the $\mathcal{H}_\infty$-norm of an unstable systems.

Based on the definitions of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$-norms, some inequalities hold. Indeed, system norms can be used to measure how "large" an output $\mathbf{y}(t)$ will be if an **LTI** model is subjected to an input $\mathbf{u}(t)$. In others words, they can be used to estimate of an upper bound of the output $\mathbf{y}(t)$, given an input $\mathbf{u}(t)$. Two of these bounds that are useful to motivate the model reduction problem.

- Let $\mathbf{H} \in \mathcal{H}_2$ and $\mathbf{u} \in L_2$, then

$$||\mathbf{y}||_{L_\infty} = \sup_{t \geq 0} ||\mathbf{y}(t)||_2 \leq ||\mathbf{H}||_{\mathcal{H}_2} ||\mathbf{u}||_{L_2}. \tag{1.4}$$

  Thus, a model $\mathbf{H}$ possessing a "small" $\mathcal{H}_2$-norm will produce output signals whose peak amplitude is also "small". The $\mathcal{H}_2$-norm is also known as the L_2-L_∞ norm.

- Let $\mathbf{H} \in \mathcal{H}_\infty$ and $\mathbf{u} \in L_2$, then

$$||\mathbf{y}||_{L_2} \leq ||\mathbf{H}||_{\mathcal{H}_\infty} ||\mathbf{u}||_{L_2}. \tag{1.5}$$

  Hence, a model $\mathbf{H}$ possessing "small" $\mathcal{H}_\infty$ norm will produce output signals with "small" energy. The $\mathcal{H}_\infty$-norm is also known as the L_2-L_2 (induced) norm.

Remark 1.2 ($\mathcal{H}_2$ and $\mathcal{H}_\infty$ input-output stability) *A system is considered to be stable if small inputs lead to responses that do not diverge. Different input-output norms can lead to different notions of stability. In this work we considered the following two notions:*

- *$\mathcal{H}_2$ stability: a system is $\mathcal{H}_2$ stable if its transfer function $\mathbf{H}$ lies in $\mathcal{H}_2$. In this case the inequality (1.4) holds, and L_2 bounded inputs produces L_∞ bounded outputs.*

- *$\mathcal{H}_\infty$ stability: a system is $\mathcal{H}_\infty$ stable if its transfer function $\mathbf{H}$ lies in $\mathcal{H}_\infty$. In this case, the inequality (1.5) holds, and L_2 bounded inputs produces L_2 bounded outputs*

Remark 1.3 ($\mathcal{H}_2$ and $\mathcal{H}_\infty$ physical interpretation) *From a physical view point, for **SISO** systems, the $\mathcal{H}_2$-norm represents the integral of the frequency response absolute value. In the **MIMO** case, the $\mathcal{H}_2$ norm is the impulse-to-energy gain of $\mathbf{y}(t)$ in response to a white noise input $\mathbf{u}(t)$ with uniform spectral density. Similarly, the $\mathcal{H}_\infty$-norm represents the maximal gain of the frequency response of the system. It is also called the worst case attenuation level in the sense that it measures the maximum amplification that the system can deliver on the whole frequency set. For **SISO** (*resp. ***MIMO*** *) systems, it represents the maximal peak value on the frequency response magnitude (*resp. *singular value) plot of $\mathbf{H}(\imath\omega)$, in other words, it is the largest gain if the system is fed by harmonic input signal. The $\mathcal{H}_\infty$-norm is one of the main ingredients in robust control, but more rarely in model approximation due to its computational complexity (see later in this chapter). Indeed, in finite dimension, an iterative bisection algorithm it usually needed, clearly inappropriate in the large-scale settings (see Boyd et al. (1988); Bruinsma and Steinbuch (1990)).*

Remark 1.4 (About the rational $\mathcal{RL}_p$ and $\mathcal{RH}_p$ spaces) *So far, all the $\mathcal{L}_p$ and $\mathcal{H}_p$ spaces defined did not mention any structure of the complex functions describing the transfer functions, but simply the fact that they are bounded on a part of the complex plane. In control theory, as in many engineering fields, most of the transfer functions are rational functions,* i.e. *a numerator and denominator which are polynomial with real coefficients. In this case,* e.g. *instead of $\mathcal{H}_2$, one talks about $\mathcal{RH}_2$. The real rational subspace of $\mathcal{H}_2$, which consists of all strictly proper and real rational stable transfer matrices, is denoted by $\mathcal{RH}_2$. Similarly, the real rational subspace of $\mathcal{H}_\infty$, which consists of all proper and real rational stable transfer matrices, is denoted by $\mathcal{RH}_\infty$ (same comment hold for $\mathcal{L}_2$ and $\mathcal{L}_\infty$). These latter are of particular interest since they are directly connected to model realisations, as clarified in the following Section 1.2.*

Example 1 - Some functions and spaces

Let us consider the following transfer function which all singularities lie in the left half plane, but which is unbounded

$$\frac{1}{1+s+se^{-s}} \notin \mathcal{L}_2 \text{ and } \in \mathcal{L}_\infty$$

More classically, the following rational and irrational functions can be considered:

$$\begin{aligned}
\frac{s+1}{(s+10)(s+6)} &\in \mathcal{RH}_2 \\
\frac{s+1}{(s-10)(s+6)} &\notin \mathcal{RH}_2 \text{ and } \mathcal{RH}_\infty \text{ but } \in \mathcal{RL}_2 \text{ and } \mathcal{RL}_\infty \\
\frac{s+1}{(s+10)} &\notin \mathcal{RH}_2 \text{ and } \in \mathcal{RH}_\infty \\
\frac{1}{1+s+e^{-s}} &\in \mathcal{H}_2
\end{aligned}$$

Without lack of generalities, the $\mathcal{R}$, denoting the rational set, term will be often omitted for simplicity and we will only denote $\mathbf{H}$ as a function belonging in $\mathcal{H}_2$, $\mathcal{L}_2$, *etc.*

1.2 Finite dimensional linear time invariant systems

Now some general notations and properties of dynamical systems and their norms have been briefly recalled, let us move to the more specific case of rational finite dimensional models, accompanied with a realisation[5]. Attention is first given on models realisation, then stable realisation, and finally the poles residues representation, standing as an important tool within model approximation. Finally, the notion of gramian will be introduced as a standard tool in linear models analysis.

1.2.1 LTI models realisation

In the case where a **MIMO LTI** continuous-time dynamical system $\mathbf{H}$ can be represented by a first order descriptor realisation $\mathcal{S} : (E, A, B, C, D)$ with n_u inputs, n_y outputs and n internal variables, the model is given by a set of differential and algebraic equations (**DAE**):

$$\mathcal{S} : \left\{ \begin{array}{rcl} E\dot{\mathbf{x}}(t) & = & A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) & = & C\mathbf{x}(t) + D\mathbf{u}(t) \end{array} \right. \text{ or } \mathcal{S} : \left[\begin{array}{c|c} E, A & B \\ \hline C & D \end{array} \right] \in \mathbb{R}^{(n+n_y)\times(n+n_u)}, \tag{1.6}$$

where, $\mathbf{x}(t) \in \mathbb{R}^n$ denotes the internal variables (the state variables if E is invertible), and $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ and $\mathbf{y}(t) \in \mathbb{R}^{n_y}$ are the input, output functions, respectively, while

$$E, A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times n_u}, C \in \mathbb{R}^{n_y\times n} \text{ and } D \in \mathbb{R}^{n_y\times n_u},$$

are constant matrices. Then, by denoting $\mathbb{S}_{n,n_y,n_u} \triangleq (\mathbb{R}^{n\times n}, \mathbb{R}^{n\times n}, \mathbb{R}^{n\times n_u}, \mathbb{R}^{n_y\times n}, \mathbb{R}^{n_y\times n_u})$, the set of all $n_y \times n_u$ realisations of dimension n, if the matrix pencil (E, A) is regular, *i.e.* is non-singular for some finite $\lambda \in \mathbb{C}$,

$$\begin{array}{rccl} \mathbf{H}(\mathcal{S}) : & \rho(E, A) & \mapsto & \mathbb{C}^{n_y\times n_u} \\ & s & \mapsto & C(sE - A)^{-1}B + D, \end{array} \tag{1.7}$$

is called the transfer function associated to $\mathcal{S} : (E, A, B, C, D) \in \mathbb{S}_{n,n_y,n_u}$ and $\rho(E, A)$ is the resolvant of the matrix pencil (E, A), as defined in Definition 1.7. Then, the set of all realisations of $\mathbf{H}(\mathcal{S})$ is denoted $\mathbb{S}(\mathbf{H}(\mathcal{S}))$ and the evaluation of the transfer function obtained from the realisation $\mathcal{S}$ is $\mathbf{H}(\mathcal{S})(s)$, but for brevity, simply $\mathbf{H}(s)$. Moreover, for $n, n_u, n_y \in \mathbb{N}$, one can define the following sets:

$$\begin{array}{rcl} \mathbb{S}^0_{n,n_y,n_u} & \triangleq & \{(E, A, B, C, D) \in \mathbb{S}_{n,n_y,n_u} \mid \imath\mathbb{R} \subset \rho(E, A)\}, \\ \mathbb{S}^+_{n,n_y,n_u} & \triangleq & \{(E, A, B, C, D) \in \mathbb{S}_{n,n_y,n_u} \mid \mathbb{C}_{\geq 0} \subset \rho(E, A)\}, \\ \mathbb{S}^\alpha_{n,n_y,n_u} & \triangleq & \{(E, A, B, C, D) \in \mathbb{S}_{n,n_y,n_u} \mid \mathbb{C}_{\geq \alpha} \subset \rho(E, A)\ \forall\ \alpha > 0\}, \\ \mathbb{S}^-_{n,n_y,n_u} & \triangleq & \{(E, A, B, C, D) \in \mathbb{S}_{n,n_y,n_u} \mid \mathbb{C}_{\leq 0} \subset \rho(E, A),\ E \text{ regular}\}, \end{array} \tag{1.8}$$

where $\mathbb{S}^+_{n,n_y,n_u}$, $\mathbb{S}^-_{n,n_y,n_u}$ and $\mathbb{S}^\alpha_{n,n_y,n_u}$ are sets of stable, anti-stable and α-stable systems, respectively. One may also note that these spaces are linked to the Hardy ones presented before. Indeed, $\mathbb{S}^0_{n,n_y,n_u}$ is the realisation version of $\mathcal{RL}_\infty(\imath\mathbb{R})$, $\mathbb{S}^+_{n,n_y,n_u}$ of $\mathcal{RH}_\infty(\mathbb{C}_+)$ and $\mathbb{S}^-_{n,n_y,n_u}$ of $\mathcal{RH}_\infty(\mathbb{C}_-)$. By realisation version, one intends the finite order version of these spaces. Therefore, they are subspaces of the Hardy spaces presented before.

[5]The term "finite order" realisation of **TDS** will also be treated in this work, but later and as an application of infinite dimensional model approximation rather than a system theoretical viewpoint. For this class of systems, reader is invited to refer to Briat (2015) and Michiels and Niculescu (2014) books providing a complete insight of the theoretical background.

Definition 1.7: (E, A) matrix pencil, eigenvalues and resolvent

We call $(E, A) \in (\mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n})$ a matrix pair or pencil. A scalar $\lambda \in \mathbb{C}$ is called a (generalised) eigenvalue of (E, A) if $\det(\lambda E - A) = 0$. If there exist $\mathbf{x} \neq 0$ such that $E\mathbf{x} = 0$ and $A\mathbf{y} \neq 0$, then ∞ is called a (generalised) eigenvalue of (E, A), otherwise every $\lambda \in \mathbb{C}$ is an eigenvalue. We denote the set of all eigenvalues of (E, A) by $\Lambda(E, A) \subseteq \mathbb{C} \cup \{\infty\}$ and the resolvent set by $\rho(E, A) = \mathbb{C}\backslash\Lambda(E, A)$. If $\rho(E, A) = \emptyset$, the (E, A) is called singular, otherwise regular. If the matrix pair is regular, then the set of eigenvalues is finite. In this case, the generalised right and left eigenvalues problem consists in finding, for $\{\lambda_i\}_{i=1}^n \in \mathbb{C}$ and non-trivial $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{C}^n$, such as

$$A\mathbf{x}_i = \lambda_i E\mathbf{x}_i \text{ and } \mathbf{y}_i^H A = \lambda_i \mathbf{y}_i^H E.$$

All together, $\{\lambda_i, \mathbf{x}_i, \mathbf{y}_i\}_{i=1}^n$ is the eigen-triplet of the pair (E, A). We call (E, A) regular, if there are n linearly independent right and left eigenvectors $\mathbf{x}_i$ and $\mathbf{y}_i$, then the pair (E, A) is called diagonalisable or non defective and the following holds true, for $i, j = 1, \ldots, n$:

$$\mathbf{y}_i^H A\mathbf{x}_i = \lambda_i \text{ and } \mathbf{y}_i^H E\mathbf{x}_j = \delta_{ij}.$$

The singularities of $\mathbf{H}$ are the poles of the realisation $\mathcal{S}$ and the eigenvalues of the matrix pencil (E, A). $\mathcal{S}$ is stable if all its finite poles are in the left-half of the complex plane (in this case, $\mathbf{H}(s) \in \mathcal{RH}_\infty$ and $\mathcal{S} \in \mathbb{S}^+_{n,n_y,n_u}$). Moreover, it is proper if its value at infinity is finite (and D is constant) and strictly proper if that value is zero (in this case, $D = 0$, $\mathbf{H}(s) \in \mathcal{RH}_2$ and $\mathcal{S} \in \mathbb{S}^\epsilon_{n,n_y,n_u}$, where $\epsilon > 0$). Reader can refer to G.W.Stewart (1972), Kurschner (2010) and Kohler (2014) for comprehensive details on eigenvalues.

The quintuple (E, A, B, C, D) is called a descriptor realisation of $\mathbf{H}$. Importantly, realisations are not unique with respect to any projector[6] $V \in \mathbb{R}^{n\times n}$, and those with the smallest possible dimension n are called minimal realisations. Furthermore $\mathbf{rank}(E)$ is the McMillan degree of $\mathcal{S}$. A realisation is minimal if it is completely controllable and observable. A descriptor system with (E, A) regular is completely controllable if $\mathbf{rank}(\lambda E - A, B) = n$, for all finite $\lambda \in \mathbb{C}$, and $\mathbf{rank}(E, B) = n$. If $E = I$, $\mathcal{S}$ is a standard model, and descriptor otherwise.

Remark 1.5 (About dimensionality, $\mathbf{dim}(\mathbf{H})$ vs. $\mathbf{dim}(\mathcal{S})$) *With the above formulation and with a slight abuse of language, let us denote as $\mathbf{dim}(\mathbf{H}) = n$, the dimension of the model $\mathbf{H}$ corresponding to its number of singularities. When rational meromorphic functions are considered (belonging to $\mathcal{RL}_2$ or $\mathcal{RL}_\infty$), a minimal realisation can be obtained, the dimension n is finite and is then the dimension of the associated state-vector $\mathbf{x}(t)$. When non-rational meromorphic functions are considered* (e.g. *functions belonging to $\mathcal{L}_2$ or $\mathcal{L}_\infty$), the dimension value n is the number of roots or singularities of the function. We denote by $\mathbf{dim}(\mathcal{S}) = n$ the dimension of the state-space vector of the model equipped with a realisation, even if it describes an infinite dimensional model. As an illustration, given a **TDS** $\mathbf{H}$ and realisation $\mathcal{S}$, with delays affecting the state vector $\mathbf{x}(t)$, one have $\mathbf{dim}(\mathbf{H}) = \infty$ because the transfer has an infinite number of eigenvalues and $\mathbf{dim}(\mathcal{S}) = n$ finite, since the realisation can be finite (see* e.g. *Michiels and Niculescu (2014)).*

The above linear system model relations (1.6)-(1.7) are the most widely used in this manuscript and, to the author's experience, clearly are the most largely used in both engineering and research applications (from automotive to aerospace, through hydroelectric *etc.*). The main reason is that they offer a compliant framework for simulation, analysis, optimisation *etc.* From a practitioner viewpoint, $\mathcal{S}$ generally results from linearisation around some equilibrium point of the nonlinear dynamical and algebraic equations governing any system, which can be given as,

$$\left\{\begin{array}{rcl} \dot{\mathbf{x}}(t) & = & \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \\ 0 & = & \mathbf{g}(\mathbf{x}, \mathbf{u}, t) \\ \mathbf{y}(t) & = & \mathbf{h}(\mathbf{x}, \mathbf{u}, t) \end{array}\right.$$

[6] $V \in \mathbb{R}^{n\times n}$ is said to be a projector if $V^TV = I_n$ and $V^2 = V$.

where $\mathbf{f}$, $\mathbf{g}$ and $\mathbf{h}$ refer to the ordinary differential, algebraic and output equations, respectively. Obviously, many other **LTI** representations can be considered, such as the bilinear (Breiten and Damm (2010); Benner and Breiten (2012)), the bilinear stochastic (Benner and Damm (2011)), the quadratic (Kurschner (2010); Benner and Breiten (2015)), the delayed (Briat (2015)), multi-agent systems Jongsma et al. (2018) *etc.* Since these descriptions are not in the scope of the core work performed in my research activities, they will not be addressed in this manuscript. Interested reader may refer to the previous references for more details.

Remark 1.6 (About linear and nonlinear models) *Note that in many applications,* $\mathbf{f}$, $\mathbf{g}$ *and* $\mathbf{h}$ *are not always "known" functions and realisation can be constructed from discretisation schemes, finite element approximation,* etc. *Indeed, in many industrial applications such as the aeronautics, space, building, biology,* etc. *linearised models are generated by dedicated tools.*

1.2.2 Controllability and observability concepts and gramians

Based on the realisation given in (1.6), let us now pay attention to the notion of gramians, playing an important role in the model approximation, as well as in many control engineering problems (norm computation, stability, sensors-actuators placements *etc.*). Additional details can be found in *e.g.* in the works of Moore (1981); Safonov and Chiang (1989); Antoulas et al. (2001) and more recently in the Ph.D. thesis of Himpe (2017).

Infinite gramians

Controllability (or reachability) $\mathcal{P}$ and observability (or detectability) $\mathcal{Q}$ gramians are useful matrices in system theory because they are related to physical system properties, independently to its realisation. The minimal energy ϵ_r required to drive a system from the state 0 to $\mathbf{x}_0$ is given by $\epsilon_r = \mathbf{x}_0^T \mathcal{P}^{-1} \mathbf{x}_0$. The larger ϵ_r is, the harder it is to reach. Similarly, the maximal observation energy ϵ_o obtained by releasing a system from an initial state $\mathbf{x}_0$ without input feeding, is given by $\epsilon_o = \mathbf{x}_0^T \mathcal{Q} \mathbf{x}_0$. A small ϵ_o means that the state is hard to observe. Based on the transfer function (1.7) and its realization (1.6), the controllability and observability infinite time-domain gramians are defined as

$$\mathcal{P} = \int_0^\infty e^{E^{-1}At} E^{-1} B B^T E^{-T} e^{A^T E^{-T} t} dt \text{ and } \mathcal{Q} = \int_0^\infty e^{A^T E^{-T} t} C^T C e^{E^{-1} A t} dt$$

or in the frequency-domain, with $T(\nu) = (\imath\nu E - A)^{-1}$,

$$\mathcal{P} = \frac{1}{2\pi} \int_{-\infty}^{\infty} T(\nu) B B^T T(\nu)^H d\nu \text{ and } \mathcal{Q} = \frac{1}{2\pi} \int_{-\infty}^{\infty} E^T T(\nu)^H C^T C T(\nu) E d\nu. \tag{1.9}$$

An important gramian property is that their computation can be obtained by solving of the following Lyapunov equations:

$$A\mathcal{P}E^T + E\mathcal{P}A^T + BB^T = 0 \text{ and } A^T\mathcal{Q}E + E^T\mathcal{Q}A + C^TC = 0. \tag{1.10}$$

Frequency-limited gramians

If instead of infinite gramians, *i.e.* the one with infinite integral bounds in (1.9), one is interested in the frequency-limited ones, following Gawronski (2004), the frequency-limited reachability and observability $\mathcal{P}_\omega$ and $\mathcal{Q}_\omega$ gramians can be defined as,

$$\mathcal{P}_\omega = \frac{1}{2\pi} \int_{-\omega}^{\omega} T(\nu) B B^T T^H(\nu) d\nu \text{ and } \mathcal{Q}_\omega = \frac{1}{2\pi} \int_{-\omega}^{\omega} E^T T^H(\nu) C^T C T(\nu) E d\nu,$$

where $T(\nu) = (\imath\nu E - A)^{-1}$. Similarly to the infinite case, they may alternatively be obtained by solving the following two Lyapunov equations,

$$A\mathcal{P}_\omega E^T + E\mathcal{P}_\omega A^T + W_c(\omega) = 0 \text{ and } A^T\mathcal{Q}_\omega E + E^T\mathcal{Q}_\omega A + W_o(\omega) = 0,$$

where

$$W_c(\omega) = S(\omega)BB^T + BB^T S^H(\omega) \text{ and } W_o(\omega) = S^H(\omega)C^TC + C^TCS(\omega)$$

and

$$S(\omega) = \frac{1}{2\pi}\int_{-\omega}^{\omega} T(\nu)d\nu = \frac{\imath}{2\pi}\mathbf{logm}\big((A+\imath\omega E)(A-\imath\omega E)^{-1}\big),$$

where $\mathbf{logm}(\cdot)$ denotes the matrix logarithm operator. By denoting $\Omega = [\omega_1, \omega_2]$ and $W_c(\Omega) = W_c(\omega_2) - W_c(\omega_1)$ and $W_o(\Omega) = W_o(\omega_2) - W_o(\omega_1)$, the Ω frequency-limited gramians are the solutions of the Lyapunov equations

$$A\mathcal{P}_\Omega E^T + E\mathcal{P}_\Omega A^T + W_c(\Omega) = 0 \text{ and } A^T\mathcal{Q}_\Omega E + E^T\mathcal{Q}_\Omega A + W_o(\Omega) = 0. \tag{1.11}$$

The case where a **DAE** is deflective is treated in Imran and Ghafoor (2015), with application to model reduction. In Chapter 3, these gramians will play an important role for approximating a model over a bounded frequency support.

1.2.3 Closest stable descriptor model in the $\mathcal{RH}_2$ and $\mathcal{RH}_\infty$ spaces

Before detailing an other linear frequency-domain model representation, let us provide two main results based on the realisation form of dynamical systems. In Kohler (2014), the best approximation of an unstable descriptor model equipped with a realisation, by a stable one in the respective $\mathcal{RH}_2$ and $\mathcal{RH}_\infty$ spaces, is defined. Mathematically, following notations of (1.8), given $p = \{2, \infty\}$ and a realisation $\mathcal{S} \in \mathbb{S}^0_{n,n_y,n_u}$ *i.e.* $\mathbf{H} \in \mathcal{RL}_\infty$ or $\mathcal{RL}_2$, one aims at finding $\tilde{\mathcal{S}} \in \mathbb{S}^+_{\tilde{n},n_y,n_u}$ $(0 < n \leq \tilde{n})$ such that,

$$||\mathbf{H}(\mathcal{S}) - \mathbf{H}(\tilde{\mathcal{S}})||_{\mathcal{H}_p} = \inf_{\mathcal{G}\in\bigcup_{\tilde{n}\in\mathbb{N}}\mathbb{S}^+_{\tilde{n},n_y,n_u}} ||\mathbf{H}(\mathcal{S}) - \mathbf{H}(\mathcal{G})||_{\mathcal{L}_p}. \tag{1.12}$$

For $p = \infty$, this problem is known as the Nehari one (see *e.g.* Francis and Doyle, 1987). In Kohler (2014), the following two main results are stated (note that in both cases we do not consider any eigenvalue on the imaginary axis). By reminding that $\mathcal{L}_2(\imath\mathbb{R}) = \mathcal{H}_2(\mathbb{C}_+) \oplus \mathcal{H}_2(\mathbb{C}_-)$, one may write

$$\mathcal{S} = \mathcal{S}_+ \oplus \mathcal{S}_-,$$

where $\mathcal{S} \in \mathbb{S}^0_{n,n_y,n_u}$, $\mathcal{S}_+ = (E_+, A_+, B_+, C_+, D_+) \in \mathbb{S}^+_{n_+,n_y,n_u}$ and $\mathcal{S}_- = (E_-, A_-, B_-, C_-, D_-) \in \mathbb{S}^-_{n_-,n_y,n_u}$, $n = n_- + n_+$. Therefore, the following theorem holds true.

Theorem 1.2: Optimal $\mathcal{RH}_2$ approximation

Given $\mathcal{S} \in \mathbb{S}^0_{n,n_y,n_u}$ (*i.e.* $\mathbf{H} \in \mathcal{L}_2$), then $\mathcal{S}_+$ solves the problem (1.12) for $p = 2$ and we have

$$\inf_{\mathcal{G}\in\bigcup_{\tilde{n}\in\mathbb{N}}\mathbb{S}^+_{\tilde{n},n_y,n_u}} ||\mathbf{H}(\mathcal{S}) - \mathbf{H}(\mathcal{G})||_{\mathcal{L}_2} = ||\mathbf{H}(\mathcal{S}) - \mathbf{H}(\mathcal{S}_+)||_{\mathcal{H}_2} = ||\mathbf{H}(\mathcal{S}_-)||_{\mathcal{H}_2(\mathbb{C}^-)}.$$

The above theorem states that the best stable rational approximant of an unstable model in the $\mathcal{RH}_2$ sense, is the stable part of the original model. Similarly, the following theorem holds true for the $\mathcal{RH}_\infty$ case.

Theorem 1.3: Optimal $\mathcal{RH}_\infty$ approximation

Given $\mathcal{S} \in \mathbb{S}^0_{n,n_y,n_u}$ (*i.e.* $\mathbf{H} \in \mathcal{RL}_\infty$), $\mathcal{S}_+$ and $\mathcal{S}_-$, then problem (1.12) for $p = \infty$ is solved by

$$\mathcal{S}_+ \oplus P\mathcal{S}_{S_{-,\sigma_1}}Q$$

where $\mathcal{S}_{S_{-,\sigma_1}}$ is given as

$$\mathcal{S}_{S_{-,\sigma_1}} = \Big(\underbrace{E_-^T R_{\mathcal{S}_{-,\gamma}}}_{\tilde{E}}, \underbrace{-A_-^T R_{\mathcal{S}_{-,\gamma}} - C_-^T C_{\mathcal{S}_{-,\gamma}}}_{\tilde{A}}, \underbrace{E_-^T \mathcal{Q}_- B_-}_{\tilde{B}}, \underbrace{C_- \mathcal{P}_- E_-^T}_{\tilde{C}}, \underbrace{D_-}_{\tilde{D}}\Big)\Big|_{\gamma=\sigma_1},$$

where $R_{\mathcal{S}_{-,\gamma}} = \mathcal{Q}_- E_- \mathcal{P}_- E^T - \gamma^2 I$ and $\mathcal{P}_-$ and $\mathcal{Q}_-$ are the solutions of the generalised Lyapunov equations (where σ_1 is the largest singular value of the unstable part),

$$A_- \mathcal{P} E_-^T + E_- \mathcal{P} A_-^T + B_- B_-^T = 0 \text{ and } A_-^T \mathcal{Q} E_- + E_-^T \mathcal{Q} A_- + C_-^T C_- = 0.$$

Moreover, if $r = \mathbf{rank}(\tilde{A})$, the projection matrices $P, Q^T \in \mathbb{R}^{r \times n_-}$ exist, such that $P\tilde{A}Q$ is regular. In this case,

$$\inf_{\mathcal{G} \in \bigcup_{\hat{n} \in \mathbb{N}} \mathbb{S}^+_{\hat{n}, n_y, n_u}} ||\mathbf{H}(\mathcal{S}) - \mathbf{H}(\mathcal{G})||_{\mathcal{L}_\infty} = \sqrt{\max \sigma \left(E_-^T \mathcal{Q}_- E_- \mathcal{P}_-\right)} = \sigma_1.$$

The above two theorems thus provide a powerful method to approximate any unstable model by a stable one. These theorems will be very important in the model-based and data-driven approximation by a realisation-based model since they allow enforcing the $\mathcal{RH}_2$ and $\mathcal{RH}_\infty$ stability property, also known as $\mathcal{H}_\infty$ and $\mathcal{H}_2$ stability. This property is obviously of great importance in the perspective of simulation, control design and performance measurement and analysis.

1.2.4 LTI models pole residue representation

The state-space realisation $\mathcal{S}$ provides a time-domain representation of a finite dimensional model. It is quite largely used in control theory since it embeds an internal knowledge of the state variation (representing the energy storage of the system), while the transfer function $\mathbf{H}$ stands as a frequency-domain "input-output" relation. Here, let us go back to the "input-output" representation given in (1.7)

$$\begin{array}{rccl} \mathbf{H}(\mathcal{S}): & \rho(E,A) & \mapsto & \mathbb{C}^{n_y \times n_u} \\ & s & \mapsto & C(sE-A)^{-1}B + D. \end{array}$$

Based on this frequency(complex)-domain representation, and considering that all singularities are simple, the i-th residue of $\mathbf{H}(s)$ with respect to λ_i is given as

$$\Phi_i = \lim_{s \to \lambda_i} (s - \lambda_i)\mathbf{H}(s).$$

In addition, if matrices are real, then any complex eigenvalue λ_i of the pencil (E, A) is accompanied with its conjugate $\overline{\lambda_i}$ and $\overline{\Phi_i}$ is the residue associated to $\overline{\lambda_i}$. Then, depending on the poles (eigenvalues) multiplicity, and considering that E is non-singular and (E, A) diagonalisable, one can then write the so-called poles - residues decomposition, or spectral representation, as

$$\begin{array}{rccl} \mathbf{H}(\mathcal{S}): & \rho(E,A) & \mapsto & \mathbb{C}^{n_y \times n_u} \\ & s & \mapsto & \displaystyle\sum_{i=1}^{n} \frac{\Phi_i}{s - \lambda_i} + R_\infty, \end{array}$$

where R_∞ is the constant contribution at $s \to \infty$ and Φ_i is obtained from right $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ and left $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_n]$ eigenvectors as,

$$\Phi_i = \mathbf{c}_i^T \mathbf{b}_i = (CX\mathbf{e}_i)(\mathbf{e}_i^T Y B) \in \mathbb{C}^{n_y \times n_u},$$

where $\mathbf{e}_i \in \mathbb{R}^n$ denotes the vector with all entry equal to 0 but the i-th equal to 1 and where $\mathbf{rank}(\Phi_i) = 1$.

In the case where (E, A) is deflective, Jordan block might appear in the diagonalisation process and a particular attention should be taken in the decomposition (see Vuillemin et al. (2014b) or Vuillemin (2014)

for details). Moreover, when the **DAE** has an infinite eigenvalue, then the system contains a polynomial part that may grow unboundedly as $s \to \infty$. This is a difference between a **DAE** system and **ODE** one, rendering many analysis tools inappropriate. In this case, the transfer function reads

$$\mathbf{H}(s) = C(sE - A)^{-1}B + D = \sum_{i=1}^{n} \frac{\Phi_i}{s - \lambda_i} + \mathbf{P}(s) = \mathbf{G}(s) + \mathbf{P}(s),$$

where $\mathbf{G}(s)$ is the strictly proper part and $\mathbf{P}(s)$, the polynomial one. Consequently, the associated $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms are unbounded and it is then standard to treat $\mathbf{P}$ a part. This case, treated *e.g.* in Beattie and Gugercin (2016) in the context of model approximation, is not covered in this manuscript. However, this remark should be kept in mind when choosing the parametrisation of the approximated model (see Chapter 2). From now on, we get rig of the polynomial part and will consider realisations and transfer functions where the (E, A) pencil is regular.

Example 2 - Finite vs. infinite meromorphic functions

In this example, two meromorphic functions are considered. First, *(i)* from the COMPl$_e$ib Leibfritz (2003) library, the Los Angeles Hospital building, mapping the ground excitation to the top lateral acceleration, described by a finite order/dimension realisation ($n = 48$) and a meromorphic function defined as $\mathbf{H}_1(z) = C(zI - A)^{-1}B$. Second, *(ii)* the following trigonometric meromorphic function

$$\mathbf{H}_2(z) = \frac{\cos(z/10)}{\cosh(1 + z) - z/10}.$$

In both cases, $z = a + ib$ is a complex number. In this academic example, both functions have been chosen for illustration purpose, however, in Chapter 4 an irrational trigonometric function will be involved for the dynamical characterisation of an open-channel (it is regularly the case when solving a **PDE** by Laplace transform). Both Figures 1.1 and 1.2 illustrate the evaluation of $|\mathbf{H}_1|$ and $|\mathbf{H}_2|$ (in dB) along the imaginary axis (red lines) and over a part of the complex space (coloured surface). In addition, Figure 1.1 also shows green dots, representing the stable poles and zeros of $\mathbf{H}_1$, respectively (poles tends to ∞, while zeros tends to 0).

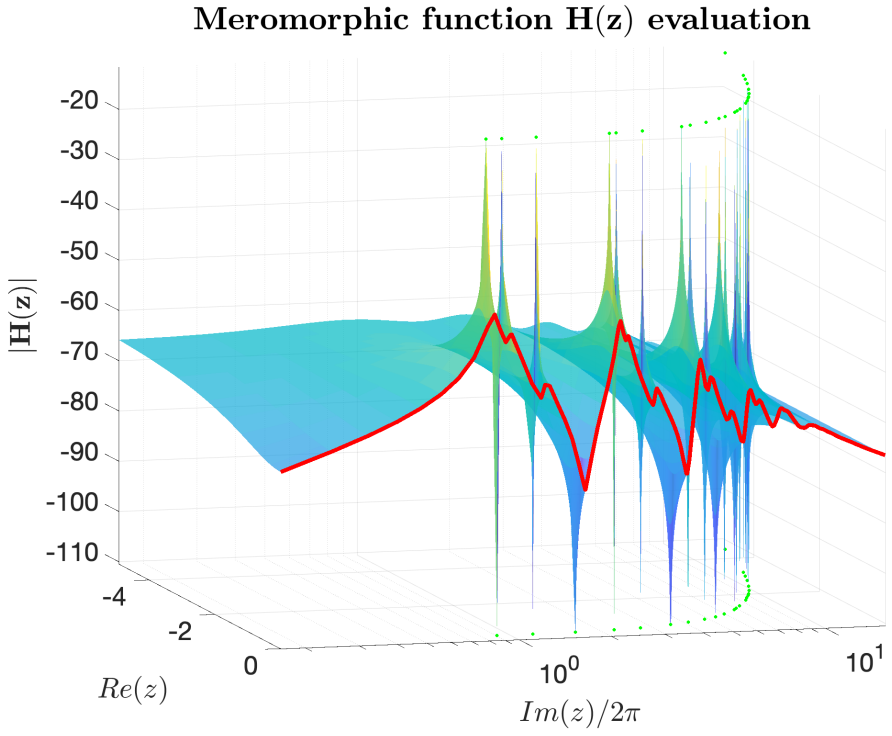


Figure 1.1: 3D meromorphic function visualisation of $\mathbf{H}_1$.

$\mathbf{H}_2$ clearly shows to have an infinite number of the singularities (illustrated by the pics). These later should be analytically computed rather than numerically. With reference to these two figures, one can see that every singularities are represented by an infinite value and zeros with a zero of the function. This representation also shows the stability property of the function. Traditionally, in the control theory, the cut section at $\Re(z) = 0$ *i.e.* the evaluation of the transfer function along the complex vertical axis $\imath\omega$ (for $\omega > 0$), is achieved by the well known Bode diagram. This "3D Bode" response still provides additional interesting informations.

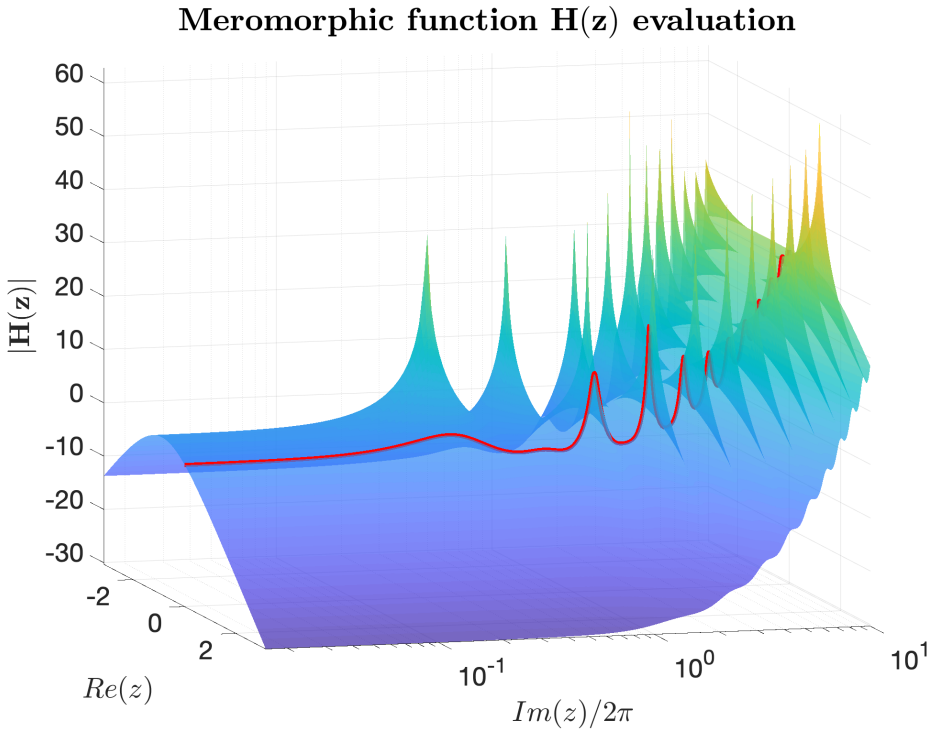


Figure 1.2: 3D meromorphic function visualisation of $\mathbf{H}_2$.

1.3 Some linear algebra and computational issues

So far, a bundle of results and definitions attached to linear systems have been recalled. Most of them are well known and, since they are grounded on linear algebra manipulations, they offer an apparently simple framework. However, despite this very appealing context, reader should keep in mind the fact that in model approximation (or reduction), all the above mentioned formula should be considered in a very large quantity. As a consequence, within the model approximation community, attention to the numerical aspects must be paid to ensure the scalability of developed approaches. This is indeed a specificity of model approximation which strongly links the theoretical aspects with the computational ones, bridging the gap between theory and practice. To the author's point of view, computational aspects are somehow the cornerstone in model approximation, and theoretical advances require numerical ones to become effective and attractive for practitioners. In this section we try to give a preview of theses practical aspects by describing some underlying issues in computational arithmetic of some important operations in linear model approximation and control engineering. We start with the eigenvalue problem, then, the linear equation and sum computation ones. Finally, a glimpse on the Lyapunov equations solvers is given. Then, some consideration regarding the $\mathcal{H}_2$, $\mathcal{H}_{2,\Omega}$ and $\mathcal{H}_\infty$-norm computations will be given. For complete and very interesting references in linear algebra, reader may refer to Saad (2000); Antoulas (2005); Van der Vorst (2010); Bai et al. (2010) books or interesting works in the computer and computational scientific communities such as the book of Higham (2002) or papers from Stewart (2000) or Nguyen and Revol (2011)[7].

1.3.1 The eigenvalues problem computation

For both practitioners and researchers, the generalised eigenvalues problem is a pivotal tool in many applications such mechanical, electrical, vibration, acoustics, control theory *etc.* (see Saad, 2000, Chapter 10). Its efficient resolution is thus crucial for many system manipulations, resolution and understanding. Obviously, so it is in the large-scale model approximation context. The standard problem consists in finding the non-trivial right and left eigenvalues/vectors $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{C}^n$ and eigenvalues $\lambda_i \in \mathbb{C}$, of the pair $(E, A) \in \mathbb{C}^{n\times n} \times \mathbb{C}^{n\times n}$, for $i = 1, \ldots, n$, such that

$$A\mathbf{x}_i = \lambda_i E\mathbf{x}_i \text{ and } \mathbf{y}_i^H A = \lambda_i \mathbf{y}_i^H E.$$

Let us give a brief overview of some of the methods solving this problem. For more detailed descriptions we refer to the books of Bai et al. (2010); Van der Vorst (2010) and the very comprehensive report of Kurschner (2010). Eigenvalues computation are usually distinguished between full space methods for dense matrices of moderate size (*e.g.* $n < 5,000$) and iterative subspace methods for very large and sparse matrices (*e.g.* n up to $10^9 \ldots$).

Direct (dense) approaches

Full space methods compute the complete eigenvalues set and, if necessary, the eigenvectors $(\mathbf{x}_i, \mathbf{y}_i)$. They are referred to as direct methods, even if in practice they also embed an iterative scheme. Usually, they transform the original matrices to diagonal or triangular ones. For the standard eigen-problem ($E = I_n$) the QR method can be used to compute a Schur decomposition. The QZ method computes a generalised Schur decomposition of a pair (E, A). Full space methods usually have a complexity of $\mathcal{O}(n^3)$. As a matter of consequence, they are largely used in control engineering but have a limited applications physics, such as in fluid mechanics applications, where dimension rapidly explodes.

Indirect (sparse) approaches

For large-scale sparse matrices (here we assume $E = I_n$ for simplicity), largely used in scientific computations, iterative subspace methods come into the picture. In this case, one usually focuses on a fraction of the spectrum, only. The underlying idea is to project the original matrix A onto a lower dimensional subspace

[7] Moreover, interested reader may refer to the Matrix Market webpage to measure the importance of the related problems `http://math.nist.gov/MatrixMarket/`

imposing some Ritz-Galerkin condition on the approximate eigen-pair $(\hat{\lambda}, \hat{\mathbf{x}})$ obtained from the resolution of the lower dimensional problem with dense methods. The basic procedure consists in computing the approximated eigenvector $\hat{\mathbf{x}}$ as

$$\hat{\mathbf{x}} = V_k \hat{\mathbf{x}}_k \in \mathbb{C}^n,$$

where $\hat{\mathbf{x}}_k \in \mathbb{C}^k$ and $V_k \in \mathbb{C}^{n \times k}$ is a k-th order orthonormal basis spanning the search space $\mathcal{V}_k \subset \mathbb{C}^n$ ($k \ll n$). The residual value r of the approximated eigen-pair should ensure

$$r = A\hat{\mathbf{x}} - \hat{\lambda}\hat{\mathbf{x}} \perp \mathcal{V}_k \text{ or } V_k^H A V_k \hat{\mathbf{x}}_k - \hat{\lambda}\hat{\mathbf{x}}_k = 0.$$

Then, $(\hat{\lambda}, \hat{\mathbf{x}}_k)$ is an eigen-pair of the reduced matrix $V_k^H A V_k \in \mathbb{C}^{k \times k}$ which can easily be computed using dense approaches. The approximated eigenvector $\hat{\mathbf{x}}_k$ is simply lifted up by V_k to obtain $(\hat{\lambda}, \hat{\mathbf{x}})$, also known as the Ritz pair of A. This process is generally repeated by adding a new basis vector until accuracy is reached. In the Petrov-Galerkin approach, a second basis W_k spanning $\mathcal{W}_k \subset \mathbb{C}^n$ enters in the process (instead of $\mathcal{V}_k$ only). This subspace $\mathcal{W}_k$ allows ensuring

$$r = AV_k\hat{\mathbf{x}}_k - \hat{\lambda}V_k\hat{\mathbf{x}}_k \perp \mathcal{W}_k \text{ or } W_k^H A V_k \hat{\mathbf{x}}_k - \hat{\lambda} W_k^H V_k \hat{\mathbf{x}}_k = 0.$$

The $\mathcal{V}_k$ space is the projection space and the $\mathcal{W}_k$ one is known as the test space. A well known and used class of iterative methods are the Krylov ones where the subspace $\mathcal{V}_k$ is spanned by the following Krylov subspace, for some initial vector $\mathbf{v}_1 \in \mathbb{C}^n$:

$$V_k = \mathcal{K}(A, \mathbf{v}_1, k) \triangleq \mathbf{span}\left(\mathbf{v}_1, A\mathbf{v}_1, \ldots, A^{k-1}\mathbf{v}_1\right). \tag{1.13}$$

This subspace can be constructed using the modified Lanczos or Arnoldi procedures. The latter one produces a sequence of orthonormal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k \in \mathbb{C}^n$ spanning the Krylov subspace $\mathcal{K}(A, \mathbf{v}_1, k)$ and an upper Hessenberg matrix $H_k = V_k^H A V_k$. The Arnoldi iteration ensures at the k-th iteration

$$AV_k = V_{k+1}\overline{H}_k = V_k H_k + \mathbf{h}_{k+1,k}\mathbf{e}_k^T,$$

where $H_k \in \mathbb{C}^{k \times k}$ and $\mathbf{h}_{k+1,k} \in \mathbb{C}^n$ is the $k+1$-th column of $\overline{H}_k = [H_k;\ \mathbf{h}_{k+1,k}] \in \mathbb{C}^{(k+1)\times k}$. As the iteration k increases, eigenvalues of H_k tend to those of A. More specifically, when spanning $\mathcal{K}(A, \mathbf{v}_1, k)$, the largest magnitude eigenvalues are firstly obtained, $\mathcal{K}(A^{-1}, \mathbf{v}_1, k)$ leads to the smallest magnitude ones and $\mathcal{K}((\sigma I - A)^{-1}, \mathbf{v}_1, k)$ the ones closest to $\sigma \in \mathbb{C}$ (see also Simoncini, 2010). This remark will play an important role in some iterative model approximation schemes, and more specifically, the interpolatory ones. In the eigenvalues problem, the $(\sigma I - A)^{-1}$ operation is also known as a pre-conditioning method, aiming at accelerating the convergence of the procedure, *i.e.* by using $(\sigma I - A)^{-1}\mathbf{v}_1$ instead of $\mathbf{v}_1$. Other pre-conditioner, like the m-th order polynomial Chebyschev one $T_m(A - \sigma I)$ can also be employed. Basically, a pre-conditioned problem consists in constructing

$$V_k = \mathcal{K}(TA, T\mathbf{v}_1, k) = \mathbf{span}\left(T\mathbf{v}_1, TA\mathbf{v}_1, \ldots, (TA)^{k-1}\mathbf{v}_1\right).$$

Remark 1.7 (A "control-systems" view) *Interestingly, if considering a **SISO LTI** model, $\mathcal{K}(A, B, n)$ is spanned by the reachability matrix $\mathcal{R} = [B, AB, \ldots, A^{n-1}B] \in \mathbb{R}^{n \times n}$, largely used in control theory. Then, by noting that $A\mathcal{R} = \mathcal{R}H_n$, where H_n is the controlability canonical form with upper Hessenberg structure, and by applying a QR factorisation $\mathcal{R} = VU$ (where V is an orthonormal basis and U an upper triangular matrix), one obtains $AV = VUHU^{-1}$. Since H_n is upper Hessenberg, so it is for UHU^{-1}. Therefore, the Krylov subspace can be viewed as a generalisation of the controllability matrix (obviously, similar results stand for the observability one).*

Example 3 - Eigenvalues via Krylov

In this example we aim at illustrating some of the numerical issues that can be encountered when solving the eigenvalue problem and the importance of a reliable numerical approach versus an intuitive one. In this example we construct the toy dynamical model which eigenvalues of the A matrix are logarithmically spaced as $\lambda_i = 10^{-1}, \ldots, 10^2 \times (-1 \pm \imath)$, for $n = 50$ and for $B = C^T = \mathbf{1}_n$ (the example can easily be scaled-up *e.g.* to $n = 10^9$). Then, eigenvalues approximation is obtained through the projection of the A matrix through projector V_k, obtained by three different methods. First, the "naive" approach involving the reachability matrix, second, a projection based on the Krylov $\mathcal{K}(A, B, k)$ subspace, and finally a projection onto a the pre-conditioned Krylov $\mathcal{K}(T_\rho A, T_\rho B, k)$ subspace, with $T_\rho = A - \rho I$ ($\rho \in \mathbb{R}_+$). Both Krylov spaces are constructed with an Arnoldi procedure, embedded in the **MOR** toolbox. The following MATLAB code illustrates the procedure and Figure 1.3 shows the convergence results after $k = 10$ iterations, for $\rho = 0$.

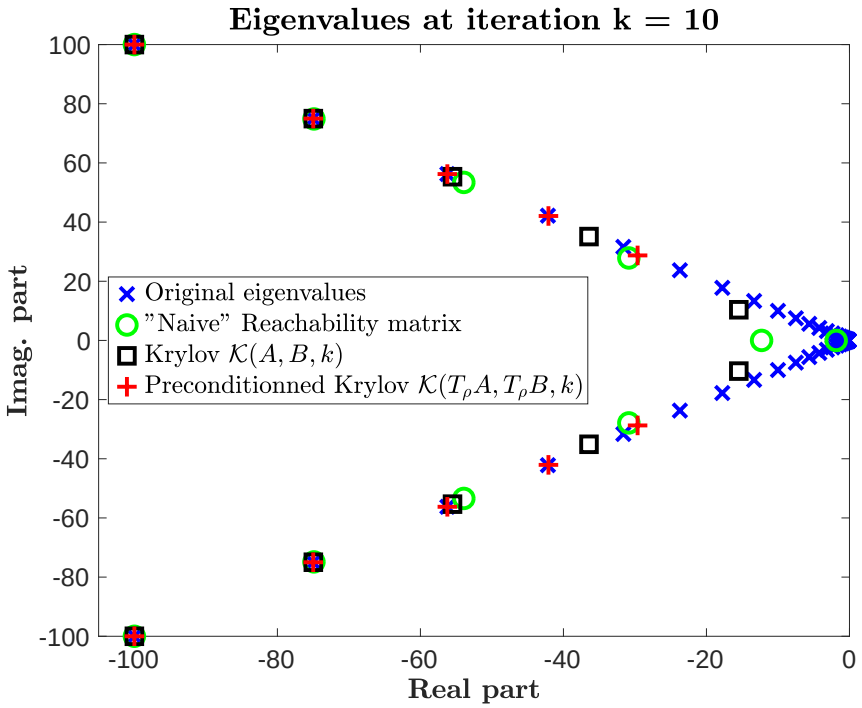


Figure 1.3: Eigenvalues estimation using different algorithms.

```
clear all, close all, clc
%% System definition
n      = 50;
cplxI  = logspace(-1,2,n/2);
cplxR  = -logspace(-1,2,n/2);
lambda = sort(complex([cplxR cplxR],[cplxI -cplxI])); % Eigenvalues of A
A      = diag(lambda);
[V,D]  = eig(A);
[V,A]  = cdf2rdf(V,D);
B      = ones(n,1);
C      = ones(1,n);
sys    = ss(A,B,C,0); % Construction of the LTI model
sysTF  = tf(sys);     % Transfer function
eigA   = eig(A);      % Eigenvalues
n      = length(A);
%% "Control" approach using the reachability matrix
R = ctrb(A,B); %[B A*B A^2*B ... A^nB]
% Canonical dynamical matrix form of the system
```

```
H       = triu(ones(n,n),-1)-triu(ones(n,n));
H(:,n) = flip(-sysTF.den{1}(2:n+1)');
% A*R = R*H check
norm(A*R - R*H) % = 6.4728e+86, instead of 0 (works better for n=10)
% QR decomposition of the reachability matrix
[V,U] = qr(R); % V'*V = I
% A*V = V*U*H*U^{-1} check
norm(A*V - V*U*H/U) % 5.0933e+05 (instead of 0)
Hbar = U*H/U;        % Projected matrix
%% "Numerical" Krylov subspace approach
% Krylov space generation
sig = inf;
Vk  = mor.alg.base.Krylov(A,eye(n),sig,B,n);
% Krylov with precondionner
rho = 1e-3; % Try also with 1e2 ... 1e5
T   = A-rho*eye(n);
Vk2 = mor.alg.base.Krylov(T*A,eye(n),sig,T*B,n);
% For k-eigenvalues, plot the approximated ones
k    = 10;
eig1 = eig(Hbar(1:k,1:k));
eig2 = eig(Vk(:,1:k)' * A * Vk(:,1:k));
eig3 = eig(Vk2(:,1:k)'* A * Vk2(:,1:k));
```

Listing 1.1: **demo_Chap1_Arnoldi** script: illustration of eigenvalue approximation.

Clearly, the reachability matrix-based approach makes difficulties in finding the eigenvalues and numerical issues already appear at the matrix construction step (note that for $n = 10$, no numerical problem appear). Then, both Krylov-based methods tend to first approximate the eigenvalues of largest magnitude, and interestingly, the pre-conditioned one with $\rho = 0$, clearly converges faster. Indeed, after 10 iterations, the pre-conditioned version already exactly catches 8 eigenvalues while the standard one only 4 (plus 2 approximately). In addition, at iteration $k = 10$, when comparing the $V_k^T A V_k$ (resp. $V_k^T T_\rho A V_k$) matrices obtained without (resp. with) pre-conditioners, one obtains the Figures 1.4 and 1.5. Both exhibit an upper Hessenberg form which is even more a tridiagonal one for the pre-conditioned version with $T = A$, standing as a perfect pre-conditioner in the case where eigenvalues at ∞ are expected. In addition, as illustrated on Figure 1.6 (resp. 1.7), an interesting numerical property appears when increasing ρ, with $\rho = 10^{-3}$ (resp. $\rho = 10^3$): the Hessenberg form tends to converge to the non pre-condioned one.

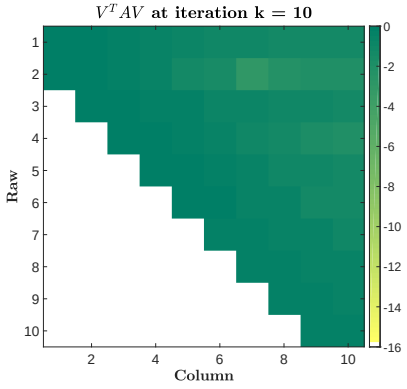


Figure 1.4: Matrix coefficients magnitude. Hessenberg form for $\mathcal{K}(A, B, k)$.

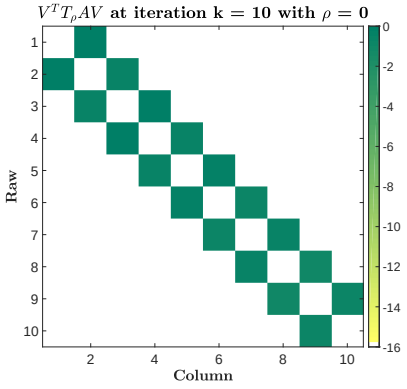


Figure 1.5: Matrix coefficients magnitude. Hessenberg form for $\mathcal{K}(T_\rho A, T_\rho B, k)$ and $\rho = 0$.

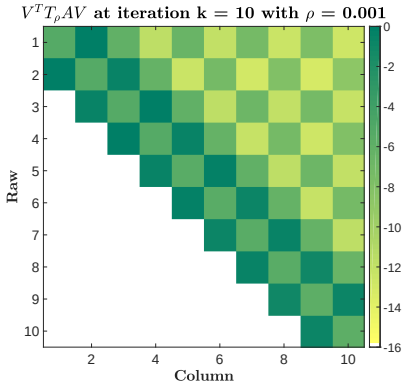


Figure 1.6: Matrix coefficients magnitude. Hessenberg form for $\mathcal{K}(T_\rho A, T_\rho B, k)$ and $\rho = 10^{-3}$.

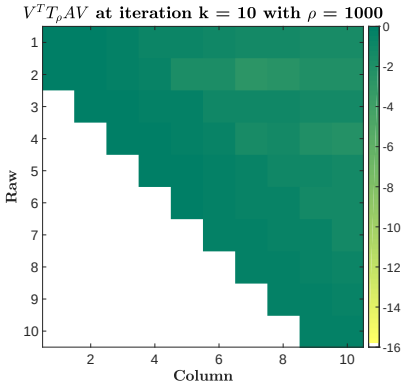


Figure 1.7: Matrix coefficients magnitude. Hessenberg form for $\mathcal{K}(T_\rho A, T_\rho B, k)$ and $\rho = 10^{3}$.

Arnoldi (or Lanczos) however, exhibit slow convergence and tend to have difficulties finding interior eigenvalues. Another important class of subspace methods are the Jacobi-Davidson ones. Initially proposed by Sleijpen and Van der Vorst (1996) for the standard linear eigenvalues problem, is has been improved and generalised in several ways to handle various problems, for instance generalised eigenvalues problem by Rommes (2008), quadratic and polynomial eigenvalues problem in Hochstenbach and Sleijpen (2008), singular values problem in Hochstenbachi (2001), and even nonlinear eigenvalues problem in Schreiber (2008). Without entering into details, the Jacobi-Davidson method is grounded on two main important phases: *(i)* the Davidson principle where a Galerkin projection is applied as previously, but where the subspace $\mathcal{V}_k$ is not necessarily a Krylov one (known as the subspace extraction) and *(ii)* a correction for the Ritz vector in which the search space $\mathcal{V}_k$ is enlarged by adding a new basis vector to it, hopefully leading to better approximate eigen-pairs in the next extraction phase (known as the expansion phase). In many cases, the subspace projection, unlike the Arnoldi or Lanczos processes, does not generate a Hessenberg approximated matrix. The Ritz correction uses a Newton scheme and the correction term added in the orthogonal complement of V_k. Additional details are given in Kurschner (2010) and in the reference handbook of Saad (2000).

From a practitioner point of view, keeping the above simple considerations is important before solving any eigenvalues problem. The MATLAB software embeds some eigenvalue solvers: the **`eig`** for dense problems and the **`eigs`** for sparse ones. While the former is traditionally used by engineers and provides all the eigenvalues, the second, instead, allows approximating a subset of the eigenvalues with large/small magnitude/real part/imaginary part *etc.* Refer to Lehoucq and Sorensen (1996); Stewart (2000) for details.

1.3.2 The $A\mathbf{x} = \mathbf{b}$ linear equation computation

An other approximation-related problem is, given $A \in \mathbb{C}^{n\times n}$ and $\mathbf{b} \in \mathbb{C}^n$, to find $\mathbf{x} \in \mathbb{C}^n$ such that

$$A\mathbf{x} = \mathbf{b}. \tag{1.14}$$

This problem is embedded in many engineering applications, *e.g.* identification, least square regression, Krylov subspace generation *etc.* As for the eigenvalues problem, direct and iterative methods co-exist. The former have a finite number of steps and ends with the exact solution $\mathbf{x}$, provided that all arithmetic operations are exact, while the latter is adapted to sparse matrices.

Direct (dense) approaches

The most used of these methods is Gaussian elimination which is basically an elimination method transforming the A matrix into an upper or lower triangular matrix (typically applying a LU decomposition) and then apply a backward substitution.

Indirect (sparse) approaches

Among the existing iterative methods, by denoting $\mathbf{x} = [x_1, \ldots, x_n]^T$, the Jacobi one aims at solving the first line of $A\mathbf{x} = \mathbf{b}$ with $x_1 \in \mathbb{C}$, the first element of $\mathbf{x}$, then the second line with x_2 and so on. Then one makes an initial guess $\mathbf{x}^{(0)}$ on $\mathbf{x}$ and substitutes these values into the right-hand side of the rewritten equations to obtain the first approximation. After this procedure has been completed, one iteration has been performed. In the same way, the second approximation is formed by substituting the first approximation $\mathbf{x}^{(1)}$ values into the right-hand side of the rewritten equations. Repeating these iterations will form a sequence of approximations $\mathbf{x}^{(k)}$ that often converges to the actual solution. A stopping criteria can be when $\mathbf{x}^{(k+1)} \approx \mathbf{x}^{(k)}$. Obviously, as many iterative methods, greater accuracy would require more iterations.

Similarly to Jacobi's approach, in the Gauss-Seidel method, one uses the new values of each i-th element of $\mathbf{x}$ as soon as they are known. That is, once x_1 has been determined from the first equation, its value is then injected in the second one to obtain the new x_2. Similarly, the new x_1 and x_2 are used in the third equation to obtain the new x_3 and so on.

However, it is possible to apply the Jacobi method or the Gauss-Seidel method to a system of linear equations and obtain a divergent sequence of approximations. Still, for a special type of coefficient matrix A, called *strictly diagonally dominant matrix*, a convergence result can be stated as in Theorem 1.4. A matrix A is strictly diagonally dominant if the absolute value of each entry on the main diagonal is greater than the sum of the absolute values of the other entries in the same row. For this class of matrices, the following holds true.

Theorem 1.4: Convergence of the Jacobi and Gauss-Seidel methods

Given $A \in \mathbb{C}^{n\times n}$, strictly diagonally dominant, then the system of linear equations given by (1.14), has a unique solution to which the Jacobi method and the Gauss-Seidel method will converge for any initial approximation.

The above theorem states that before performing any linear system resolution, a column / row permutation and scaling is generally needed to ensure numerical stability of the initial problem. In addition, as in the eigenvalues problem, pre-conditioning is usually performed, *i.e.* replace $A\mathbf{x} = \mathbf{b}$ with $T^{-1}A\mathbf{x} = T^{-1}\mathbf{b}$ with T a matrix "close" to A. Many numerical efficient tools, involving pre-conditioners, such as **CG**, **BiCG** or **GMRES** implement these methods with varying specificity, depending on the structure of the A matrix (*e.g.* sparsity, symmetric structure, *etc.*). Finally, one should also note that the associated complexity of linear equation is $\mathcal{O}(n^3)$. The MATLAB software embeds some interface with the **LAPACK** solvers: the code **`x=A\b`** performs the resolution for both dense and sparse problems and provides the "exact" solution. Moreover, an approximation $\hat{\mathbf{x}}$ is also possible using the **`gmres`** function implementing the **GMRES** method with or without pre-conditioner, see *e.g.* Saad and Schultz (1986) or recent developments in Sonneveld and Van Gijzen (2008).

More recently, interval-based methods ensuring a bounded error, are also largely being investigated, as *e.g.* in Nguyen and Revol (2011). This class of method is attached to the numerical issues accompanied by the limited arithmetic floating precision and take it into consideration when solving the problem to contain the error propagation.

1.3.3 About the vector $\mathbf{x} = [x_1, \ldots, x_n]$ sum operation

In this section, we will not detail the many summation methodologies. Indeed, this problematic is close to the numerical research field and interested reader should *e.g.* refer to the work from Demmel and Nguyen (2014) and references therein. Still, from a didactical viewpoint, author believes that it is interesting to point out this operation in order to emphasise the importance of the numerical impact when dealing with a large number of data. Let us present a simple MATLAB based example to illustrate the loss of accuracy when summing values in a vector.

Example 4 - Simple sum operation loss of accuracy

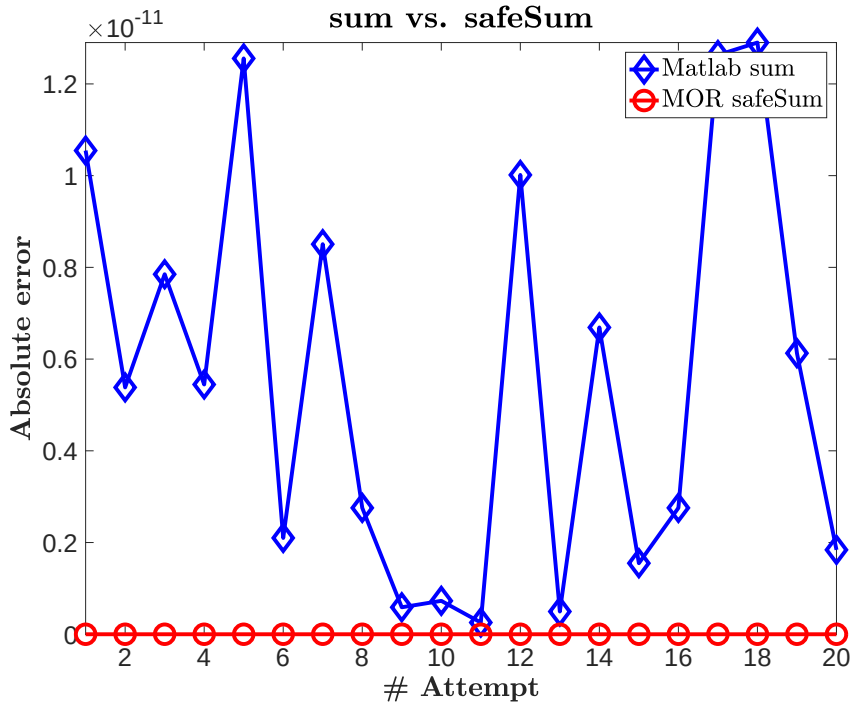


Figure 1.8: Error comparison of two implementations of the sum computation: MATLAB **`sum`** vs. **`mor.linalg.safeSum`** safe (but also very greedy) implementation used in the **MOR** toolbox.

Let us consider the three following examples through the following MATLAB code.

```
clear all, close all, clc
%% Example 1: Exact solution = 0
x = [1e12 -1e-5 -1e12 1e-5];
% Classic sum
sum(x)
% 1.0000e-05 (absolute error 1e-5)
% Now ordered sum
x = sort(x);
sum(x) % 0 (exact)
%% Example 2: Exact solution = 0.1
x = [1e12 1e-1 -1e-5 -1e12 1e-5];
% Classic sum
sum(x) % 9.9986e-02 (rel. err. 0.014%)
% Now ordered sum
x = sort(x);
sum(x) % 9.9976e-02 (rel. err. 0.024%)
% Now ordered sum by amplitude
[~,idx] = sort(abs(x));
x = flip(x(idx));
sum(x) % 1.0000e-01 (exact)
%% Example 3 : Exact solution = 0
for i = 1:2e1
    x       = [100 -1e-2*ones(1,1e4)];
    P       = randperm(numel(x));
    x       = x(P);
    err1(i) = sum(x);
    err2(i) = mor.linalg.safeSum(x);
end
```

Listing 1.2: **`demo_Chap1_Sum`** script: simple illustration of the vector sum accuracy.

This example considers first $\mathbf{x} = [10^{12}, -10^{-5}, -10^{12}, 10^{-5}]$ which sum is obviously equal to 0, then, $\mathbf{x} = [10^{12}, 10^{-1}, -10^{-5}, -10^{12}, 10^{-5}]$ which sum is equal to 10^{-1} and finally $\mathbf{x} = [100\ -10^{-2} \times \mathbf{1}_{10^4}]$ (randomly permuted 20 times).
With reference to the above code, the errors of examples 1 and 2 are shown as comments, illustrating that even on small vectors, errors can occur. Moreover, considering example 3, where computation is performed 20 times for different permutation of $\mathbf{x}$, the obtained errors are reported on Figure 1.8, showing that, depending on the ordering of vector $\mathbf{x}$, the error from the **`sum`** implementation vary in a quite important manner while the **`mor.linalg.safeSum`** always provides the exact solution.
The **`mor.linalg.safeSum`** implementation is obviously way more time and computationally demanding since values ordering is performed at every iteration in order to sum the smallest values first. Nevertheless, this algorithmic trick ensures a good accuracy, at the price of a computational effort.

Interestingly, according to the ordering sequence of the vector, the result vary in a relavitely important manner. When dealing with complex numbers, this effect might be even more amplified. It is obvious that the rounding error and propagation strongly depends on the variability of the values embedded in the vector and/or matrix. This effect can have some importance, as highlighted in the rest of the manuscript where matrix vector multiplication, sums and complex arithmetic are present at many steps of the proposed algorithms. As in the linear equation resolution, research in numerical computation are still investigating methods to perform a (simple) stable sum computation, as in Demmel and Nguyen (2014) using the Rump's algorithm.

After this brief insight on some classical issues on computational methods, let us now turn back our attention to more model reduction and control-oriented computational considerations.

1.3.4 Lyapunov equation computational considerations

As previously mentioned, the Lyapunov equations plays an important role in the gramian definition, model approximation and $\mathcal{H}_2$-norm computation, therefore, numerical scheme solving (for $E = I_n$)

$$A\mathcal{P} + \mathcal{P}A^T + BB^T = 0,$$

are of particular importance. Without entering to much in details, methods can be classified in two broad categories, the *(i)* dense (exact) approaches including the vectorisation of the Lyapunov equation, the Bartels-Stewart method (for Sylvester and Lyapunov equations) or the Hammarling's method, for stable Lyapunov equations, and *(ii)* the iterative approximate ones, including the so-called **ADI** methods for Alternating Direction Implicit or Krylov-like approaches *etc.* Let us now briefly present three direct methods schemes and some of their limitations.

The vectorisation (naive) method

By transforming $A\mathcal{P} + \mathcal{P}A^T + BB^T = 0$, into

$$\underbrace{\left(I_n \otimes A + A \otimes I_n\right)}_{\mathbb{A}} \underbrace{\mathbf{vect}(\mathcal{P})}_{\mathbf{x}} = \underbrace{\mathbf{vect}(-BB^T)}_{\mathbf{b}},$$

then solving $\mathbb{A}\mathbf{x} = \mathbf{b}$, where $\mathbb{A} \in \mathbb{R}^{n^2 \times n^2}$, $\mathbf{b} \in \mathbb{R}^{n^2}$, leads to the solution by solving a linear equation problem. The complexity of the original $n \times n$ problem is then $\mathcal{O}\big((n^2)^3\big)$. Interestingly, even if theoretically this approach seems appealing and simple, in practice it is to be avoided (obviously) due to numerical limitations.

The Bartels-Stewart method

By applying a Schur factorisation (in real arithmetic) $A = QTQ^T$ and left right multiplying by Q (orthonormal) and Q^T leads to $Q^TA\mathcal{P}Q + Q^T\mathcal{P}A^TQ + Q^TBB^TQ = 0$. Noticing that $QQ^T = I$, the Lyapunov equation becomes

$$\underbrace{Q^TAQ}_{T}\underbrace{Q^T\mathcal{P}T}_{Y} + \underbrace{Q^T\mathcal{P}Q}_{Y}\underbrace{Q^TA^TQ}_{T^T} + Q^TBB^TQ = 0$$

where

$$T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix}, \; Y = \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{12}^T & Y_{22} \end{bmatrix}, Q^T BB^T Q = \begin{bmatrix} \mathcal{C}_{11} & \mathcal{C}_{12} \\ \mathcal{C}_{21} & \mathcal{C}_{22} \end{bmatrix},$$

where the last block of T, $T_{22} \in \mathbb{R}^{n_r \times n_r}$ is of dimension $n_r = \{1, 2\}$. By separating the four blocks in the projected equation, we have four equations which can be solved by backward substitution (like the Gaussian elimination),

$$\begin{aligned} \mathcal{C}_{11} &= T_{11}Y_{11} + T_{12}Y_{21} + Y_{11}T_{11}^T + Y_{12}T_{12}^T \\ \mathcal{C}_{12} &= T_{11}Y_{12} + T_{12}Y_{22} + Y_{12}T_{22}^T \\ \mathcal{C}_{21} &= T_{22}Y_{21} + T_{21}Y_{11}^T + Y_{22}T_{12}^T \\ \mathcal{C}_{22} &= T_{22}Y_{22} + T_{22}Y_{22} \end{aligned}$$

An iterative algorithm can then be stated by solving the three last equations explicitly and by inserting the solution in the first one. Then, repeat the procedure on the reduced Lyapunov equation in $Y_{11} \in \mathbb{R}^{(n-n_r)\times(n-n_r)}$ (first equation). After iterations, the solution is finally given by

$$\mathcal{P} = Q^T Y Q.$$

This algorithm has a complexity of $\mathcal{O}(n^3)$, *e.g.* similar to an eigenvalues or linear equation problem. The MATLAB software embeds the function **`lyap`** performing the exact resolution for dense matrices, involving a version of the Bartels-Stewart method.

The Hammarling's method

By noticing that if A is stable, then $\mathcal{P} > 0$ and we can write $\mathcal{P} = UU^T$, where U is upper triangular (Schur basis). Hammarling (1982) observed that U can be computed without explicitly computing $\mathcal{P}$. By denoting

$$\mathcal{P} = UU^T = \begin{bmatrix} \mathcal{P}_{11}^{1/2} & \mathcal{P}_{12}\mathcal{P}_{22}^{-1/2} \\ 0 & \mathcal{P}_{22}^{1/2} \end{bmatrix} \begin{bmatrix} \mathcal{P}_{11}^{1/2} & 0 \\ \mathcal{P}_{12}\mathcal{P}_{22}^{-1/2} & \mathcal{P}_{22}^{1/2} \end{bmatrix},$$

the problem now consists in computing the smaller part of $\mathcal{P}$, i.e. $\mathcal{P}_{22} \in \mathbb{R}^{(n-r)\times(n-r)}$, $\mathcal{P}_{12} \in \mathbb{R}^{r\times(n-r)}$ and finally, $\hat{\mathcal{P}}_{11} \in \mathbb{R}^{r\times r}$, where $r \ll n$. To that effect, three equations have to be solved:

$$\begin{aligned} A_{22}\mathcal{P}_{22} + \mathcal{P}_{22}A_{22}^T + B_2B_2^T &= 0, \\ A_{11}\mathcal{P}_{12} + \mathcal{P}_{12}\left(\mathcal{P}_{22}^{-1}A_{22}\mathcal{P}_{22} + \mathcal{P}_{22}^{-1}B_2B_2^T\right) + A_{12}\mathcal{P}_{22} + B_1B_2^T &= 0, \\ A_{11}\hat{\mathcal{P}}_{11} + \hat{\mathcal{P}}_{11}A_{11}^T + B_1B_1^T &= 0. \end{aligned}$$

We can thus solve the first low order Lyapunov equation for $\mathcal{P}_{22}$. Then, we solve the second equation which is a Sylvester equation for $\mathcal{P}_{12}$. Finally, since (A_{11}, B_1) is reachable, the problem of solving a Lyapunov equation of size n is reduced to that of solving a Lyapunov equation of smaller size $r \ll n$. The MATLAB software embeds the function **`lyapchol`** performing the exact resolution for dense matrices, using the Hammarling's method.

Lyapunov solvers for large and sparse problems

Even if the Bartels-Stewart is both efficient and robust for large dense problems, when matrices are sparse, dedicated methods may also be constructed. Among them, the low-rank methods and the Krylov ones are still being investigated from the numerical community (see *e.g.* Simoncini (2007); Shank et al. (2016); Hached and Jbilou (2017)). The latter is concerned in computing directly a low-rank solution of the Lyapunov equation by determining $V \in \mathbb{R}^{n\times r}$ and $P \in \mathbb{R}^{r\times r}$ such that

$$\tilde{\mathcal{P}} = VPV^T,$$

where V is an orthogonal matrix. By this transformation, it is expected that $A\tilde{\mathcal{P}} + \tilde{\mathcal{P}}A^T + BB^T \approx 0$. A common approach aims at computing an approximation by the application of a Galerkin condition. More

precisely, given V, one may compute P by imposing the Galerkin condition and simplifying ($W = BB^T$)

$$\begin{aligned} V^T(A\tilde{\mathcal{P}} + \tilde{\mathcal{P}}A^T + W^T)V &= 0, \\ V^TAVPV^TV + V^TVPV^TA^TV + V^TW^TV &= 0, \\ \tilde{A}P + P\tilde{A}^T + \tilde{W} &= 0, \end{aligned}$$

where $\tilde{A} = V^TAV$. Then the Lyapunov equation to solve is now of dimension $r \ll n$ and can be solved using a dense approach. Obviously, the matrix V corresponds to a basis of a search subspace (or projection space). The success of the approach highly depends on the choice of V and many different search subspaces have been proposed in the literature (see *e.g.* Simoncini (2007) for the case of Krylov subspace).

1.3.5 $\mathcal{H}_2$-norm computation

So far, very generic linear algebra and computational methods were pointed to highlight the numerical issues that researchers and engineer are faced to. Now, following the norm definitions given in Section 1.1 while focussing on the finite dimensional framework defined in Section 1.2, let us derive some methods to efficiently compute them.

The gramian-based method

As rooted on the Lyapunov equations, when $\mathbf{H} \in \mathcal{H}_2$ and equipped with a realisation $\mathcal{S}$ (1.6), then $||\mathbf{H}||_{\mathcal{H}_2}$ is finite and can be computed as

$$||\mathbf{H}||^2_{\mathcal{H}_2} = \mathbf{tr}\left(B^T\mathcal{Q}B\right) = \mathbf{tr}\left(C\mathcal{P}C^T\right),$$

where $\mathcal{P}$ and $\mathcal{Q}$ are the controllability and observability gramians solution of the Lyapunov equations (1.10). Clearly the $\mathcal{H}_2$-norm computation is recast as a Lyapunov equation resolution. The MATLAB software embeds the function **`norm`** which allows the $\mathcal{H}_2$-norm computation using dedicated Lyapunov solvers and pre-conditioning methods (such as QR decomposition, *etc.*).

Remark 1.8 (About inner product and Sylvester equations) *Being given* $\mathbf{H}$ *and* $\hat{\mathbf{H}}$ *(and their realisation* $\mathcal{S}$ *and* $\hat{\mathcal{S}}$ *as in* (1.6)*), one can compute the* $\mathcal{H}_2$ *inner product of the two models through the use of the so-called cross-gramians, as follows*

$$\langle \mathbf{H}, \hat{\mathbf{H}} \rangle^2_{\mathcal{H}_2} = \mathbf{tr}(B^T\mathcal{Y}\hat{B}) = \mathbf{tr}(C\mathcal{X}\hat{C}^T) = \mathbf{tr}(C^T\mathcal{Z}\hat{B}),$$

where $\mathcal{X}$*,* $\mathcal{Y}$ *and* $\mathcal{Z}$ *are the solution of the Sylvester equations,*

$$A\mathcal{X}\hat{E}^T + E\mathcal{X}\hat{A}^T + B\hat{B}^T = 0 \ , \ A^T\mathcal{Y}\hat{E} + E^T\mathcal{Y}\hat{A} + C^T\hat{C} = 0 \ \text{ and } \ A\mathcal{Z}\hat{E}^T + E\mathcal{Z}\hat{A}^T + B\hat{C}^T = 0.$$

This last inner product is of importance in the context of $\mathcal{H}_2$ *model approximation, as detailed in Chapters 2 and 4.*

The spectral-based method

Moreover, if $\mathcal{S}$ has simple poles λ_i ($i = 1, \ldots, n$), the $\mathcal{H}_2$-norm can also be computed as (Antoulas, 2005, Chapter 5)

$$||\mathbf{H}||^2_{\mathcal{H}_2} = \frac{1}{2\pi}\oint_\Gamma \mathbf{tr}\left(\overline{\mathbf{H}(\imath\nu)}\mathbf{H}^T(\imath\nu)\right)d\nu = \sum_{i=1}^{n}\mathbf{tr}\left(\Phi_i\mathbf{H}^T(-\lambda_i)\right) = \sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{tr}\left(\frac{\Phi_i\Phi_j^T}{\lambda_i + \lambda_j}\right), \quad (1.15)$$

where λ_i, Φ_i are the poles and residues of $\mathbf{H}$, respectively, and Γ denotes the integral contour of all stable poles of $\mathbf{H}$. To the authors' knowledge, within the control community, this residue-based formulation is mainly used in $\mathcal{H}_2$ optimal model reduction since it allows one to derive first-order optimality conditions directly in term of interpolation conditions. Indeed, this formulation is largely used later in this manuscript in Chapter 2, 3 and 4, as well as in Van Dooren et al. (2008) and Gugercin et al. (2008). Obviously, here, the norm computation is recast as an eigenvalues one. From a pure computational point of view, both gramian and spectral-based methods are of the same complexity. Nevertheless, depending on the framework (control, model approximation, *etc.*), one may be preferred with respect to the other.

1.3.6 ($\mathcal{L}_{2,\Omega}$ and) $\mathcal{H}_{2,\Omega}$-norm computation

Similarly to the $\mathcal{H}_2$-norm, it might be interesting computing the norm over a limited frequency range only. This is achieved by the $\mathcal{H}_{2,\Omega}$-norm, where $\Omega = [0, \omega]$, $\omega > 0$. As in the infinite case, both gramian (Lyapunov) and spectral formulation exist (see also Petersson (2013); Garulli et al. (2013); Petersson and Lofberg (2014) for applications).

The gramian-based method

As rooted on the frequency-limited Lyapunov equations, for a stable and strictly proper system $\mathcal{S} \in \mathcal{H}_2$, the resulting frequency-limited $\mathcal{H}_2$-norm can be defined as

$$||\mathbf{H}||^2_{\mathcal{H}_{2,\Omega}} = \mathbf{tr}(B^T \mathcal{Q}_\Omega B) = \mathbf{tr}(C \mathcal{P}_\Omega C^T),$$

where $\mathcal{P}_\Omega$ and $\mathcal{Q}_\Omega$ are the frequency-limited reachability and observability gramians given in (1.11) (see also Gawronski (2004) for details).

The spectral-based method

As in the infinite case, a new expression of the $\mathcal{H}_{2,\Omega}$-norm based on the spectral information of the system, *i.e.* the residues and eigenvalues and of the transfer function $\mathbf{H}(s)$, has been proposed in Vuillemin et al. (2013a, 2014b) and extended for model approximation purpose in Vuillemin et al. (2019). Before detailing this result, let us note that this norm is actually defined for finite order models $\mathbf{H} \in \mathcal{L}_\infty$, thus, the results are first given for this class of functions, and then reduced to the specific case of models $\mathbf{H} \in \mathcal{H}_\infty$. First, let us defining the complex inverse tangent function $\mathbf{atan}(z)$ as suggested in Kahan (1987), which states that for $z \in \mathbb{C} \setminus \{\pm\imath\}$

$$\mathbf{atan}(z) \triangleq \frac{1}{2\imath} \left(\mathbf{log}(1 + \imath z) - \mathbf{log}(1 - \imath z) \right),$$

where $\mathbf{log}(z) \triangleq \mathbf{ln}(|z|) + \imath \mathbf{arg}(z)$ is the principal value of the logarithm of z, defined for $z \neq 0$ with $-\pi < \mathbf{arg}(z) \leq \pi$. Besides, to simplify the notations as it will be a recurrent term in Chapter 3, let us define the function $a_\omega(s)$ as

$$a_\omega(s) \triangleq \frac{2}{\pi} \mathbf{atan}\left(\frac{\omega}{s}\right).$$

As rooted on the arctangent definition, let us now formulate of the $\mathcal{L}_{2,\Omega}$-norm as in Theorem 1.5.

Theorem 1.5: Spectral expression of the $\mathcal{L}_{2,\Omega}$-norm

Given a continuous **MIMO LTI** dynamical system $\mathbf{H} \in \mathcal{L}_\infty$ and equipped with a realization $\mathcal{S}$ of degree n with simple poles λ_i and $\Phi_i \in \mathbb{C}^{n_y \times n_u}$ the corresponding residues of $\mathbf{H}(s)$ at λ_i, for $i = 1, \ldots, n$. Suppose that the purely imaginary poles λ_k^{im}, $k = 1, \ldots, n_{im}$ ($0 \leq n_{im} \leq n$) of $\mathbf{H}(s)$ are such that $\omega < min\left(|\lambda_k^{im}|\right)$. Then, the frequency-limited $\mathcal{L}_2$-norm can be written as

$$||\mathbf{H}||^2_{\mathcal{L}_{2,\Omega}} \triangleq \sum_{i=1}^{n} \sum_{j=1}^{n} a_{i,j} + \frac{\omega}{\pi} \mathbf{tr}(DD^T) - \frac{2}{\pi} \sum_{i=1}^{n} \mathbf{tr}(\Phi_i D^T) \mathbf{atan}\left(\frac{\omega}{\lambda_i}\right) \tag{1.16}$$

where

$$a_{i,j} \triangleq \begin{cases} \frac{2}{\pi} \mathbf{tr}\left(\frac{\Phi_i \Phi_j^T}{\lambda_i + \lambda_j}\right) \mathbf{atan}\left(\frac{\omega}{\lambda_i}\right) & \text{if } \lambda_i + \lambda_k \neq 0 \\ -\frac{1}{\pi} \mathbf{tr}\left(\frac{\omega \Phi_i \Phi_j^T}{\omega^2 + \lambda_i \lambda_i}\right) & \text{otherwise.} \end{cases}$$

In the above formulation, the system belongs to $\mathcal{L}_\infty$. If instead, one considers the case where $\mathbf{H} \in \mathcal{H}_\infty$, relation (1.16) becomes:

$$||\mathbf{H}||^2_{\mathcal{H}_{2,\Omega}} = \sum_{i=1}^{n} -\frac{2}{\pi} \mathbf{tr}\left(\Phi_i \mathbf{H}(-\lambda_i)^T\right) \mathbf{atan}\left(\frac{\omega}{\lambda_i}\right) + \frac{\omega}{\pi} \mathbf{tr}\left(DD^T\right). \tag{1.17}$$

and if $\mathbf{H} \in \mathcal{H}_2$, then (1.17) becomes

$$||\mathbf{H}||^2_{\mathcal{H}_{2,\Omega}} = \frac{2}{\pi} \sum_{i=1}^{n} -\mathbf{tr}\left(\Phi_i \mathbf{H}(-\lambda_i)^T\right) \mathbf{atan}\left(\frac{\omega}{\lambda_i}\right). \tag{1.18}$$

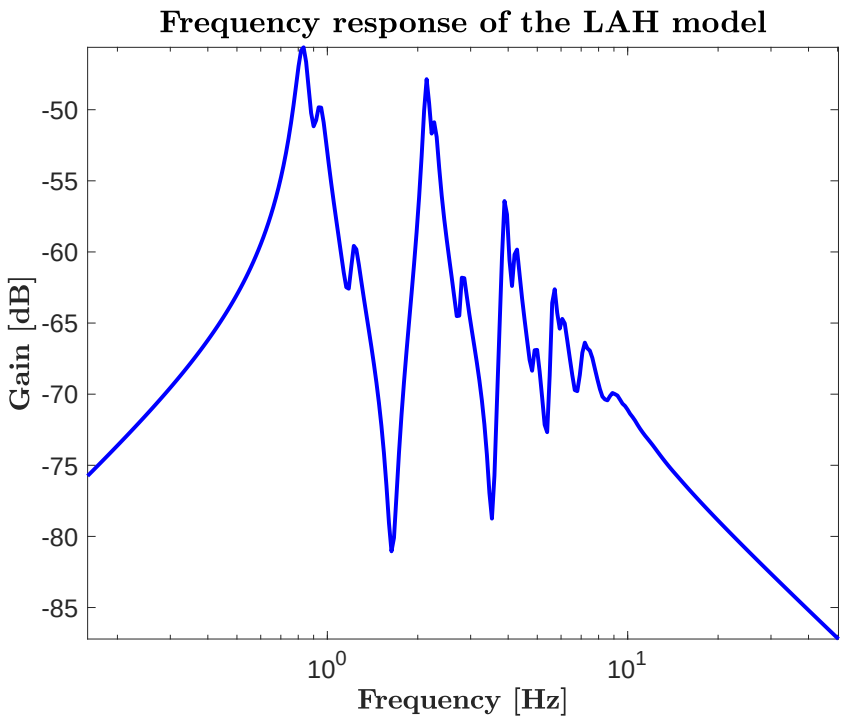
By comparing (1.18) with (1.15), its is interesting noticing that the main difference comes from the $(2/\pi)\mathbf{atan}(\omega/\lambda_i)$ term, playing the role of weight on each modal contribution. This new simple, yet efficient formulation, allows new perspectives from a control, observer, filter design or model reduction point of view and provides an alternative to the gramian-based approach for the frequency limited $\mathcal{H}_2$-norm computation of large-scale models.

Example 5 - Illustration of the $\mathcal{H}_2$ and $\mathcal{H}_{2,\Omega}$-norm behaviour

To illustrate the $\mathcal{H}_{2,\Omega}$-norm, one evaluates it over the range $\Omega = [0, \omega]$, where $\omega \in \mathbb{R}_+$, as a function of the ω value, let us consider the example of the Los Angeles Hospital from the COMPl$_e$ib Leibfritz (2003) which frequency response is given in Figure 1.9 (up). The following code is executed and the resulting frequency-limited $\mathcal{H}_2$-norm is shown on Figure 1.9 (bottom).

```
clear all, close all, clc
%% Load COMPleib model and construct the state space model
name = 'LAH';
[A,B1,B,C1,C,D11,D12,D21,nx,nw,nu,nz,ny] = COMPleib(name);
sys = ss(A,B,C,0);
%% H2 norm computation
H2  = norm(sys,2);
%% H2w norm
wSpace = logspace(0,2.5,300);
for i = 1:numel(wSpace)
    H2w(i) = mor.norm(sys,2,[0 wSpace(i)],[]);
end
```

Listing 1.3: `demo_Chap1_H2norm` script: illustration of the $\mathcal{H}_{2,\Omega}$-norm value as a function of the upper bound $\omega \in \mathbb{R}_+$.



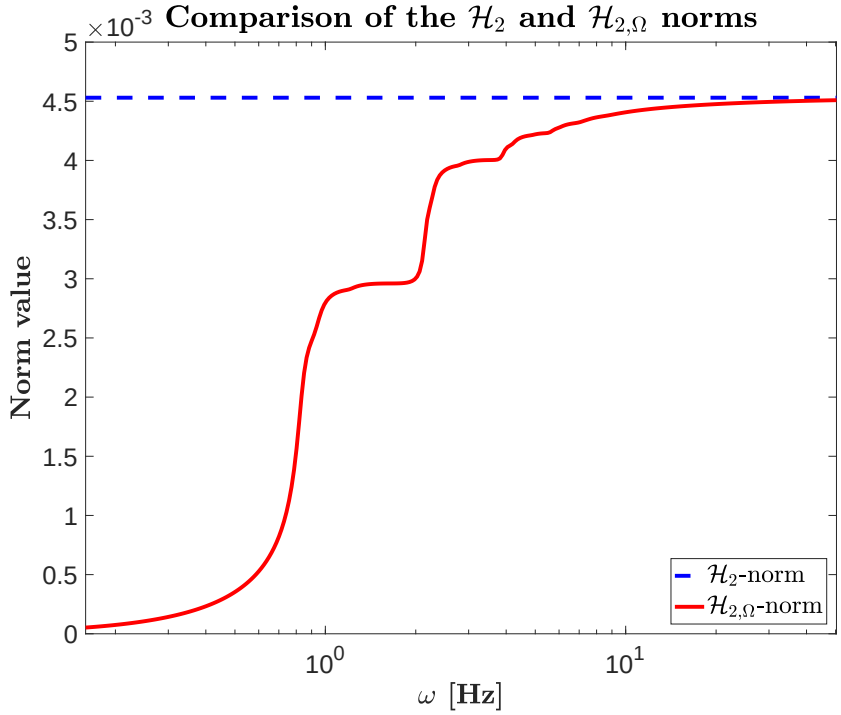


Figure 1.9: Top: frequency response of the LAH model. Bottom: ω upper bound frequency-dependent $\mathcal{H}_{2,\Omega}$-norm.

With reference to Figure (1.9), one should first note that when $\omega \to \infty$, the $\mathcal{H}_{2,\Omega}$-norm tends to the $\mathcal{H}_2$ one. In addition, as an interesting property, at each pic of the frequency response, the $\mathcal{H}_{2,\Omega}$-norm strongly increases. Moreover, the higher the pic is, the higher the derivative of the $\mathcal{H}_{2,\Omega}$-norm is.

Due to the analytic solutions provided for the $\mathcal{H}_2$ and $\mathcal{H}_{2,\Omega}$-norms, these latter are clearly the most largely used in the model reduction field. Yet, in the control design and analysis fields, due to its nice robustness properties, the $\mathcal{H}_\infty$ one is still largely used in many academic researches, as well as industrial applications including automotive, aerospace, biology *etc.* (see *e.g.* Zhou and Doyle (1997); Apkarian and Noll (2006); Burke et al. (2006)). This norm computational aspects is briefly discussed now.

1.3.7 $\mathcal{H}_\infty$-norm computation

The $\mathcal{H}_\infty$-norm of transfer $\mathbf{H} \in \mathcal{H}_\infty$ is given as in (1.3), and recalled as

$$||\mathbf{H}||_{\mathcal{H}_\infty} = \max_{\omega \in \mathbb{R}} \sigma\left(\mathbf{H}(\imath\omega)\right) = \sup_{\omega \in \mathbb{R}} \overline{\sigma}\left(\mathbf{H}(\imath\omega)\right) = \max_{w \in L_2} \frac{||z||_2}{||w||_2},$$

where $\sigma(\cdot)$ and $\overline{\sigma}(\cdot)$ are the singular and maximal singular value operators, respectively. The computation of this norm still is a quite active field where dedicated eigen-solvers are constructed (see *e.g.* Benner and Mitchell (2017) for recent advances and relevant references).

The bisection procedure

To the author best of knowledge, the main model realisation-based method allowing to compute this norm, is based on the eigenvalues problem applied on the Hamiltonian matrix $\mathcal{H}$. For a given $\gamma \in \mathbb{R}_+$ (with $E = I_n$ and $D = 0$), one defines

$$\mathcal{H}(A, B, C, \gamma) = \begin{bmatrix} A & \gamma BB^T \\ -\gamma C^T C & -A^T \end{bmatrix}$$

which should not have any eigenvalue along the imaginary axis. In practice, this is obtained by using a bisection procedure, as suggested in Boyd et al. (1988). The MATLAB software embeds the function **`norm`** which allows the $\mathcal{H}_\infty$-norm computation for dense models. Practitioners might also note also that the MATLAB function **`norm(H,inf)`** applied on an unstable model **`H`**, gives the $\mathcal{L}_\infty$-norm (see also Boyd et al., 1988; Boyd and Balakrishnan, 1989; Gahinet and Apkarian, 1992).

$\mathcal{H}_{2,\Omega}$-based $\mathcal{H}_\infty$-norm approximation and inequalities

As the computational cost of the $\mathcal{H}_{2,\Omega}$-norm is much lower than the $\mathcal{H}_\infty$ one, Vuillemin et al. (2014c) did exploit it to approximate the $\mathcal{H}_\infty$ worst-case pulsations values without computing the $\mathcal{H}_\infty$-norm itself. In addition, in the same paper, Vuillemin et al. (2014c) did compute two upper bounds of the $\mathcal{H}_\infty$-norm. These latter offer a limited computational cost, and can then be applied to large-scale models. These two bounds are recalled in what follows.

Theorem 1.6: Upper bounds of the $\mathcal{H}_\infty$-norm

Given $\mathbf{H} \in \mathcal{H}_\infty$ and $\omega \in \mathbb{R}_+$, its $\mathcal{H}_\infty$-norm is upper bounded as

$$||\mathbf{H}||_{\mathcal{H}_\infty} \leq \max_{\omega \in \mathbb{R}_+} ||\mathbf{H}(\imath\omega)||_F = \max_{\omega \in \mathbb{R}} \sqrt{\pi \frac{\partial ||\mathbf{H}||_{\mathcal{H}_2,[0\ \omega]}}{\partial \omega}}.$$

Then, based on the poles-residues expression of the $\mathcal{H}_{2,\Omega}$-norm given in (1.18), the following holds true:

$$\frac{\partial ||\mathbf{H}||_{\mathcal{H}_2,[0\ \omega]}}{\partial \omega} = -\frac{2}{\pi} \sum_{i=1}^{n} \mathbf{tr}\left(\Phi_i \mathbf{H}(-\lambda_i)^T\right) \frac{\lambda_i}{\lambda_i^2 + \omega^2} = \sum_{i=1}^{n} f_i(\omega).$$

By denoting $g_i(\omega) = \Re(f_i(\omega))$, it comes that

$$||\mathbf{H}||_{\mathcal{H}_\infty} \leq \underbrace{\left(\max_{\omega \in \mathbb{R}} \pi \sum_{i=1}^{n} g_i(\omega)\right)^{\frac{1}{2}}}_{\Gamma(\mathbf{H})} \leq \underbrace{\left(\pi \sum_{i=1}^{n} \max_{\omega \in \mathbb{R}} g_i(\omega)\right)^{\frac{1}{2}}}_{\bar{\Gamma}(\mathbf{H})}.$$

Interestingly, in the **SISO** case, $\Gamma(\mathbf{H}) = ||\mathbf{H}||_{\mathcal{H}_\infty}$ and in the **MIMO** case, one can expect the bound to be more or less conservative depending on the singular values of the transfer function. Moreover, as in the $\mathcal{H}_{2,\Omega}$ case, one can also bound the maximum singular value over some frequency interval Ω, by restricting the search domain, *i.e.*

$$\max_{\omega \in \Omega} \sigma_{max}(\mathbf{H}) \leq \underbrace{\left(\max_{\omega \in \Omega} \pi \sum_{i=1}^{n} g_i(\omega)\right)^{\frac{1}{2}}}_{\Gamma_\Omega(\mathbf{H})} \leq \underbrace{\left(\pi \sum_{i=1}^{n} \max_{\omega \in \Omega} g_i(\omega)\right)^{\frac{1}{2}}}_{\bar{\Gamma}_\Omega(\mathbf{H})}$$

Computing $\bar{\Gamma}(\mathbf{H})$ (or $\bar{\Gamma}_\Omega(\mathbf{H})$) requires to find the maximums of n simple rational functions, which can be achieved analytically. Computing $\Gamma(\mathbf{H})$ (or $\Gamma_\Omega(\mathbf{H})$) consists in finding the maximum of a sum of rational functions of ω, which requires an optimisation procedure. See Vuillemin et al. (2014c) for details on the computation and statistical examples.

1.4 Conclusions

In this chapter, a brief overview of the linear system spaces, norms and representations, has been done. Starting from the quite general Hardy spaces definitions and properties, the "control-oriented" $\mathcal{L}_2$, $\mathcal{H}_2$, $\mathcal{L}_\infty$ and $\mathcal{H}_\infty$ spaces were detailed. Then, based on the description of the complex-valued meromorphic functions belonging to these spaces, their associated norms were defined in a general manner. The specific case of rational complex valued functions, which is of specific interest in the control community, was emphasised together with the (stable) realisation representation and the pole residue one. Finally, discussions on linear algebra operations and norms computational aspects were drawn. Indeed, when dealing with large-scale systems, these issues become crucial in the control engineering practice as they can render a theoretical result inoperable in practice if not numerically carefully handled. Moreover, author truly believes that these considerations should always be kept in mind even for operations and small-scale systems. Obviously, this chapter lacks from many details, but still provides a quick overview accompanied by some very complete references, and specifies the basic definitions employed in the rest of the manuscript.

By keeping in mind the elements provided in this chapter, the next chapter will try to give a picture of the model approximation methods, as treated in this work. More than a collection of mathematical theorems and lemmas, in the rest of the manuscript will try to illustrate though examples and use cases, the theoretical and numerical difficulties (and sometime solutions), and results within model reduction and approximation, as well as some related applications.

Chapter 2

Introduction to linear large-scale dynamical model approximation

View it, code it, jam - unlock it,
Surf it, scroll it, pause it, click it,
Cross it, crack it, switch - update it, ...

Daft Punk (Technologic)

Contents

After the first chapter dedicated to preliminary definitions on linear dynamical systems, this chapter, without being exhaustive, provides a brief overview of the linear large-scale model approximation problem, framework and main results. For a more complete overview, reader is invited to refer to the book of Antoulas (2005) and survey papers of Benner et al. (2015); Benner and Stykel (2016). Section 2.1 both provides the context and motivations for model approximation as well as a classification of the linear approximation problems as treated in this work. Then, Sections 2.2 and 2.3 define, similarly to an optimisation problem, the approximation criteria and parametrisation of the solution set. Finally, Section 2.4 gives a lecture grid (as well as a brief overview) of the approximation methods in the linear framework (most of the details will be omitted for brevity and let to the reader's curiosity through relevant references). Conclusions are given in Section 2.5 which closes this chapter.

2.1 Motivations and context

Due to the ever increasing need to enhance system's performances, reliability and safety while reducing development time, costs and experimental phases, dedicated numerical softwares and digital-based tools are being more and more used to accurately catch and restitute any physical phenomena. These softwares (generic or dedicated to a specific application) generally embed mathematical dynamical models. For many practitioners, these numerical models and softwares are the starting point to simulate, optimise, control, analyse and more generally, to understand different phenomena and systems. Depending on the nature and desired accuracy of the physical system or phenomena to represent, or the approach used to construct such a model, this latter can become more or less complex and representative of the reality.

Obviously, very accurate models are usually preferred. However, in practice, such accuracy is accompanied with a high complexity, making models manipulation complicated and sometimes inappropriate for engineers. Indeed, due to computers limited computational burden, storage capacity and floating point arithmetic, simulation and optimisation of these complex models might become un-tractable or ineffective and disturb theoretical results. In addition, as the complexity growth (*e.g.* number or nature of the equations), the time cost to obtain a simulation and/or optimisation result might growth, which is a slowing factor when developing digital-based solutions, where simulation plays a central role.

As a mater of consequence, it seems appropriate to restrain the complexity of dynamical models before deploying the numerical bundle of algorithm exploiting them. This complexity reduction should probably be performed during the construction of the model. Nevertheless, this is not always possible nor suitable for many theoretical (*e.g.* methodology ensuring optimality is missing) or practical reasons (*e.g.* physicians construct models that will be used for different purpose, and the multiplication of models would lead to a waste of time). Indeed, on one hand, numerical modelling tools (such as identification, finite elements, ...) do not enable to limit the complexity without loosing informations. On the other hand, dynamical models are usually constructed for different purpose (*e.g.* simulation, control, analysis, inverse problem ...), and their complexity usually vary in consequence. In practice and in many industrial applications, it is preferred to construct a single high fidelity model and to simplify it afterward depending on the application and utilisation. Seeking for efficient methods allowing constructing a simpler low complexity dynamical model tailored to its utilisation, that well restitutes the original input-output model transfer and its main characteristics, is the purpose of model approximation.

In this work, dynamical models approximation is mainly considered. More specifically, among the different dynamical model representations, the linear ones as described in Chapter 1, are considered only. The approximated or reduced order models considered in this work are obtained from the following different type of inputs that be classified in three folds:

(#1) the linear first order state-space models $\mathcal{S}$, defined by a set of algebraic-differential equations (realisation-based framework),

(#2) the complex-valued meromorphic functions $\mathbf{H}$ (realisation-free framework) and

(#3) the frequency-domain discrete input-output data set $\{\omega_i, \mathbf{\Phi}_i\}_{i=1}^{N}$ (data-driven framework).

For each cases, the common objective is to find a reduced-order linear dynamical model. These three cases are illustrated on Figure 2.1, and detailed and illustrated in what follows:

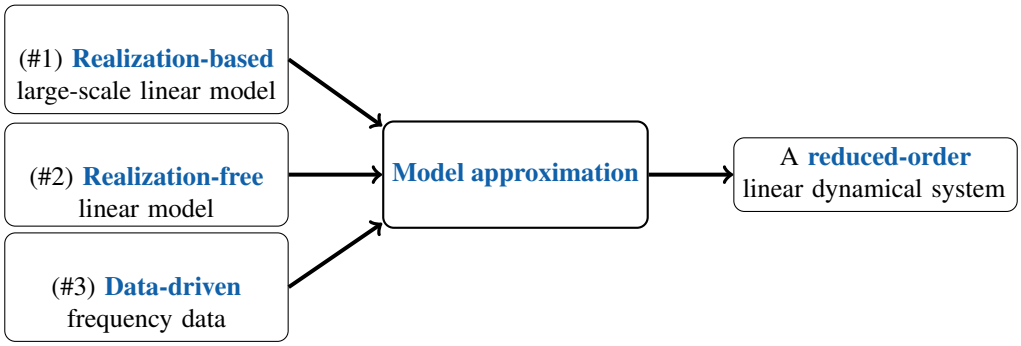


Figure 2.1: Three problem classes treated in this manuscript.

(#1) Realisation-based framework

This framework includes the linear n_u inputs n_y outputs finite n-th order (typically, $n \approx 10 \dots 10^9$) state-space models governed by a set of differential-algebraic equations $\mathcal{S} : (E, A, B, C, D)$, defined as

$$E\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) \text{ and } \mathbf{y}(t) = C\mathbf{x}(t) + D\mathbf{u}(t),$$

where $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$, $C \in \mathbb{R}^{n_y\times n}$ and $D \in \mathbb{R}^{n_y\times n_u}$. Since in the control community many tools are based on this model class, it is one of the most largely studied. Models of this form typically result from discretisation of **PDE** or from linearisation of physical equations. In this configuration, the model approximation goal - generally - consists in constructing $\hat{\mathbf{H}}$, a linear reduced-order models equipped with a realisation of the form $\hat{\mathcal{S}} : (\hat{E}, \hat{A}, \hat{B}, \hat{C}, \hat{D})$ and a reduced state-space vector $\hat{\mathbf{x}}(t) \in \mathbb{R}^r$ of dimension $r \ll n$, which catches the original input-output behaviour and preserves some properties of the original model. Obviously, as pointed along the chapter, approximation involving other realisation structures can be used, but these latter will be marginally treated in this manuscript.

Example 6 - A finite element model approximation

This example comes from the discretisation of a Timoshenko clamped beam described in a technical note by Panzer et al. (2009), providing a fast and easy way to produce an **ODE LTI** model **H** with a realisation $\mathcal{S}$, of a 3D cantilever Timoshenko beam using finite element approximation, with a user-defined discretisation N step.
To illustrate the benefit of model approximation, we construct an original accurate model by discretising the beam with $N = 75$, leading to Model 1 with dimensions $n = 900$, $n_u = 1$ (vertical acceleration at the beam extremity), $n_y = 150$ (vertical displacements along the beam), then with $N = 2$, leading to Model 2 with dimensions $n = 24$, $n_u = 1$, $n_y = 4$. Then, Model 1 is approximated with an order $r = 24$, $n_u = 1$, $n_y = 150$, using the **MOR** Toolbox, leading to Model 3. Figures 2.2-2.3 illustrates the results.

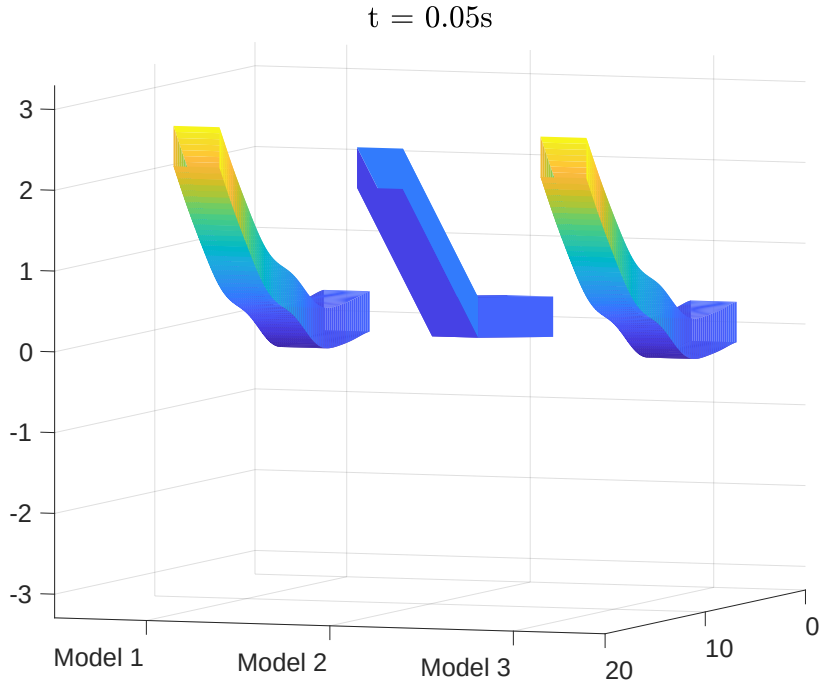


Figure 2.2: Snapshot of the impulse response of the Timoshenko beam. 3D representation of the beam Models 1, 2 and 3 at $t = 0.05$.

Clearly, one obtains better simulation results by performing an accurate model followed by a model

approximation, rather than a low dimensional model directly. Indeed, the integral error between the original Model 1 and Model 2, and Model 1 and Model 3, in response to an impulse at the extremity leads to 5.2107 for Model 2 and 0.088015 with Model 3. Moreover the simulation computational time for one second simulation is of 0.2967s for Model 1 and 0.0015s for Model 2 and Model 3. Finally, applying the reduction with the MATLAB Toolbox leads to an $\mathcal{H}_2$ mismatch error of 1.630411×10^{-3} and a solution obtained in 7.3424s, while the **MOR** Toolbox reaches a mismatch of 1.200435×10^{-3} in 6.1899s, only.

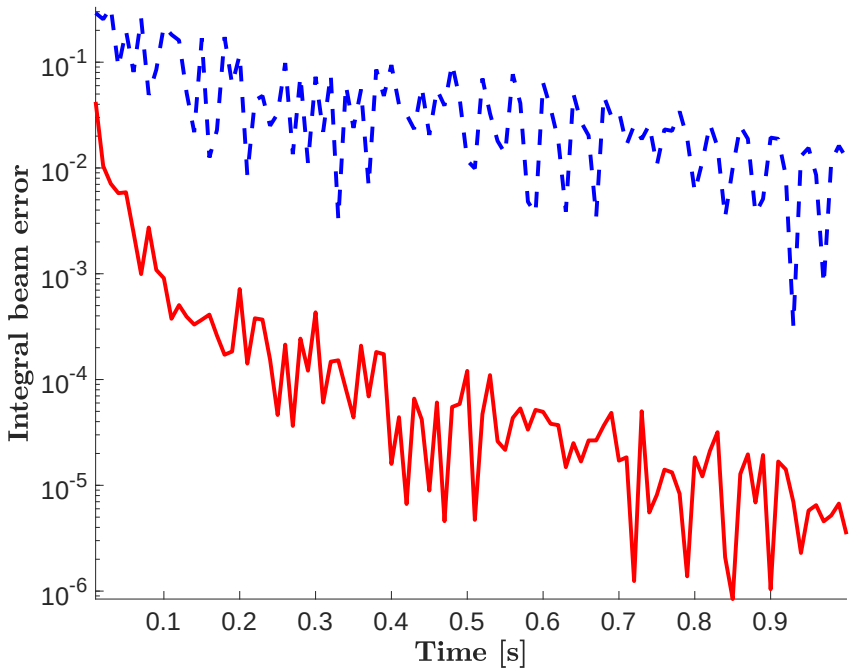


Figure 2.3: Impulse response of the Timoshenko beam. Error at the extremity of the beam between Model 1 and Model 2 (dashed blue) / Model 3 (solid red), as a function of time.

(#2) Realisation-free framework

This framework concerns any n_u inputs n_y outputs complex-valued meromorphic functions, not necessarily rational, described as

$$\mathbf{y}(s) = \mathbf{H}(s)\mathbf{u}(s),$$

where $\mathbf{H} \in \mathcal{H}_2$ or $\mathcal{H}_\infty$. Such a model can be obtained after Laplace transform of linear partial differential equations (see *e.g.* Curtain and Morris (2010) for examples). As in case #1, in this configuration, the objective is to approximate the complex-valued meromorphic function by $\hat{\mathbf{H}}$ a finite order linear model, usually equipped with a realisation $\hat{\mathcal{S}}$ with a state-space vector $\hat{\mathbf{x}}(t)$ of dimension r.

The main difference with the previous framework #1 stands is the fact that here, neither internal representation nor state-space vector is a-priori known. Therefore, we clearly talk here of model approximation rather than model reduction.

Example 7 - Linear PDE solution approximation

When linear **PDE** with constant coefficients are considered, the Laplace transform can be used to solve them (sometimes leading to irrational transfer functions). Let us be given the following **PDE**

$$\begin{aligned} \frac{\partial \tilde{\mathbf{y}}(x,t)}{\partial x} + 2x\frac{\partial \tilde{\mathbf{y}}(x,t)}{\partial t} &= 0 \\ \tilde{\mathbf{y}}(x,0) &= 0 \\ \tilde{\mathbf{y}}(0,t) &= \frac{1}{\sqrt{t}} * \tilde{\mathbf{u}}_f(0,t) \\ \frac{\omega_0^2}{s^2 + m\omega_0 s + \omega_0^2}\mathbf{u}(0,s) &= \mathbf{u}_f(0,s), \end{aligned}$$

where $x \in [0\ L]$, $L = 3$ is the space variable and $\omega_0 = 3$ and $m = 0.5$ are the input filter parameters. The scalar input of the model is $\tilde{\mathbf{u}}(0,t)$ (or $\mathbf{u}(0,s)$ in the Laplace domain), the vertical force applying at the left boundary. By applying the Laplace transform, one obtains

$$\frac{\partial \mathbf{y}(x,s)}{\partial x} + 2x\left(s\mathbf{y}(x,s) - \tilde{\mathbf{y}}(x,0)\right) = 0,$$

which solution can be given as $\mathbf{y}(x,s) = a(s)e^{\int -2xsdx} = a(s)e^{-x^2 s}$. Due to boundary condition $\tilde{\mathbf{y}}(0,t) = \frac{1}{\sqrt{t}} * \tilde{\mathbf{u}}_f(t)$, we have $\mathbf{y}(0,s) = \frac{\sqrt{\pi}}{\sqrt{s}}\mathbf{u}_f(s)$, and consequently $a(s) = \frac{\sqrt{\pi}}{\sqrt{s}}\mathbf{u}_f(s)$. Therefore,

$$\mathbf{y}(x,s) = \frac{\sqrt{\pi}}{\sqrt{s}}e^{-x^2 s}\frac{\omega_0^2}{s^2 + m\omega_0 s + \omega_0^2}\mathbf{u}(s) = \mathbf{H}(x,s)\mathbf{u}(s),$$

which links the input to the output through an irrational transfer function $\mathbf{H}$. Figure 2.4 then illustrates the irrational model frequency response at point $x = L$, and Figure 2.5, the time responses along x for the exact and rational approximation (note that for this toy example, the exact time-domain solution is given by $\tilde{\mathbf{y}}(x,t) = \tilde{\mathbf{u}}_f(t - x^2)/\sqrt{t}$).

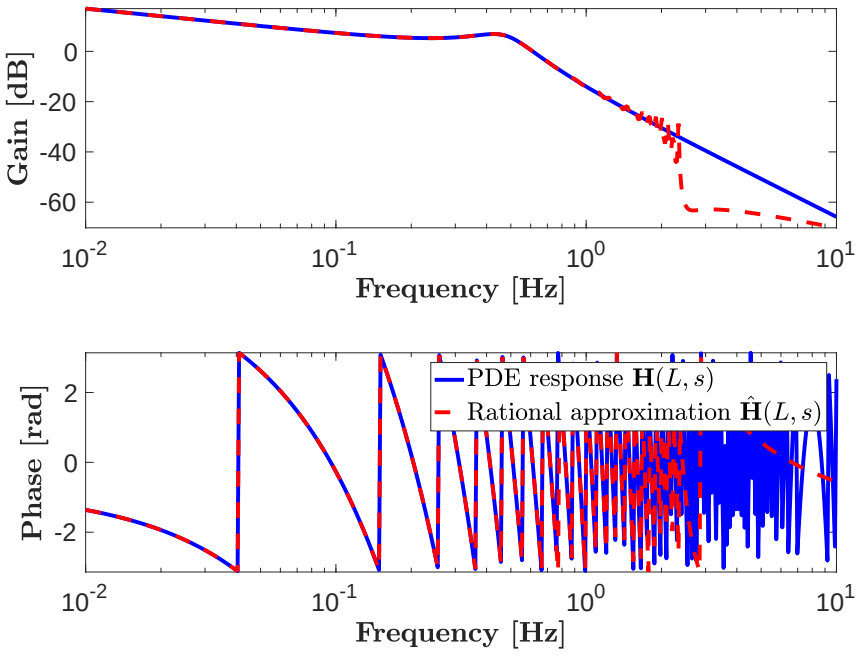


Figure 2.4: Frequency response of the **PDE** at $x = L$ and its rational approximation with $r = 50$.

Interestingly, the rational approximation model well restitutes the response, and even in a more numerically accurate way. Indeed, due to the singularity of the exact solution, some unexpected values occurs. Then, based on the rational model $\hat{\mathbf{H}}$ equipped with $\hat{\mathcal{S}}$, it is now easy to design a linear control law to damp and stabilise the system with any standard tools. This is illustrated on Figure 2.5, with a snapshot of the impulse response. With reference to Figure 2.5, where the considered measure is materialised by the vertical line, the system is looped at the left boundary with a PI controller given by this realisation $\dot{\mathbf{x}}_c = \mathbf{y}(x_{meas}, t) - r(t)$ and $\mathbf{u}(t) = 0.0283\mathbf{x}_c + 0.2284(\tilde{\mathbf{y}}(x_{meas}, t) - r(t))$, where $r(t)$ is the reference signal. The control scheme is not described, but simply consists in a tracking performance objective[a]. The controller gains are obtained through $\mathcal{H}_\infty$ norm minimisation using the **`hinfstruct`** function from Apkarian and Noll (2006) (see also Gahinet and Apkarian (2011); P. Gahinet (2013)).

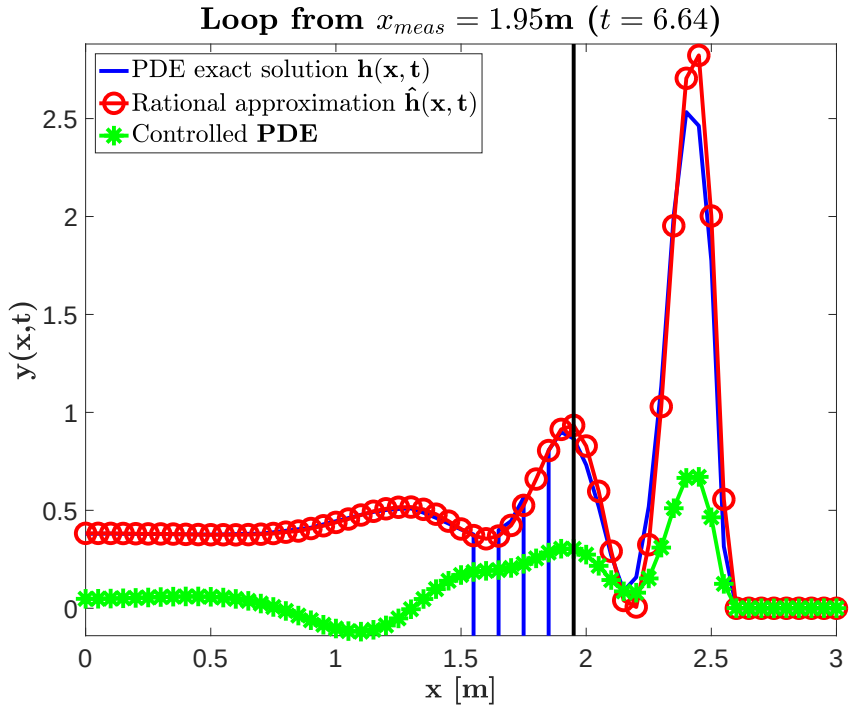


Figure 2.5: Snapshot of impulse response of the uncontrolled (rounded red line and blue solid line) and controlled (stared green line) at $t = 6.6$s, when looped with the PI control designed based on the rational finite order model $\hat{\mathbf{H}}$.

[a] Details are given in the attached script files.

(#3) Data-driven framework

Within this framework, instead of an analytical description of the model such as the descriptor form of *(#1)* or any meromorphic transfer function form *(#2)*, we are given a set of frequency-domain (or complex-domain) n_u inputs n_y outputs data obtained from experimental measurements or from any numerical simulation, as

$$\mathbf{y}(s_i) = \mathbf{\Phi}_i \mathbf{u}(s_i),$$

where $\{s_i\}_{i=1}^N \in \mathbb{C}$ and $\{\mathbf{\Phi}_i\}_{i=1}^N \in \mathbb{C}^{n_y \times n_u}$. In this configuration, at the boundaries with the traditional model identification problem (see overview paper of Ljung (2013)), the objective is to find $\hat{\mathbf{H}}$ a model usually equipped with a realisation $\hat{\mathcal{S}}$ with a given complexity, that well reproduces the data. Note that when the data $\{\mathbf{\Phi}_i\}_{i=1}^N$ are collected from experimental test, usually one have $s_i = \imath\omega_i$, where $\omega_i \in \mathbb{R}_+$ is the pulsation of the experiment.

Example 8 - Aero-structural aircraft model approximation

This use-case is extracted from the MOR Wiki webpage[a] and represents a flexible aircraft model in response to discrete gust disturbances. Its primal description is detailed in Quero (2017) and its formalisation for the data-driven model approximation, described in Poussot-Vassal et al. (2018). Such a model is used by aero-structural engineers to understand the physics of the aircraft and to perform simulations for authorities in order to assess gust load alleviation functions (see also Moulin and Karpel (2007); Wang and Chen (2017)).
Usually, the simulator allowing to calculate the response to gust disturbances requires many time to compute the output responses (*e.g.* effort along the wing). By an adequate data-driven model approximation, this computational time has been divided by hundred, as exposed in Poussot-Vassal et al. (2018). Figure 2.6 illustrates the frequency response of the **GLRA** (Generic Long Range Aircraft) along the right wing span for the exact model and its rational reduced order approximation with $r = 100$.

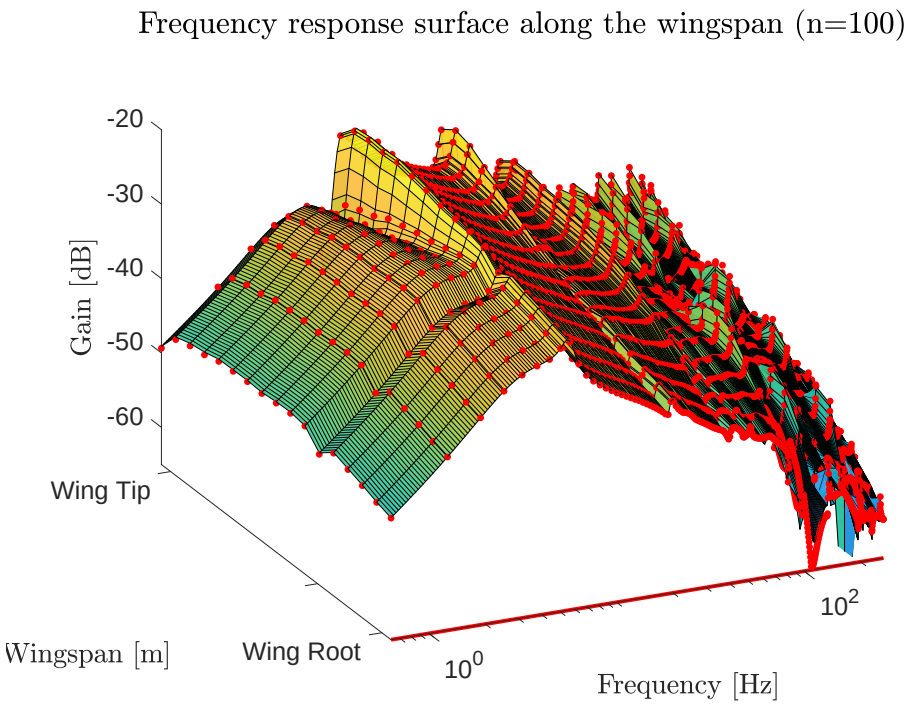


Figure 2.6: Frequency response of the **GLRA** efforts along the right wing. Red dots are the original model responses (from the **HiFi** simulator) and the surface stands for the approximation $\hat{\mathbf{H}}$ with $r = 100$.

As the Loewner framework has been employed, the Loewner matrix $\mathbb{L}$ normalised singular values decay is also shown on Figure 2.7, exhibiting that from $r = 100$, the dynamic is already well caught (see later in this chapter for details).

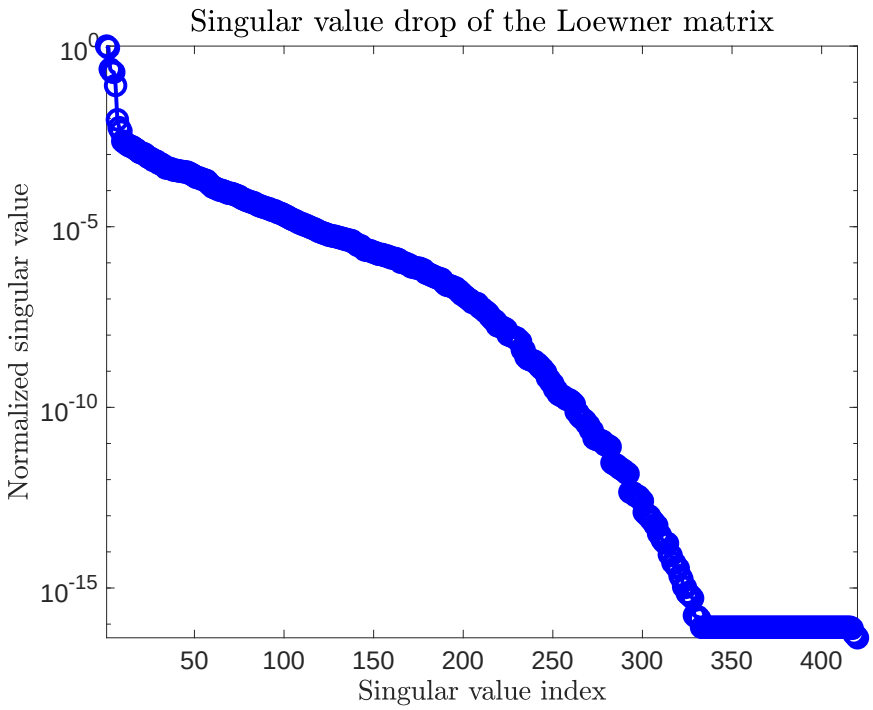


Figure 2.7: Loewner matrix $\mathbb{L}$ singular values decay.

[a] https://morwiki.mpi-magdeburg.mpg.de/morwiki/index.php/Flexible_Aircraft

Now the main use cases considered in this work have been introduced and exemplified, illustrating some of the many model approximation interests, let us now mathematically define the most standard approximation objectives in Section 2.2, and then, some of the most standard parametrisation of the solution in Section 2.3.

2.2 Approximation criteria

Following the three model input types presented above, let us now formulate the approximation objectives (or criteria), which will then determine the methodology and numerical scheme to deploy.

2.2.1 General problem setting and objective

Given the meromorphic complex-valued transfer function matrix $\mathbf{H}$ equipped or not with a realisation, mapping the n_u inputs $\mathbf{u}$ to the n_y outputs $\mathbf{y}$ such that

$$\mathbf{y}(s) = \mathbf{H}(s)\mathbf{u}(s), \tag{2.1}$$

or the input-output data collection $\{s_i, \mathbf{\Phi}_i\}_{i=1}^N = \{s_i, \mathbf{H}(s_i)\}_{i=1}^N$ and with $\mathbf{H}(\cdot)$ unknown, such that

$$\mathbf{y}(s_i) = \mathbf{\Phi}_i\mathbf{u}(s_i), \tag{2.2}$$

the approximation problem aims at producing the approximate meromorphic (rational) transfer function matrix $\hat{\mathbf{H}}$ mapping the n_u inputs $\mathbf{u}$ to the n_y approximate outputs $\hat{\mathbf{y}}$ such that

$$\hat{\mathbf{y}}(s) = \hat{\mathbf{H}}(s)\mathbf{u}(s).$$

Some general goals for reduced order model are:

(i) the reduced inputs to outputs map should be uniformly "close" to the original, *i.e.* for the same $\mathbf{u}$, $\mathbf{y} - \hat{\mathbf{y}}$, should be "small" in an appropriate sense,

(ii) the critical system features and structure should be preserved, *e.g.* stability, passivity, real boundedness, eigenvalues, subsystem interconnectivity, first / second / delayed order structure *etc.* and,

(iii) the strategies for computing the reduced system should lead to robust and numerically stable algorithms, and require minimal user tuning parameters.

Before detailing the main algorithms, reader should note the difference nature between use-case *(#1)-(#2)* and *(#3)*. Indeed, while the two first cases embed a mathematical dynamical model description as (2.1), they belong to the so-called *model-based* category. On the contrary, the latter case, which only provides discrete inputs to outputs informations as in (2.2), belongs to the *data-based* one. Following this classification, both problems formulations and resolutions will be different and objectives will be defined accordingly (even if, as illustrated later on in this chapter, some elegant bridges exist).

2.2.2 Model-based approximation

Let us first describe the model-based model approximation criteria related to *(#1)-(#2)*. The specificity of this class is that a model description as in (2.1) exist and can be exploited. Then, dynamical models norms can enter into the picture. Within this class, one of the most standard and appreciated problem is the so-called $\mathcal{H}_2$ one, recalled hereafter.

Problem 2.1: $\mathcal{H}_2$ model approximation

Given $\mathbf{H} \in \mathcal{H}_2$, the $\mathcal{H}_2$ model approximation problem consists of seeking an approximation $\hat{\mathbf{H}} \in \mathcal{H}_2$ of $\mathbf{H}$, such that

$$\hat{\mathbf{H}} := \arg \min_{\substack{\mathbf{G} \in \mathcal{H}_2 \\ \mathbf{dim}(\mathbf{G}) = r \ll n}} ||\mathbf{H} - \mathbf{G}||_{\mathcal{H}_2}. \tag{2.3}$$

Interestingly, although the $\mathcal{H}_2$-norm is smooth and differentiable, it is worth noticing that (2.3) is non-linear and exhibits multiple local minima, as illustrated in Vuillemin (2014). Keeping in mind that this problem should be solved in a very large-scale context, it is particularly challenging to solve it in practice. However, this problem received many attentions in the literature and first order optimality conditions have been derived through different approaches and for different parametrisation of the solution, *i.e.* different structure of $\mathbf{G}$ (*e.g.* rational, state / inputs / outputs delayed, *etc.*). Later in this chapter, we will come back to the existing solutions but it can already be mentioned that Wilson (1974) did provide the theoretical solution using a state-space form and Lyapunov and Sylvester equations, and Gallivan et al. (2004a) then Gugercin et al. (2008) solve it using a pole-residue form and using the tangential framework and connection with generalised Krylov subspace, embedding a numerically efficient procedure. Reader should remind that the importance of optimising the $\mathcal{H}_2$ mismatch error when performing model approximation can be motivated by connecting the norm of the models mismatch and the signals as follows:

$$||\mathbf{y} - \hat{\mathbf{y}}||_{L_\infty} \leq ||\mathbf{H} - \hat{\mathbf{H}}||_{\mathcal{H}_2} ||\mathbf{u}||_{L_2},$$

This last expression directly links the mismatch error with the output signals. Indeed, minimising the $\mathcal{H}_2$ mismatch directly provides a bound on the worst case output signal error, which is usually expected by the practitioners specifications.

As a direct extension of the $\mathcal{H}_2$ approximation problem, the frequency-limited $\mathcal{H}_2$ one, also called $\mathcal{H}_{2,\Omega}$, being the frequency band restriction of the $\mathcal{H}_2$ one, can be formulated as follows.

Problem 2.2: $\mathcal{H}_{2,\Omega}$ model approximation

Given $\mathbf{H} \in \mathcal{H}_\infty$, the $\mathcal{H}_{2,\Omega}$ model approximation problem consists of seeking an approximation $\hat{\mathbf{H}} \in \mathcal{H}_\infty$ of $\mathbf{H}$, such that

$$\hat{\mathbf{H}} := \arg \min_{\substack{\mathbf{G} \in \mathcal{H}_\infty \\ \mathbf{dim}(\mathbf{G}) = r \ll n}} ||\mathbf{H} - \mathbf{G}||_{\mathcal{H}_{2,\Omega}}. \tag{2.4}$$

Similarly to the $\mathcal{H}_2$ one, (2.4) is nonlinear and multiple local minima exist. This problem will be specifically treated in Chapter 3. However, as for Problem 2.1, Problem 2.2 can be solved by involving Lyapunov and Sylvester equations and using a state-space form as proposed by Petersson and Lofberg (2014) or an interpolatory ones with the pole-residue one, as done by Vuillemin et al. (2013a); Vuillemin (2014); Vuillemin et al. (2019).

Remark 2.1 (Existence and unicity) *With reference to Problems 2.1 and 2.2, one question before entering into methodological or practical considerations, is to consider the existence (and unicity) of a solution, especially when no specific structure of* $\mathbf{H}$ *and* $\hat{\mathbf{H}}$ *are given. This problem clearly is at the frontier of my expertise and is closer to the complex functional analysis. Still, interested reader may refer to the article of Baratchart (1986), in which the existence of the best approximation of a given stable dynamical model* $\mathbf{H}$ *by a finite order one* $\hat{\mathbf{H}}$*, is done (in the discrete-time domain). The paper first proves the existence, and then its unicity (in the case of discrete-time model, this unicity is in fact cyclic).*

As previously mentioned, both $\mathcal{H}_2$ and $\mathcal{H}_{2,\Omega}$ problems are linked to the interpolatory problem which play a pivotal role in many model approximation methods. This latter is exposed hereafter.

Problem 2.3: Model-based interpolation

Given $\mathbf{H}$, left interpolation points $\{\mu_j\}_{j=1}^q \in \mathbb{C}$ with left tangential directions $\{\mathbf{l}_j\}_{j=1}^q \in \mathbb{C}^{n_y}$ and right interpolation points $\{\lambda_i\}_{i=1}^k \in \mathbb{C}$ with right tangential directions $\{\mathbf{r}_i\}_{i=1}^k \in \mathbb{C}^{n_u}$, find a (low order) transfer function $\hat{\mathbf{H}}$ that is a tangential interpolant to $\mathbf{H}$, *i.e.* satisfies the following left and right interpolation conditions (let us assume that μ_j and λ_i are distinct):

$$\left.\begin{array}{c} \mathbf{l}_j^H \hat{\mathbf{H}}(\mu_j) = \mathbf{l}_j^H \mathbf{H}(\mu_j) \\ \text{for } j = 1, \ldots, q \end{array}\right\} \text{ and } \left\{\begin{array}{c} \hat{\mathbf{H}}(\lambda_i)\mathbf{r}_i = \mathbf{H}(\lambda_i)\mathbf{r}_i \\ \text{for } i = 1, \ldots, k \end{array}\right. . \tag{2.5}$$

As made cleared in the rest of this chapter and along this manuscript, this problem will enter into consideration for many models structures and for different approximation objectives.

Obviously, instead of the $\mathcal{H}_2$ and $\mathcal{H}_{2,\Omega}$ norms, other systems-oriented norms might be considered. Among them, the same optimization problems using the $\mathcal{H}_\infty$-norm is explored in Flagg et al. (2013) and involving the ν-gap metrics in Sootla et al. (2009). However, they both suffer for high numerical cost due to the iterative nature of their computation which render complicated deriving analytical solutions, as in the $\mathcal{H}_2$ case. In this manuscript, these metrics are not directly treated, and interested reader is invited to refer to the above mentioned references.

2.2.3 Data-based approximation

By now considering the class of frequency domain data-driven input type *(#3)*, the approximation problem turns to be defined in a different way. Here, continuous complex functions-oriented metrics as presented in Chapter 1 and in the previous section, do not apply anymore. Instead, discrete metrics may be defined before performing any optimisation or interpolation. Thus, the following problems describe two angles of attack: first the interpolatory framework (similar to the model-based one) and then the least square one[1].

Problem 2.4: Data-driven interpolation

Given left interpolation driving frequencies $\{\mu_j\}_{j=1}^{q} \in \mathbb{C}$ with left output or tangential directions $\{\mathbf{l}_j\}_{j=1}^{q} \in \mathbb{C}^{n_y}$, producing the left responses $\{\mathbf{v}_j\}_{j=1}^{q} \in \mathbb{C}^{n_u}$ and right interpolation driving frequencies $\{\lambda_i\}_{i=1}^{k} \in \mathbb{C}$ with right input or tangential directions $\{\mathbf{r}_i\}_{i=1}^{k} \in \mathbb{C}^{n_u}$, producing the right responses $\{\mathbf{w}_i\}_{i=1}^{k} \in \mathbb{C}^{n_y}$ (where $q + k = N$), find a (low order) transfer function $\hat{\mathbf{H}}$ that is a tangential (approximate) interpolant of the data, *i.e.* satisfies the following left and right interpolation conditions (let us assume that μ_j and λ_i are distinct):

$$\left.\begin{array}{l} \mathbf{l}_j^H \hat{\mathbf{H}}(\mu_j) = \mathbf{v}_j^H \\ \text{for } j = 1, \ldots, q \end{array}\right\} \text{ and } \left\{\begin{array}{l} \hat{\mathbf{H}}(\lambda_i)\mathbf{r}_i = \mathbf{w}_i \\ \text{for } i = 1, \ldots, k \end{array}\right. . \tag{2.6}$$

Through the use of Loewner matrices, this problem have been solved by Mayo and Antoulas (2007). It provides an analytical solution to the interpolatory problem for approximated model $\hat{\mathbf{H}}$ rational and of finite order, equipped with a realisation $\hat{\mathcal{S}}$. In addition, Mayo and Antoulas (2007) also provide the minimal dimension and McMillian degree of this interpolating model, when described by a first order **ODE** as in (1.6). Extensions to more complex structures have also been obtained by Schulze et al. (2018). We will come back to these points later in Section 2.3, dedicated to the parametrisation of the solutions.

The least square problem is also largely addressed in many engineering fields as well as its declination in the model approximation. Contrary to the interpolatory problem presented above, this one then involves an optimisation framework, rather than an analytical resolution one. This problem can be presented as a least square (see Gugercin and Antoulas (2006); Demourant and Poussot-Vassal (2017)) or vector fitting problem (see also Lefteriu and Antoulas (2013); Drmac et al. (2015b,a).

Problem 2.5: Least square model approximation

Given driving frequencies $\{s_i\}_{i=1}^{N} \in \mathbb{C}$, producing the responses $\{\mathbf{\Phi}_i\}_{j=1}^{N} \in \mathbb{C}^{n_y \times n_u}$, find a (low order) transfer function $\hat{\mathbf{H}}$ which (approximately) matches the data at the driving frequencies s_i, *i.e.* minimises the following criteria, where p is any matrix / vector norm:

$$\hat{\mathbf{H}} := \arg \min_{\substack{\mathbf{G} \in \mathcal{H}_\infty \\ \mathbf{rank}(\mathbf{G}) = r}} ||\mathbf{\Phi}_i - \mathbf{G}(s_i)||_p. \tag{2.7}$$

Obviously, it is nothing to mention that so many attempts to solve this problem have been suggested involving approaches such as nonlinear optimisation, genetic algorithm, sub-space *etc.* In my activities, I have marginally addressed this problem through the sub-space angle.

Now the objective criteria and different frameworks have been presented, let us move to the parametrisation of the solutions of the research space, denoted $\mathbf{G}$ in the above problems. As presented in Chapter 1, this transfer is a meromorphic function and might thus take multiple mathematical forms. Still, we will focus on some of them, more tailored to the control, simulation and optimisation communities and, we believe to practitioners needs[2].

[1] Other approaches based on discrete data exist in the identification domain, but are out of the scope of this manuscript. Author suggests to refer to Brunot (2017), Janot (2017) or Ljung (2013), for some details.

[2] However, author believes that there always is space for extending the models structure to address dedicated applications exhibiting specific needs (*e.g.* electronics, mechanical systems, fluid flow *etc.*).

2.3 Parametrization of the solutions

By either considering the $\mathcal{H}_2$ (2.3), $\mathcal{H}_{2,\Omega}$ (2.4), interpolation (2.6) or least square (2.7) problem described above, one may define different parametrisation of the solutions spaces $\mathbf{G}$. Obviously, there exists an infinite number of possibilities requiring a dedicated attention. However, in this manuscript we will more specifically target some traditional linear transfer functions, equipped with a realisation. Indeed, for simulation, analysis or control issues relevant in physics and control communities, they are of particular interest and are widely used. We start with the (delayed, parametric) first order form, then briefly introduce some other ones (second order, generalised coprime *etc.*).

2.3.1 First order LTI ODE and DAE models

The most commonly developed structure of the solution $\hat{\mathbf{H}}$ in the control literature is the first order model one, equipped with the following realisation,

$$\hat{\mathcal{S}} : \left\{ \begin{array}{rcl} \hat{E}\dot{\hat{\mathbf{x}}}(t) & = & \hat{A}\hat{\mathbf{x}}(t) + \hat{B}\mathbf{u}(t) \\ \hat{\mathbf{y}}(t) & = & \hat{C}\mathbf{x}(t) + \hat{D}\mathbf{u}(t) \end{array} \right. ,$$

where $\hat{\mathbf{x}}(t) \in \mathbb{R}^r$, $\mathbf{u}(t) \in \mathbb{R}^{n_u}$, $\hat{\mathbf{y}}(t) \in \mathbb{R}^{n_y}$ are the reduced state, input and approximated output vectors, respectively, and $\hat{E}, \hat{A} \in \mathbb{R}^{r\times r}$, $\hat{B} \in \mathbb{R}^{r\times n_u}$, $\hat{C} \in \mathbb{R}^{n_y\times r}$ and $\hat{D} \in \mathbb{R}^{n_y\times n_u}$. Its associated transfer function is,

$$\hat{\mathbf{H}}(s) = \hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B} + \hat{D}.$$

Remark 2.2 (About the direct feed-through) *As model approximation is concerned to match the transfer behaviour, a reduced order model usually features an equal feed-through term,* i.e. $D = \hat{D} \in \mathbb{R}^{n_y\times n_u}$. *One should note that if $D \neq \hat{D}$, then $(\mathbf{H}-\hat{\mathbf{H}})(\imath\infty) = D-\hat{D} \neq 0$ and, hence, $||\mathbf{H}-\hat{\mathbf{H}}||_{\mathcal{H}_2} = \infty$. Consequently, if one requires a good approximation in the sense of the $\mathcal{H}_2$-norm, then $D = \hat{D}$. Still, the feed-through $\hat{D}$ term may be used as an additional degree of freedom in order to improve the approximation. For example, the work of Flagg et al. (2013) presents a framework which uses this parameter in order to improve the approximation in the $\mathcal{H}_\infty$-norm sense. In addition, the weighted (see Anic et al. (2013)) and the frequency limited (see Vuillemin et al. (2014b) and Petersson and Lofberg (2014)) model approximation might provide reduced order models having the feed-through $\hat{D} \neq D$.*

Remark 2.3 (About DAE index) *The index (or differential index) is usually considered as a measure of the **DAE** singularity. By differentiating as many time as needed the equations, one can eliminate algebraic variables until obtaining an explicit **ODE** set. The index of a **DAE** is the number of derivative actions needed. Thus, **ODE** have an index of 0. Mathematically, it is always possible to take derivatives and rewrite **DAE** as **ODE** but algebraic equations replaced by their derivatives might no longer include the original constraints, then the numerical solution can drift and fail in reproducing the correct system's behaviour. Some research teams do develop specific work in this field (see* e.g. *Imran and Ghafoor (2015); Benner and Stykel (2016)). In this work, we will often consider that E and $\hat{E}$ are full row rank.*

As it is the main model structure employed in many control problems, this parametrisation has received a deep attention from the model approximation community and is thus the mostly employed one in many works such as in the reference book of Antoulas (2005) or survey papers Antoulas et al. (2001, 2016).

Details on some of the (many) proposed solutions will be given in the bibliographical Section 2.4. Still, some results addressing the $\mathcal{H}_2$ optimal objective (2.3), are relevant to notice now. Among them, one can mention the Gramian-based exposed by Wilson (1974) and the Krylov-based by Gugercin et al. (2008). Even if in the original work of Wilson (1974) it was not clearly shown, as highlighted in Gallivan et al. (2004a); Van Dooren et al. (2008); Gugercin et al. (2008), both solve a tangential interpolation problem as given in expression (2.5). In Gugercin et al. (2008), authors also have accompanied their developments with an iterative algorithm. The **IRKA** (Iterative Rational Krylov Algorithm), initially set for **SISO** systems in Gugercin et al. (2008), allows satisfying the $\mathcal{H}_2$ first-order optimality conditions but did not a priori preserves stability. Later, in Beattie and Gugercin (2009a), authors extended it to **MIMO** systems, with a complex Trust Region algorithm, guaranteeing $\mathcal{H}_2$ mismatch error monotonic descent and stability

preservation. Nowadays, **IRKA** is incontestably one of the most successful procedure to approximate very large-scale models. Almost at the same period, the works of Van Dooren et al. (2008); Van Dooren et al. (2010) has provided substantial results in the **MIMO** case, and especially, a connection between gramian and interpolatory conditions. Later, the Two-Sided Iterative Algorithm, **TSIA** by Xu and Zeng (2010), which iteratively solves two Sylvester equations, is shown to be equivalent to the tangential interpolations (see also Benner et al. (2011) for extensions and analysis of the potential stopping criteria). Around the same period, it can be noticed that rational approximation by using a Schur form by Marmorat et al. (2002) has also been exploited, allowing ensuring the model stability through a dedicated model structure. In this case, the process is achieved by an optimisation procedure. A side from these works, Vuillemin et al. (2014a) also addressed the $\mathcal{H}_2$ problem through an optimisation framework and using a pole residue representation of $\mathbf{H}$ and $\hat{\mathbf{H}}$, and using complex descent algorithm instead of projection approaches. This last formalisation has also led to solution of the $\mathcal{H}_{2,\Omega}$ optimal problem (2.4). In addition, the following related works may be considered Willcox and Megretski (2005); Gugercin and Willcox (2008), involving Fourier series. As for the model-based methods, in the context where no model exist but where frequency-domain data are available, Mayo and Antoulas (2007) did present a methodology based on the Loewner matrices, allowing constructing first order **DAE** models that interpolate these data, as targeted in Problem 2.3 (see also Antoulas et al. (2016) for a more detailed description). Note that most of these approaches consider that the E matrix is full rank and that the original model does not have any polynomial part. The case of rank defective E matrix is treated in Antoulas et al. (2010); Borggaard and Gugercin (2014); Imran and Ghafoor (2015); Benner and Stykel (2016). It is to be stressed that in the control community, **DAE** models are often neglected. We will come back to this case later in the chapter.

2.3.2 First order p-LTI ODE and DAE models

By continuing within the first order **ODE** and **DAE** models, and following the same notations, one can also consider the parametric ones, defined as follows:

$$\hat{\mathcal{S}}_{\mathbf{p}} : \left\{ \begin{array}{rcl} \hat{E}(\mathbf{p})\dot{\hat{\mathbf{x}}}(t) & = & \hat{A}(\mathbf{p})\hat{\mathbf{x}}(t) + \hat{B}(\mathbf{p})\mathbf{u}(t) \\ \hat{\mathbf{y}}(t) & = & \hat{C}(\mathbf{p})\mathbf{x}(t) \end{array} \right. ,$$

where $\mathbf{p} \in \mathbb{R}^{n_p}$, $\hat{E}(\mathbf{p}) \in \mathbb{R}^{r\times r}$, $\hat{A}(\mathbf{p}) \in \mathbb{R}^{r\times r}$, $\hat{B}(\mathbf{p}) \in \mathbb{R}^{r\times n_u}$ and $\hat{C}(\mathbf{p}) \in \mathbb{R}^{n_y\times r}$ are operators parametrised by vector $\mathbf{p}$. At a given parametric value, the associated transfer function is given as:

$$\hat{\mathbf{H}}(s,\mathbf{p}) = \hat{C}(\mathbf{p})\big(s\hat{E}(\mathbf{p}) - \hat{A}(\mathbf{p})\big)^{-1}\hat{B}(\mathbf{p}).$$

Remark 2.4 (About the parameter p and its (non) variations) *In the control community, **LPV** (Linear Parameter Varying) systems are largely used to extend the linear modeling, control design and analysis tools to a class of nonlinear models. In this case, the parameter* $\mathbf{p}$ *is actually a time-varying one and reads* $\mathbf{p}(t)$*. The associated "transfer function" cannot be written as it, since* $\mathbf{H}(s)$ *is no longer meromorphic. As a matter of consequence, the system's norm (such as the* $\mathcal{H}_2$ *of the* $\mathcal{H}_\infty$ *ones) are no longer applicable as it, and one talks* e.g. *about* L_2*-*L_2 *induced norm, instead of the* $\mathcal{H}_\infty$*. Still, many works in this area have been carried out since the 80's for control design, analysis, filtering design* etc. *Reader should refer* e.g. *to Arzelier (2014); Balas et al. (2003); Bruzelius et al. (2004); Biannic (2010); Toth et al. (2008); Alkhoury et al. (2018); Ghosh et al. (2018) for references and basics.*

Within the model approximation field, the works of Theis et al. (2015); Gozse et al. (2016) address this specific class of model using both Lyapunov equations and heuristic approaches, but results are contained to restricted model complexity. In this work, parameter dependent but time independent are considered instead. From the iterative approach family including the Krylov ones, the works by Eid et al. (2009); Lohmann and Eid (2009a,b) provide a methodology to construct parametric models from a set of local **LTI** ones. This approach has been re-used and extended by Poussot-Vassal and Roos (2012) in the case where the **LTI** set is sparse and specific adjustment have to be done for accurate interpolation of modes (see *e.g.* Chapter 3 for application on the long range parametric model construction). Moreover, from the data-driven side, the Loewner approach, initially settled in Mayo and Antoulas (2007) has also been extended for parametric models by Ionita and Antoulas (2014); Ionita (2013). It can be noticed that in this last work, the parametric model has a specific state-space structure, different from the one presented above.

2.3.3 Second order LTI ODE models

Back to non parametric models, the second order ones, well appreciated by practitioners in *e.g.* the mechanical, aero-elastic communities, *etc.* are given by the following form:

$$\hat{\mathcal{S}}_2 : \left\{ \begin{array}{rcl} \hat{M}\ddot{\hat{\mathbf{x}}}(t) + \hat{L}\dot{\hat{\mathbf{x}}}(t) + \hat{K}\hat{\mathbf{x}}(t) & = & \hat{B}\mathbf{u}(t) \\ \hat{\mathbf{y}}(t) & = & \hat{C}\hat{\mathbf{x}}(t) \end{array} \right. ,$$

where $\hat{M} \in \mathbb{R}^{r\times r}$, $\hat{L} \in \mathbb{R}^{r\times r}$, $\hat{K} \in \mathbb{R}^{r\times r}$, $\hat{B} \in \mathbb{R}^{r\times n_u}$ and $\hat{C} \in \mathbb{R}^{n_y\times r}$. The corresponding transfer functions reads

$$\hat{\mathbf{H}}_2(s) = \hat{C}(s^2\hat{M} + \hat{L}s + \hat{K})^{-1}\hat{B}.$$

One interest in the second order formulation lies in the physical meaning of the matrices (*e.g.* $\hat{M}$ is a mass matrix, $\hat{L}$ and $\hat{K}$ are damping and stiffness ones, respectively). This model structure is relevant for engineers in order to understand and optimise the physical elements. Clearly this parametrisation embeds an interest for specific applications. An illustration of model approximation and optimisation, the tuning of the damping parameters is explored in Tomljanovic et al. (2018). Still, from a mathematical point of view, any second order model can be recast as a first order one as

$$\hat{\mathcal{S}}_{2*} : \left\{ \begin{array}{rcl} \left[\begin{array}{cc} I & 0 \\ 0 & \hat{M} \end{array} \right] \left[\begin{array}{c} \dot{\hat{\mathbf{x}}}(t) \\ \ddot{\hat{\mathbf{x}}}(t) \end{array} \right] & = & \left[\begin{array}{cc} 0 & I \\ -\hat{K} & -\hat{L} \end{array} \right] \left[\begin{array}{c} \hat{\mathbf{x}}(t) \\ \dot{\hat{\mathbf{x}}}(t) \end{array} \right] + \left[\begin{array}{c} 0 \\ \hat{B} \end{array} \right] \mathbf{u}(t) \\ \mathbf{y}(t) & = & \left[\begin{array}{cc} \hat{C} & 0 \end{array} \right] \left[\begin{array}{c} \hat{\mathbf{x}}(t) \\ \dot{\hat{\mathbf{x}}}(t) \end{array} \right] \end{array} \right. .$$

Thus, all first order **ODE** methods can apply to this class, obviously at the price of twice the original dimension.

2.3.4 First order input-output delayed LTI ODE and DAE models

Following the same notations, the input-output delayed model structure stands as a natural extension of the **DAE LTI** model one, and is defined as follows, with $\{\tau_i\}_{i=1}^{n_u} \in \mathbb{R}_+$ and $\{\gamma_o\}_{o=1}^{n_y} \in \mathbb{R}_+$:

$$\hat{\mathcal{S}}_d : \left\{ \begin{array}{rcl} \hat{E}\dot{\hat{\mathbf{x}}}(t) & = & \hat{A}\hat{\mathbf{x}}(t) + \hat{\mathbf{\Delta}}_{\boldsymbol{i}}\big(\hat{B}\mathbf{u}(t)\big) \\ \hat{\mathbf{y}}(t) & = & \hat{\mathbf{\Delta}}_{\boldsymbol{o}}\big(\hat{C}\hat{\mathbf{x}}(t)\big) \end{array} \right. ,$$

where $\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(\cdot)$ and $\hat{\mathbf{\Delta}}_{\boldsymbol{o}}(\cdot)$ are the linear input and output fixed delayed operators, respectively defined as:

$$\left\{ \begin{array}{rcl} \hat{\mathbf{\Delta}}_{\boldsymbol{i}}\big(\mathbf{u}(t)\big) & = & \left[\begin{array}{ccc} u(t-\tau_1) & \dots & u(t-\tau_{n_u}) \end{array} \right]^T \in \mathbb{R}^{n_u} \\ \hat{\mathbf{\Delta}}_{\boldsymbol{o}}\big(\hat{\mathbf{y}}(t)\big) & = & \left[\begin{array}{ccc} \hat{y}(t-\gamma_1) & \dots & \hat{y}(t-\gamma_{n_y}) \end{array} \right]^T \in \mathbb{R}^{n_y} \end{array} \right. .$$

By keeping the notations, the associated transfer function is,

$$\hat{\mathbf{H}}_d(s) = \hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s)\hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B}\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s) = \hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s)\hat{\mathbf{H}}(s)\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s),$$

where $\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s)$ and $\hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s)$ are the input and output fixed delayed operators, defined as:

$$\left\{ \begin{array}{rcl} \hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s) & = & \mathbf{diag}(e^{-\tau_1 s}, \dots, e^{-\tau_{n_u} s}) \in \mathcal{H}_\infty^{n_u\times n_u} \\ \hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s) & = & \mathbf{diag}(e^{-\gamma_1 s}, \dots, e^{-\gamma_{n_y} s}) \in \mathcal{H}_\infty^{n_y\times n_y} \end{array} \right. .$$

This parametrisation naturally extends the delay-free **LTI** one. In the model approximation context, such a parametrisation has been first considered in the work of Halevi (1992), providing an $\mathcal{H}_2$ optimal condition set through Lyapunov and gramian equations. Later, Pontes Duff et al. (2018) provide optimality conditions in term of interpolatory equations as well as an approach, similar to the **IRKA** one (dedicated to delay-free reduced order models), called **IO-dITIA** for Inputs-Outputs delayed Iterative Tangential Interpolation Algorithm. Note that although no gain variations are observed when adding input-output delay, the mismatch between a delayed and a delay-free model affects the norm computation through the phase shift embedded in the delay term[3].

[3] We will come back to this parametrization in Chapter 4, but interested reader can refer to Briat (2015); Seuret (2017) for good references and details on time delayed systems.

2.3.5 Generalized coprime LTI models

Until now, both the full and the reduced-order models were assumed to describe a finite dimensional transfer function $\hat{\mathbf{H}}$, and described by a realisation $\hat{\mathcal{S}}$ representing an **ODE** and **DAE** set. In what follows, we consider a transfer function having the following generalised coprime and parametric coprime representation:

$$\hat{\mathbf{H}}(s) = \hat{\mathcal{C}}(s)\hat{\mathcal{K}}(s)^{-1}\hat{\mathcal{B}}(s) \text{ and } \hat{\mathbf{H}}(s,\mathbf{p}) = \hat{\mathcal{C}}(s,\mathbf{p})\hat{\mathcal{K}}(s,\mathbf{p})^{-1}\hat{\mathcal{B}}(s,\mathbf{p}),$$

where both $\hat{\mathcal{C}}(s), \hat{\mathcal{C}}(s,\mathbf{p}) \in \mathbb{C}^{n_y \times r}$ and $\hat{\mathcal{B}}(s), \hat{\mathcal{B}}(s,\mathbf{p}) \in \mathbb{C}^{r \times n_u}$ are analytic in the right half plane, and $\hat{\mathcal{K}}(s), \hat{\mathcal{K}}(s,\mathbf{p}) \in \mathbb{C}^{r \times r}$ is analytic and full rank over the right half plane. Obviously, this coprime representation encapsulates all the above listed models (first, second order, input-output delayed), and even more, such as the (state) time-delayed systems given as

$$\hat{\mathbf{H}}(s) = \hat{C}\left(s\hat{E} - \sum_{k}^{n_d} \hat{A}_k e^{\tau_k s}\right)^{-1} \hat{B},$$

where $\hat{A}_k \in \mathbb{R}^{r \times r}$ and $\tau_k \in \mathbb{R}_+$, for $k = 1, \ldots, n_d$. In Beattie and Gugercin (2009b); Gugercin et al. (2012) authors present an interpolatory method to construct a reduced orde model that preserve the system's original structure through the generalised coprime form. However, it is to be noticed that in these work, the proposed procedures do not solve any $\mathcal{H}_2$ (or other norm) optimal problem, but rather take inspiration from mechanisms used in the classical **ODE** case for this extended form.

2.3.6 Schur form

In Fulcheri and Olivi (1998) and in Marmorat et al. (2002), authors use a parametrisation of the reduced-order model state-space representation based on its Douglas-Shapiro-Shield factorisation (see for instance Fuhrmann, 1994). Such a parametrisation enables to have a minimal (in terms of the number of parameters) and smooth representation that directly embeds some properties such as the stability. This last form is known as the Schur form and is also largely used in identification.

In the works of Baratchart et al. (1991); Fulcheri and Olivi (1998); Marmorat et al. (2002), authors also use a parametrisation in the disk rather than in the complex plane. This latter shows nice numerical properties[4]. In the context of discrete-time models (Baratchart, 1986, see) has presented some theoretical results including the minimum existence. In the articles of Baratchart et al. (1991); Fulcheri and Olivi (1998); Marmorat et al. (2002) different algorithms to solve the problem using a dedicated parametrization, are presented. It should be noticed that there is an isometry from $\mathcal{H}_2(\mathbb{C}_+)$ to the orthogonal of $\mathcal{H}_2(\mathbb{D})$ (Hardy space for discrete-time systems) so that methods used for discrete-time systems can also be used for continuous-time and vice versa (see Olivi et al., 2013).

Still, in practice few algorithm are deployed in the discrete time version while the unit disk provides very interesting properties for analysis, numerical issues, *etc.*

[4]The method is also made available in the software RARL2 by Marmorat and Olivi (2002) at http://www-sop.inria.fr/apics/RARL2/rarl2.html.

2.4 Overview of existing methods and bibliographical notes

2.4.1 Forewords

Based on the approximation objectives described in Section 2.2 and on the parametrisation of the solution in Section 2.3, we will now give some of the important literature results. However, before reading the following pages, author stresses that many results exist, and this monograph does not pretend giving an exhaustive view, but rather a view of the references and methods which drove my research activities. The following pages can be viewed as the necessary background before reading Chapters 3, 4 and 5. Indeed, the approximation problem has a long lasting history and has been addressed by both the numerical and control communities. Both angles of attack are interesting and lead to a wide variety of approaches. To the author's point of view, a very good starting point are the books of Antoulas (2005) and Saad (2000)[5]. The former addressing the theoretical and methodological aspects on approximation of dynamical systems, and the latter, the numerical and linear algebra computational considerations.

> This research filed is, to the author's point of view, deeply connected to dynamical system's theory, linear algebra and complex functions analysis. Therefore, it is fair considering that model approximation is somehow at the intersections of these research fields and it is thus possible to find relevant contributions in all these domains.

Providing an overview and a classification of the linear model approximation methods is quite complicated and many patterns can be used. One can either subdivide them by original model class (finite order vs. infinite order, model-based vs. data-based), by methods family (projection vs. non-projection, direct vs. iterative) or even by objective / target ($\mathcal{H}_2$, $\mathcal{H}_\infty$, interpolatory) *etc.* In what follows, we chose to keep the path described by the workflow illustrated on Figure 2.1, by first presenting the model-based approximation methods, firstly based on a realisation as in $\#1$ (by projection) and then, based on a realisation-free models as in $\#2$. Then, the data-driven framework $\#3$ is finally presented. According to the author, such an organisation allows non-familiar reader to select the approximation method according to its starting problematic, but also to make some bridges between the methods.

Remark 2.5 (Original and reduced models structures) *In general, and it is the case in most model approximation results, the goal is to construct a reduced-order model $\hat{\mathbf{H}}$, keeping the same structure as the original one $\mathbf{H}$. Indeed, practitioners usually chose a mathematical representation of their model and then, aim at simplifying it. As a matter of consequence, most of the results presented hereafter will consider state-space representations of first order **ODE** structures. A different structure of the reduced model is considered in Pontes Duff et al. (2018) and exposed in Chapter 4.*

Before detailing some approximation methods, author stresses that even if most of them are based on **LTI** models $\mathbf{H}$ equipped with a realisation $\mathcal{S} : (E, A, B, C, D)$, extensions to more complex structure ma exist. Approximation involving some more general representations will be exposed, however, attention is mainly given on models with realisation as presented in Section 2.3 and involving the criteria given in Section 2.2.

2.4.2 Model-based approximation by projection and Petrov-Galerkin framework

One of the most efficient medium and large-scale dynamical model approximation techniques, are the projection-based ones. These latter are based on the Petrov-Galerkin projection problem. Given a linear dynamical system $E\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t))$ ($\mathbf{x}(t) \in \mathbb{R}^n$ and $\mathbf{u}(t) \in \mathbb{R}^{n_u}$), this problem consists in finding projectors *(i)* $V \in \mathbb{R}^{n\times r}$ ($r \ll n$) forming a basis of subspace $\mathcal{V}$, and *(ii)* $W = (\overline{W}^T EV)^{-1}\overline{W}$ (where $\overline{W} \in \mathbb{R}^{n\times r}$ is a basis of subspace $\mathcal{W}$), such that W^TV is invertible and

$$\hat{\mathbf{x}}(t) \in \mathcal{V} \text{ and } W^T\left(EV\hat{\mathbf{x}}(t) - AV\hat{\mathbf{x}}(t) - B\mathbf{u}(t)\right) = 0$$

[5] People with a mathematical background may also be interested in having close-by the book of Partington (2004).

Here $\mathbf{x}(t) \approx V\hat{\mathbf{x}}(t)$ and $\hat{\mathbf{x}}(t) \approx W^T\mathbf{x}(t)$ are the full original and reduced order state-space vectors, respectively. The projection-based model approximation is then given as follows.

Problem 2.6: Projection-based linear model approximation

Given $\mathbf{H}$, a n_u inputs n_y outputs **LTI** dynamical model with realisation $\mathcal{S} : (E, A, B, C, 0)$ of order n, defined as

$$\mathcal{S} : \begin{cases} E\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) \end{cases}$$

where $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$ and $C \in \mathbb{R}^{n_y\times n}$. The projection-based approximation problem consists of finding a projector $\Pi_{V,W} = VW^T \in \mathbb{R}^{n\times n}$ (with $V, W \in \mathbb{R}^{n\times r}$, $W^TV = I_r$), such that $\hat{\mathcal{S}} : (\hat{E}, \hat{A}, \hat{B}, \hat{C}, 0)$, a reduced model of order $r \ll n$, defined as

$$\hat{\mathcal{S}} : \begin{cases} \hat{E}\dot{\hat{\mathbf{x}}}(t) &= \hat{A}\hat{\mathbf{x}}(t) + \hat{B}\mathbf{u}(t) \\ \hat{\mathbf{y}}(t) &= \hat{C}\hat{\mathbf{x}}(t) \end{cases}$$

where $\hat{A} = W^TAV$, $\hat{B} = W^TB$ and $\hat{C} = CV$, well approximates $\mathbf{H}$ in the sense of a given measure.

In this specific case, research can be divided in two mainstreams: *(i)* the theoretical one, where attention is devoted to the development of new methods guaranteeing some approximation properties (*e.g.* stability, structure, passivity, $\mathcal{H}_2$ optimality…), and, *(ii)* the numerical one, which focusses on the development of numerically robust and fast procedures implementing the aforementioned theoretical methods. A projection framework for model approximation was introduced by Villemagne and Skelton (1987). In this regard, matrices V and W are called projection matrices since they are the fundamental blocks to generate the reduced order model. Indeed, given a large-scale model represented by its realisation $\mathcal{S}$, a reduced order model $\hat{\mathcal{S}}$ can be constructed by a pair of V and W projection matrices. These matrices V and W span the subspaces $\mathcal{V}$ and $\mathcal{W}$, respectively. If subspaces $\mathcal{V}$ and $\mathcal{W}$ are equal, one talks about orthogonal projection. Otherwise, one talks about oblique projection. In this framework, finding a good approximation $\hat{\mathcal{S}}$ is equivalent in finding matrices V, W such that $\hat{\mathcal{S}}$, the reduced order model constructed by projection, has a similar behaviour as $\mathcal{S}$. In the reasoning, only the subspaces $\mathcal{V} = \mathbf{span}\,(V)$ and $\mathcal{W} = \mathbf{span}\,(W)$ play an important role in the projection framework and the projection matrices are not important in this regard (see Gallivan et al. (2004a)). This statement is summarised as follows.

> Choosing two different bases V' and W' that respectively span the same subspaces $\mathcal{V}$ and $\mathcal{W}$ result in the same reconstructed solution $\mathbf{x}(t)$. Thus, subspaces are relevant, not basis. One can say that a reduced order model is uniquely defined by its projector $\Pi_{V,W}$ which are defined by the two subspaces
>
> $$\mathcal{V} = \mathbf{span}\,(V) \quad \text{and} \quad \mathcal{W} = \mathbf{span}\,(W)\,,$$
>
> where $\mathcal{V}$ and $\mathcal{W}$ belong to the Grassmann manifold $\mathcal{G}(r, n)$, known as the set of all subspaces of dimension r in $\mathbb{R}^n$. More specifically, within model reduction, for numerical reasons, one is often seeking for orthonormal vectors V and W. One talk of Stiefel manifolds $\mathcal{S}(r, n)$

An important number of model approximation techniques can be clipped to this projection-based framework. This chapter does not intend providing an exhaustive review of all of them, however, in what follows, a list of famous model approximation techniques is given, involving some well known projection subspaces $\mathcal{V}$ and $\mathcal{W}$.

2.4.3 Model-based approximation by projection and truncation

General idea

The underlying framework of truncation consists in dividing the state-space vector $\mathbf{x}(t)$ in two subsets

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{bmatrix},$$

where $\mathbf{x}_1(t) \in \mathbb{R}^r$ and $\mathbf{x}_2(t) \in \mathbb{R}^{n-r}$ and by applying the orthogonal projection matrices

$$V = \begin{bmatrix} I_r \\ 0_{n-r} \end{bmatrix} \in \mathbb{R}^{n \times r} \text{ and } W = V.$$

In the general case, the truncation does not have any particular property from the original model $\mathbf{H}$. However, if the original model is pre-processed in a certain way, this projection (truncation) is meaningful. In what follows, as they are clearly the most largely used in the literature and industry, two approaches based on this idea are briefly presented: the modal and balanced truncations.

Modal truncation

This method basically consists in truncating the original realisation $\mathcal{S}$ in its modal basis. It thus requires first a preliminary transformation of the original system into its modal form. This is usually done by computing of the (A, E) pencil, followed by the truncation. Because it perfectly reproduces a subset of the original model modal content, this methods is particularly appreciated by practitioners. Moreover, it is accompanied with an error bound on the approximation error. Still, the way to choose the eigenvalues to preserve, requires an expert eye. However, it can be mentioned that dedicated algorithms have been developed to efficiently compute iteratively the dominant poles of a large-scale model. In Martins et al. (1996), the so-called dominant pole algorithm is presented. This algorithm uses the Newton's method to compute a dominant pole of a **SISO** model. Later, this algorithm is improved and extended to a robust and efficient method for **MIMO** ones in Rommes and Martin (2006) and for second order models in Rommes and Martins (2008). Then, Rommes and Sleijpen (2008) and Rommes et al. (2010) present some convergence properties, theoretical results and some comparisons with the Rayleigh quotient iteration. Additionally, passivity preserving methods were also proposed by Ionutiu et al. (2008). We encourage the reader to refer to these articles and references theirein for more details. The MATLAB Robust Control Toolbox embeds routines allowing this truncation, with the possibility to simply discard fast dynamics (*i.e.* truncating according to the magnitude of the eigenvalues).

Balanced truncation

Similarly to the modal one, the balanced truncation consists in first balancing the original model realisation $\mathcal{S}$, then in truncating (see Moore (1981); Safonov and Chiang (1989); Zhou and Doyle (1997); Xu and Zeng (2010); Benner et al. (2011)). Without entering into details, a system is said to be balanced if and only if

$$\mathcal{P} = \mathcal{Q} = \Sigma = \mathbf{diag}(\sigma_1, \ldots, \sigma_n)$$

where $\mathcal{P}$ and $\mathcal{Q}$ are the solution of the Lyapunov equations (1.10) and $\{\sigma_i\}_{i=1}^n$ are the singular values of them. The balanced form is obtained by applying the following projection matrices:

$$V = UZ\Sigma^{-1/2} \text{ and } W = LY\Sigma^{-1/2},$$

where $\mathcal{P} = UU^T$ and $\mathcal{Q} = LL^T$ (Cholesky factorisation) and $U^TL = Z\Sigma Y^T$ (SVD factorisation). Applying the truncation, usually by watching the singular values decay leads to the approximated model $\hat{\mathcal{S}}$. The resulting reduced order system $\mathbf{H}$, is asymptotically stable (if $\mathbf{H}$ is stable), $\hat{\mathbf{H}}(\infty) = \mathbf{H}(\infty)$, and the mismatch error between original and reduced systems is bounded by the following relation,

$$\sigma_r \leq ||\mathbf{H} - \hat{\mathbf{H}}||_{\mathcal{H}_\infty} \leq 2(\sigma_{r+1} + \cdots + \sigma_n).$$

Although it is the only method (with the eigenvalue one) to embed an a-priori mismatch upper bound, this latter is really pessimistic and is not really useful in practice (see Vuillemin, 2014).

Once the original realisation is in the balanced form, after some simple algebraic manipulation, it also possible to compute the so-called balanced singular perturbation model, allowing ensuring that $\hat{\mathcal{S}}$ is asymptotically stable and $\hat{\mathbf{H}}(0) = \mathbf{H}(0)$, ensuring static gain matching.

The MATLAB Robust Control Toolbox (see Glover, 1984; Safonov and Chiang, 1989) also implements the balanced truncation methods. In addition, the SLICOT interface by Varga (1999), which is a MATLAB Toolbox, implements numerically reliable and efficient techniques, including balanced and truncated singular perturbation approximation, balanced stochastic truncation, frequency-weighting balancing, *etc.* Very recently, the MORLAB Toolbox by P. Benner's research group, encapsulating modal truncation, balanced truncation, bounded-real balanced truncation, positive-real balanced truncation, balanced stochastic truncation, linear-quadratic-Gaussian balanced truncation, $\mathcal{H}_\infty$ balanced truncation and Hankel-norm approximation, for dense problems, has been released (see Benner and Werner, 2017).

Some truncation-based extensions

For a general overview, the interested reader can refer to the books Antoulas (2005) and Schilders et al. (2008), the classical survey papers by Antoulas et al. (2001); Gugercin and Antoulas (2004); Benner et al. (2005); Antoulas (2009); Baur et al. (2009, 2011), and the recent survey papers by Antoulas et al. (2010, 2016); Beattie and Gugercin (2016). In addition, for model approximation of parametric dynamical systems, see the recent survey of Benner et al. (2015). For differential algebraic systems see also Benner and Stykel (2016). The following lists some of the specific model classes methods:

- Second-order systems: both papers by Meyer and Srinivasan (1996) and Chahlaoui et al. (2006) develop different structures preserving model reduction methods for second-order systems.
- Unstable systems: an algorithm was first developed in Therapos (1989) for unstable non-minimal systems and then generalised in Zhou et al. (1999).
- Inhomogeneous initial conditions: Heinkenschloss et al. (2011) did present a new method allowing model approximation for systems with inhomogeneous initial conditions, by adding auxiliary inputs derived from the initial conditions.
- Infinite dimensional systems: Glover et al. (1988) extends the balanced truncation to the class of infinite-dimensional continuous-time systems. More recently, papers by Reis and Selig (2014); Guiver and Opmeer (2014) generalise those results.
- Time-varying systems: one extension of balanced truncation for time-varying systems is developed in Sandberg and Murray (2008); Siahaan (2008). Applications using a projection method are presented in Sandberg (2006)[6].
- Uncertain systems: approximation of parametric and uncertain systems are also addressed in Benner and Grundel (2015).
- Nonlinear systems: balanced truncation for nonlinear systems is introduced in Scherpen (1993). A form of truncation based on the system's trajectory is done in the so-called Proper Orthogonal Decomposition (see Antoulas (2005); Peherstorfer and Willcox (2018) and the references therein for further information). Moreover, Himpe and Ohlberger (2013); Himpe et al. (2013); Benner et al. (2018) did extends the gramian notion to the empirical gramian, allowing to deal with nonlinear systems. Interestingly, C. Himpe also provides a numerical interface through a MATLAB software called **emgr** [7].

Let us now give some attention to the so-called $\mathcal{H}_2$ approximation problem. The solution to this problem is first given in the gramian fashion, then in the interpolatory one.

[6] One may note that in Melchior et al. (2012), an interpolatory framework applied to periodic discrete-time models is presented.
[7] See www.gramian.de and Himpe (2016)

2.4.4 Model-based approximation by projection and $\mathcal{H}_2$-optimality

Up to author's knowledge, within the projection framework involving system's realisation, Wilson (1974) was the first to derive the so-called first order $\mathcal{H}_2$ optimality conditions. Interestingly, these conditions were described in a theoretical manner in the 70's, while a connection with the interpolatory Problem 2.3, performed only in the late 2000, and quite recently, embedded in an efficient numerical iterative scheme. The main results of Wilson (1974) are recalled hereafter.

Theorem 2.1: State-space first-order $\mathcal{H}_2$ optimality conditions

Given $\mathbf{H} \in \mathcal{H}_2$ equipped with realisation $\mathcal{S} : (I_n, A, B, C, 0)$ and $\hat{\mathbf{H}} \in \mathcal{H}_2$ equipped with realisation $\hat{\mathcal{S}} : (I_r, \hat{A}, \hat{B}, \hat{C}, 0)$, the following controllability and observability gramian of the error system $\tilde{\mathbf{E}} = \mathbf{H} - \hat{\mathbf{H}}$ can be computed

$$\tilde{\mathcal{P}} = \begin{bmatrix} P & X \\ X^T & \hat{P} \end{bmatrix} \text{ and } \tilde{\mathcal{Q}} = \begin{bmatrix} Q & Y \\ Y^T & \hat{Q} \end{bmatrix},$$

If,

$$\hat{Q}\hat{P} + Y^T X = 0, \;\; \hat{Q}\hat{B} + Y^T B = 0, \;\; \text{and} \;\; \hat{C}\hat{P} - CX = 0. \tag{2.8}$$

then, by denoting with $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}) = ||\mathbf{H} - \hat{\mathbf{H}}||_{\mathcal{H}_2}$, shortly $\mathcal{J}_{\mathcal{H}_2}$, one have

$$\frac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{A}} = 0, \;\; \frac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{B}} = 0 \;\; \text{and} \;\; \frac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{C}} = 0.$$

Following Theorem 2.1 and optimality conditions as a set of equations depending on the gramian of the system's error $\tilde{\mathbf{E}}$, the reduced model is then obtained by linking (2.8) with the projectors as follows.

Corollary 2.1: State-space first-order $\mathcal{H}_2$ optimality conditions

Following notations of Theorem 2.1, at every stationary point of functional $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}})$, *i.e.* $\nabla \mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}) = 0$, where $\hat{P}$ and $\hat{Q}$ are invertible, we have the following identities: $\hat{A} = W^T A V$, $\hat{B} = W^T B$ and $\hat{C} = CV$ with

$$W^T V = I_r, \;\; W = -Y\hat{Q}^{-1} \;\; \text{and} \;\; V = X\hat{P}^{-1}$$

where X, Y, $\hat{P}$ and $\hat{Q}$ satisfy the following Sylvester and Lyapunov equations,

$$\begin{aligned} AP + PA^T + BB^T = 0 \quad &, \quad QA + A^T Q + C^T C = 0 \\ \hat{A}X^T + X^T A^T + \hat{B}B^T = 0 \quad &, \quad A^T Y + Y\hat{A} - C^T \hat{C} = 0 \\ \hat{A}\hat{P} + \hat{P}\hat{A}^T + \hat{B}\hat{B}^T = 0 \quad &, \quad \hat{Q}\hat{A} + \hat{A}^T \hat{Q} + \hat{C}^T \hat{C} = 0. \end{aligned}$$

Corollary 2.1 determines the projectors V and W as functions of the mismatch model gramians. This observation was exploited later by Van Dooren et al. (2008) to suggest and iterative procedure to construct the projectors. In Xu and Zeng (2010), authors derived an iterative procedure, called **TSIA** for Two-Sided Iterative Algorithm. This latter iteratively solves two Sylvester equations in order to compute the quadruplet $\{\hat{P}, \hat{Q}, Y, X\}$, defining the projectors V and W. The procedure proposed by Xu and Zeng (2010) preserves stability of the original model but suffers of two main drawbacks: first, it requires a good projector initialisation to converge, and secondly, no stopping criterion is given. Later, Benner et al. (2011) explored three different stopping criteria that can be connected to this algorithm. Moreover, as exposed in the following section and in Gallivan et al. (2004a); Van Dooren et al. (2008), these conditions have been shown to be equivalent to the tangential interpolation ones. During my research activities, I have been mostly using the interpolatory techniques, rather than Lyapunov ones. Some reasons are its numerical efficiency, its flexibility and variety of extended possibilities. Indeed, as clarified later along the manuscript, this class of method are well appropriate for sparse and very large-scale models and shows an impressive versatility. Let us now move to this interpolatory framework.

2.4.5 The model-based approximation by interpolations

Forewords interpolation

Interpolation, as described in Problem 2.3, is a simple and yet effective approach that is used for the general approximation of complex functions using simpler ones. The accuracy of the resulting approximations and the connections with strategic placement of interpolating points λ_i and μ_j has been studied in many broad context. Indeed, in the case of interpolation of meromorphic functions by polynomials or rational functions, one can also associate it with classical complex mathematical analysis. Before getting into Problem 2.3, let us first define the moment matching problem and its links with the projection framework (author believes that this concept is quite didactic before going to multi-point and rational interpolation). Then, we will show how the interpolatory conditions can be achieved by appropriate projection. Finally, the tangential interpolatory problem will be stated.

As in the previous subsections, this one is also concerned with case #1 of Figure 2.1. In addition, and it's quite strong advantage of this method family, we will show that the interpolatory framework also fits the use-case #2 of Figure 2.1, for which no realisation is available. This framework embeds then a wider class of transfer functions.

Approximation by moment matching and connection with projectors

Let us be given two **LTI** models $\mathbf{H}$ and $\hat{\mathbf{H}}$, which can be expanded at $\sigma \in \mathbb{C}$ as

$$\mathbf{H}(s)|_{\sigma} = \sum_{i=0}^{\infty} \eta_i(\sigma)(s-\sigma)^i \quad \text{and} \quad \hat{\mathbf{H}}(s)\Big|_{\sigma} = \sum_{i=0}^{\infty} \hat{\eta}_i(\sigma)(s-\sigma)^i,$$

the moment matching problem consists in ensuring that $\forall i \in 1, \ldots, r$, $\eta_i(\sigma) = \hat{\eta}_i(\sigma)$. This condition can easily be explicitly obtained in a pure mathematical reasoning. However, moments computation is known to be numerically ill-conditioned, especially in the large-scale context (as illustrated for the special case of single-input single-output dynamical systems in Feldman and Freund (1995)). Still, the moment matching property may be achieved through projection, by carefully choosing test and trial subspaces $\mathcal{V}$ and $\mathcal{W}$, without computing them explicitly.

Moment matching and interpolatory projections for model reduction were introduced by Yousouff and Skelton (1985); Yousouff et al. (1985); Villemagne and Skelton (1987). Later, Grimme et al. (1996); Grimme (1997) embed this approach into a numerically efficient framework by using the rational Krylov subspace method of Ruhe (1984), defined in (1.13). The projection framework for the problem setting we are interested in, that is, for rational tangential interpolation of **MIMO** dynamical systems, has been then developed by Gallivan et al. (2004a) for the case of Problem 2.3 and by Mayo and Antoulas (2007) for the case of Problem 2.4. Let us first provide what one can consider as a pivotal result for moment matching at one single point σ, also called interpolation point, in the following theorem.

Theorem 2.2: Single point two-sided interpolatory conditions

Given an **LTI** dynamical model $\mathcal{S} : (E, A, B, C, D)$ and $\sigma \in \mathbb{C}$ s.t. $(\sigma E - A)$ is full rank. If $V, W \in \mathbb{C}^{n \times r}$ are full column rank matrices such that

$$\begin{array}{rcccl} \mathcal{K}\big((\sigma E - A)^{-1}, (\sigma E - A)^{-1}B, r\big) & \subseteq & \mathcal{V} & = & \mathbf{span}\,(V) \\ \mathcal{K}\big((\sigma E - A)^{-H}, (\sigma E - A)^{-H}C^T, r\big) & \subseteq & \mathcal{W} & = & \mathbf{span}\,(W) \end{array} \tag{2.9}$$

then, the $2r$ first moments of the reduced-order model $\hat{\mathbf{H}}$, obtained by projection, matches the $2r$ first moments of $\mathbf{H}$ at σ, *i.e.*

$$\eta_i(\sigma) = \hat{\eta}_i(\sigma), \quad i = 1, \ldots, 2r.$$

According to Theorem 2.2, it is possible to match moments without explicitly computing them. Moreover, it states that it is possible to reach this property by a Krylov subspace (2.9) construction, which can be done in a very efficient numerical way by mean of Arnoldi or Lanczos procedures, as illustrated in Chapter 1. Practically, constructing $\mathcal{K}\left(E^{-1}A, E^{-1}B, r\right)$ allows matching moments in $\sigma = \infty$, known as the

Markov coefficients, $\mathcal{K}\left(EA^{-1}, B\right)$ matches in $\sigma = 0$, known as the Padé ones, and $\mathcal{K}\left((\sigma E - A)^{-1}, B\right)$ at any $\sigma \in \mathbb{C}$. Then, the straightforward extension of Theorem 2.2 is the multi-point moment matching, given in Theorem 2.3.

Theorem 2.3: Multi-points point two-sided interpolatory conditions

Given an **LTI** dynamical model $\mathcal{S} : (E, A, B, C, D)$ and $\{\sigma_k\}_{k=1}^{n_\sigma} \in \mathbb{C}^{n_\sigma}$, $\{r_k\}_{k=1}^{n_\sigma} \in \mathbb{N}$ s.t. $\forall k = 1, \dots n_\sigma$, $(\sigma_k E - A)$ is full rank. If $V, W \in \mathbb{C}^{n \times r}$ are full column rank matrices such that

$$\begin{array}{rcccl} \bigcup_{k=1}^{n_\sigma} \mathcal{K}\big((\sigma_k E - A)^{-1}, (\sigma_k E - A)^{-1} B, r_k\big) & \subseteq & \mathcal{V} & = & \mathbf{span}\,(V) \\ \bigcup_{k=1}^{n_\sigma} \mathcal{K}\big((\sigma_k E - A)^{-H}, (\sigma_k E - A)^{-H} C^T, r_k\big) & \subseteq & \mathcal{W} & = & \mathbf{span}\,(W) \end{array} \tag{2.10}$$

then, the $2r_k$ first moments of the reduced-order model $\hat{\mathbf{H}}$, obtained by projection, matches the $2r_k$ first moments of $\mathbf{H}$ at each σ_k, *i.e.* for $k = 1, \dots, n_\sigma$,

$$\eta_i(\sigma_k) = \hat{\eta}_i(\sigma_k), \quad i = 0, \dots, 2r_k - 1.$$

Theorem 2.3 provides a way to interpolate $\mathbf{H}$ (here equipped with a **DAE** realisation $\mathcal{S}$) at any interpolation points $\{\sigma_k\}_{k=1}^{n_\sigma}$ by constructing projectors V and W through appropriate Krylov subspaces, and, as consequence, to ensure moment matching of $\mathbf{H}$ and its projected (reduced) version $\hat{\mathbf{H}}$. Still, attentive reader should notice that if a **MIMO** model is considered, the dimension of (2.10) will with the number of inputs and outputs. This constatation is one of the justifications for interest in the so-called tangential interpolation, exposed in Problem 2.3.

Remark 2.6 (About block Arnoldi and "tangential" Arnoldi) *When referring to (2.10), it is clear that for **SISO** models, a standard Arnoldi procedure allows constructing $V, W \in \mathbb{C}^{n \times r}$. However, applying such a process on a **MIMO** case will lead to larger projectors $V \in \mathbb{C}^{n \times r n_u}$ and $W \in \mathbb{C}^{n \times r n_y}$, which may present rank losses and a projected model with dimension $r n_y \times r n_u$ Willcox et al. (2002). To avoid this, block Arnoldi or deflating Arnoldi techniques may be involved. This mechanism aims at detecting rank deflection (see* e.g. *Lehoucq and Sorensen (1996); Yin and Lu (2006); Heyouni and Jbilou (2009)). More recently, Abidi et al. (2017) did suggest a procedure involving a new global rational Arnoldi for model reduction. Still, as it will be made clearer though the manuscript, the tangential Arnoldi framework proposed by Gallivan et al. (2004a) offers an appealing solution to avoid these mechanisms and is thus preferred here (see also Bentbib and Jbilou, 2018).*

Interpolatory conditions and link with projectors

Now, let us come back to the interpolation framework described in Problem 2.3, and provide the main solution to construct V and W, as a bi-tangential Hermite interpolation condition set rather than simple moment matching. First, we consider that the original model $\mathbf{H}$ can be described as a realisation $\mathcal{S}$.

Theorem 2.4: Multi-points bi-tangential Hermite interpolatory conditions

Given an **LTI** dynamical model $\mathcal{S} : (E, A, B, C, D)$ and let $\{\mu_j\}_{j=1}^{q}, \{\lambda_i\}_{i=1}^{k} \in \mathbb{C}$ be such that $(sE - A)$ and $(s\hat{E} - \hat{A})$ are invertible for $s = \mu_j, \lambda_i$, and $\{\mathbf{l}_j\}_{j=1}^{q} \in \mathbb{C}^{n_y}$ and $\{\mathbf{r}_i\}_{i=1}^{k} \in \mathbb{C}^{n_u}$ be fixed nontrivial vectors. Let $V, W \in \mathbb{C}^{n \times r}$ have full-rank, then,

$$\begin{array}{lccccl} \text{(i) if} & (\lambda_i E - A)^{-1} B \mathbf{r}_i & \in \mathbf{span}\,(V), & \text{then} & \mathbf{H}(\lambda_i)\mathbf{r}_i & = & \hat{\mathbf{H}}(\lambda_i)\mathbf{r}_i, \\ \text{(ii) if} & (\mathbf{l}^H C(\mu_j E - A)^{-1})^H & \in \mathbf{span}\,(W), & \text{then} & \mathbf{l}_j^H \mathbf{H}(\mu_j) & = & \mathbf{l}_j^H \hat{\mathbf{H}}(\mu_j), \\ \text{(iii) if} & \text{both (i) and (ii) hold, } r = q = k & & \text{then} & \mathbf{l}_l^H \mathbf{H}'(\sigma_l)\mathbf{r}_l & = & \mathbf{l}_l^H \hat{\mathbf{H}}'(\sigma_l)\mathbf{r}_l, \\ & \{\sigma_l\}_{l=1}^{r} = \{\lambda_i\}_{i=1}^{k} = \{\mu_j\}_{j=1}^{q} & & & & & \end{array}$$

Before getting to the key elements of this model-based interpolatory theorem, let us derive its generalised version, when the original model $\mathbf{H}$ has a generalised coprime form as $\mathbf{H}(s) = \mathcal{C}(s)\mathcal{K}(s)^{-1}\mathcal{B}(s)$.

Theorem 2.5: Multi-points bi-tangential Hermite interpolatory conditions for coprime forms

Given an **LTI** dynamical model with coprime form $\mathbf{H}(s) = \mathcal{C}(s)\mathcal{K}(s)^{-1}\mathcal{B}(s)$ and let $\{\mu_j\}_{j=1}^q, \{\lambda_i\}_{i=1}^k \in \mathbb{C}$ be such that $\mathcal{B}(s)$ $\mathcal{K}(s)$ and $\mathcal{C}(s)$ are analytic for $s = \mu_j, \lambda_i$, and $\{\mathbf{l}_j\}_{j=1}^q \in \mathbb{C}^{n_y}$ and $\{\mathbf{r}_i\}_{i=1}^k \in \mathbb{C}^{n_u}$ be fixed nontrivial vectors. Let $\mathcal{K}(\lambda_i)$, $\mathcal{K}(\mu_j)$, $W^T\mathcal{K}(\lambda_i)V$ and $W^T\mathcal{K}(\mu_j)V$ have full-rank, then,

$$\begin{array}{lllllll}
\text{(i) if} & D_{\lambda_i}\left(\mathcal{K}^{-1}(s)\mathcal{B}(s)\right)\mathbf{r}_i & \in \mathbf{span}\,(V)\,, & \text{then} & \mathbf{H}(\lambda_i)\mathbf{r}_i & = & \hat{\mathbf{H}}(\lambda_i)\mathbf{r}_i, \\
\text{(ii) if} & \left(\mathbf{l}_j^H D_{\mu_j}\left(\mathcal{B}(s)\mathcal{K}^{-1}(s)\right)\right)^H & \in \mathbf{span}\,(W)\,, & \text{then} & \mathbf{l}_j^H\mathbf{H}(\mu_j) & = & \mathbf{l}_j^H\hat{\mathbf{H}}(\mu_j), \\
\text{(iii) if} & \text{both (i) and (ii) hold, } r = q = k & & \text{then} & \mathbf{l}_l^H\mathbf{H}'(\sigma_l)\mathbf{r}_l & = & \mathbf{l}_l^H\hat{\mathbf{H}}'(\sigma_l)\mathbf{r}_l, \\
& \{\sigma_l\}_{l=1}^r = \{\lambda_i\}_{i=1}^k = \{\mu_j\}_{j=1}^q & & & & &
\end{array}$$

where $D_\sigma(\mathbf{f})$ denotes the 0 derivative of the univariate function $\mathbf{f}(s)$, evaluated at $s = \sigma$.

By using the above theorem, one can easily construct a reduced-model satisfying the desired interpolation conditions. For example, for $\mathbf{H}(s) = \mathcal{C}(s)\mathcal{K}(s)^{-1}\mathcal{B}(s)$ and given r interpolation points and tangential directions $\{\sigma_l, \mathbf{l}_l, \mathbf{r}_l\}_{l=1}^r$, by constructing,

$$\begin{array}{rcccl}
\left[\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1)\mathbf{r}_1, \ldots, \mathcal{K}(\sigma_r)^{-1}\mathcal{B}(\sigma_r)\mathbf{r}_r\right] & \subseteq & \mathcal{V} & = & \mathbf{span}\,(V) \\
\left[\mathcal{K}(\sigma_1)^{-H}\mathcal{C}(\sigma_1)^H\mathbf{l}_1, \ldots, \mathcal{K}(\sigma_r)^{-H}\mathcal{C}(\sigma_r)^H\mathbf{l}_r\right]^H & \subseteq & \mathcal{W} & = & \mathbf{span}\,(V)\,,
\end{array}$$

leads to the reduced transfer function with the same coprime structure $\hat{\mathbf{H}}(s) = \hat{\mathcal{C}}(s)\hat{\mathcal{K}}(s)^{-1}\hat{\mathcal{B}}(s)$, which satisfies the bi-tangential Hermite interpolation conditions given in (2.5) and solving Problem 2.3.

Now, by observing the above results, the only remaining questions is the appropriate choice of the $\{\sigma_l, \mathbf{l}_l, \mathbf{r}_l\}_{l=1}^r$ triplet. This is the reason why the $\mathcal{H}_2$ optimal Problem 2.1 now enters into the picture.

2.4.6 Model-based approximation $\mathcal{H}_2$-optimal interpolatory framework

$\mathcal{H}_2$ approximation error and interpolatory conditions

Let us write the $\mathcal{H}_2$ mismatch error as follows,

$$\mathcal{J}_{\mathcal{H}_2}^2(\hat{\mathbf{H}}) = ||\mathbf{H} - \hat{\mathbf{H}}||_{\mathcal{H}_2}^2 = ||\mathbf{H}||_{\mathcal{H}_2}^2 + ||\hat{\mathbf{H}}||_{\mathcal{H}_2}^2 - 2\langle\mathbf{H}, \hat{\mathbf{H}}\rangle_{\mathcal{H}_2} \tag{2.11}$$

Based on Theorems 2.4 and 2.5, and following the contributive works of Gallivan et al. (2004a); Van Dooren et al. (2008); Gugercin et al. (2008), similarly to the Lyapunov and Sylester approach of Wilson (1974), the following theorem and corollary hold.

Theorem 2.6: First-order $\mathcal{H}_2$ optimality conditions

Let $\hat{\mathbf{H}}$ be a r-th order asymptotically stable model with semi-simple poles only, equipped with $\hat{\mathcal{S}} : (\hat{E}, \hat{A}, \hat{B}, \hat{C}, \hat{D})$. If $\hat{\mathbf{H}}$ is solution of the $\mathcal{H}_2$ approximation problem, then

$$\begin{array}{rcl}
\mathbf{H}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{H}}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \\
\hat{\mathbf{c}}_l^H\mathbf{H}(-\hat{\lambda}_l) & = & \hat{\mathbf{c}}_l^H\hat{\mathbf{H}}(-\hat{\lambda}_l) \\
\hat{\mathbf{c}}_i^H\mathbf{H}'(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{c}}_l^H\hat{\mathbf{H}}'(-\hat{\lambda}_l)\hat{\mathbf{b}}_l
\end{array} \tag{2.12}$$

where $[\hat{\mathbf{b}}_1, \ldots, \hat{\mathbf{b}}_r]^H = R\hat{B}$ and $[\hat{\mathbf{c}}_1, \ldots, \hat{\mathbf{c}}_r] = \hat{C}L$ and where L and R are the left and right eigenvectors associated to $\hat{\lambda}_l$, the eigenvalues of $(\hat{E}, \hat{A})$.

First, note that unlike in Theorem 2.1, no structure on $\mathbf{H}$ are required in these optimality conditions. Then, similarly to Wilson (1974)'s approach, Corollary 2.2 shows how constructing the projectors V and W to fulfil any tangential interpolations (see Gugercin et al. (2008); Van Dooren et al. (2008)).

Corollary 2.2: First-order $\mathcal{H}_2$ optimality conditions

Let $V \in \mathbb{C}^{n\times r}$ and $W \in \mathbb{C}^{n\times r}$ be full column rank matrices such that $W^TV = I_r$. Let $\{\hat{\lambda}_l\}_{l=1}^r \in \mathbb{C}$, $\{\hat{\mathbf{b}}_l\}_{l=1}^r \in \mathbb{C}^{n_u}$ and $\{\hat{\mathbf{c}}_l\}_{l=1}^r \in \mathbb{C}^{n_y}$ be given sets of interpolation points and left and right tangential directions, respectively. Assume that points $\hat{\lambda}_l$ are selected such that $(\hat{\lambda}_l E - A)$ are invertible. If,

$$\begin{array}{rcccl} \left[(-\hat{\lambda}_1 E - A)^{-1}B\hat{\mathbf{b}}_1, \ldots, (-\hat{\lambda}_r E - A)^{-1}B\hat{\mathbf{b}}_r\right] & \subseteq & \mathcal{V} & = & \mathbf{span}\,(V) \\ \left[(-\hat{\lambda}_1 E - A)^{-H}C^T\hat{\mathbf{c}}_1, \ldots, (-\hat{\lambda}_r E - A)^{-H}C^T\hat{\mathbf{c}}_r\right] & \subseteq & \mathcal{W} & = & \mathbf{span}\,(W) \end{array} \tag{2.13}$$

then, the projected model $\hat{\mathbf{H}}$, satisfies the tangential interpolation conditions given in (2.12).

Theorem 2.6 provides the first order $\mathcal{H}_2$ optimality conditions and Corollary 2.2, the connection of these latter with the V, W projection matrices (2.13). Note that Corollary 2.2 assumes a first order **DAE** representation.

An algorithm for $\mathcal{H}_2$ approximation

Since these theorems only provide a characterisation and an equivalence of the first-order optimality conditions from a tangential point of view, the challenge consists in creating procedures achieving these conditions. More specifically, based on Corollary 2.2, the $\mathcal{H}_2$ optimal problem consists of finding the triplet $\{\sigma_l, \mathbf{r}_l, \mathbf{l}_l, \}_{l=1}^r = \{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r$ satisfying Theorem 2.6. The most important result and well known algorithm to determine $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r$, is the **IRKA** for Iterative Rational Krylov Algorithm, presented by Gugercin et al. (2008) and recalled in Algorithm 1.

Algorithm 1 MIMO IRKA - **MIMO** Iterative Rational Krylov Algorithm

Require: $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$, $C \in \mathbb{R}^{n_y\times n}$, $\left\{\sigma_l^{(0)}, \mathbf{r}_l^{(0)}, \mathbf{l}_l^{(0)}\right\}_{l=1}^r \in \{\mathbb{C} \times \mathbb{C}^{n_u} \times \mathbb{C}^{n_y}\}$

1: Build $V, W \in \mathbb{R}^{n\times r}$ s.t.

$$\begin{array}{rcl} \left[(\sigma_1^{(0)}E - A)^{-1}B\mathbf{r}_1^{(0)}, \ldots, (\sigma_r^{(0)}E - A)^{-1}B\mathbf{r}_r^{(0)}\right] & = & \mathbf{span}\,(V) \\ \left[(\sigma_1^{(0)}E - A)^{-H}C^T\mathbf{l}_1^{(0)}, \ldots, (\sigma_r^{(0)}E - A)^{-H}C^T\mathbf{l}_r^{(0)}\right] & = & \mathbf{span}\,(W) \end{array}$$

2: Ensure bi-orthogonality: $W \leftarrow W(W^TEV)^{-T}$
3: Set $i = 0$
4: **while** not converged **do**
5: Set $i \leftarrow i + 1$
6: Project the model: $\hat{E} = W^TEV$, $\hat{A} = W^TAV$, $\hat{B} = W^TB$, $\hat{C} = CV$
7: Solve the eigenvalue problem $(\hat{E}, \hat{A})$ and obtain eigenvalues $\{\hat{\lambda}_l\}_{l=1}^r$ and left $L \in \mathbb{C}^{r\times r}$ and right $R \in \mathbb{C}^{r\times r}$ eigenvectors
8: Set $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r = \{\hat{\lambda}_l, (R\hat{B})^H\mathbf{e}_l, \hat{C}L\mathbf{e}_l\}_{l=1}^r$
9: Set $\{\sigma_l^{(i)}, \mathbf{r}_l^{(i)}, \mathbf{l}_l^{(i)}\}_{l=1}^r = \{-\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r$
10: Build $V, W \in \mathbb{R}^{n\times r}$ s.t.

$$\begin{array}{rcl} \left[(\sigma_1^{(i)}E - A)^{-1}B\mathbf{r}_1^{(i)}, \ldots, (\sigma_r^{(i)}E - A)^{-1}B\mathbf{r}_r^{(i)}\right] & = & \mathbf{span}\,(V) \\ \left[(\sigma_1^{(i)}E - A)^{-H}C^T\mathbf{l}_1^{(i)}, \ldots, (\sigma_r^{(i)}E - A)^{-H}C^T\mathbf{l}_r^{(i)}\right] & = & \mathbf{span}\,(W) \end{array}$$

11: Ensure bi-orthogonality: $W \leftarrow W(W^TEV)^{-T}$
12: **end while**
13: Apply projectors V and W and obtain $\hat{\mathbf{H}}$

Ensure: $V, W \in \mathbb{R}^{n\times r}$ and (2.12) are satisfied

The **IRKA** is a fixed point optimisation algorithm which convergence has been proved in some specific cases, only. However, having successfully used it in many academic and industrial applications, it generally converges in few iterations, and is, to author's viewpoint very efficient in practice (see *e.g.* example #1 at the beginning of this chapter). If the algorithm converges, it is not guaranteed to be a local minimum (theoretically it can be a local maxima), but in practice, the $\mathcal{H}_2$-norm of the error generally decreases during the iterations and maximum have been shown to be rejective. Consequently, we can honestly consider that **IRKA** leads to a stationary point of the $\mathcal{H}_2$ approximation problem which is a local minima. Moreover, it is numerically very efficient thanks to Krylov spaces[8] and quite simple to implement, but neither control on the approximation error nor stability preservation are guaranteed. In the following use-case, we illustrates the effectiveness of the **IRKA** procedure, applied on a very-large scale (and unstable) model.

Use-case 1 - Unstable parametric fluid-flow model

This use-case illustrates a very-large scale unstable model approximation of a fluid-flow configuration (see Poussot-Vassal and Sipp, 2015, INTACOO project). We consider a two-dimensional open square cavity flow problem described in Barbagallo et al. (2008), which is used by engineers in aeronautics to study some configurations *e.g.* in flight landing (left frame of Figure 2.8). For simulation, the mesh used is composed of $193,874$ triangles, corresponding to $n = 680,974$ degrees of freedom for the velocity variables along the x and y axis. After linearisation around five fixed points for varying Reynolds numbers $Re = \{4000, 5250, 6000, 7500, 10000\}$, and discretisation along the flow axis, five dynamical models $\{\mathbf{H}_i\}_{i=1}^5$ can be described as a state-space realisation of order $n = 680,974$ where input $\mathbf{u}(t)$ is the vertical pressure actuator located upstream of the cavity and the output $\mathbf{y}(t)$ is a shear stress sensor, located downstream of the cavity. By approximating each model with an order $r = 18$, using **IRKA** from Gugercin et al. (2008) at the different Re values, one obtains the following frequency responses (right frame of Figure 2.8)[a].

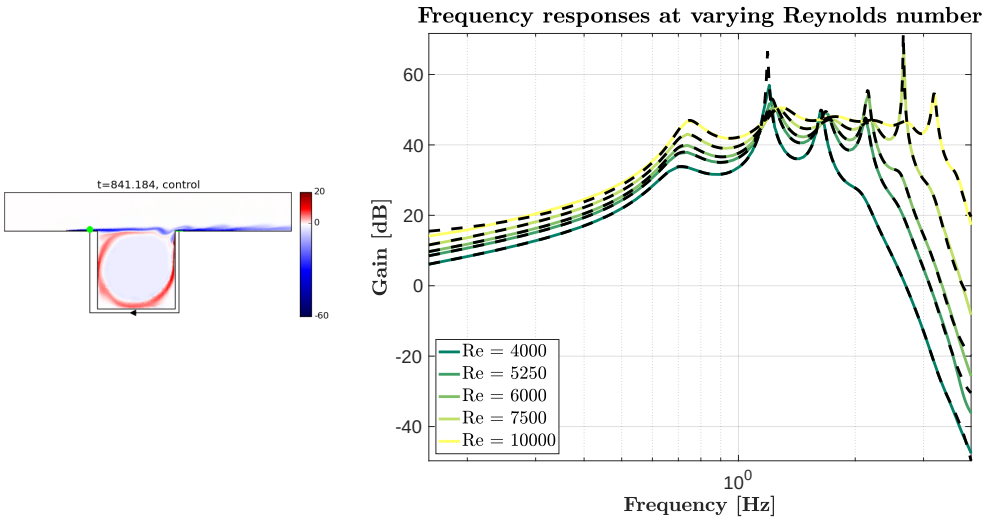


Figure 2.8: Left: Navier-Stokes based fluid flow open cavity setting. Right: frequency responses of the open cavity model and its approximation at varying Reynolds number configurations. Original large-scale model with $n = 680,974$ (solid coloured lines), reduced order with $r = 18$ (black dashed).

Figure 2.9 illustrates the eigenvalues of the reduced order models $\{\hat{\mathbf{H}}_i\}_{i=1}^5$. Interestingly, the unstable configurations are restituted by the reduced models obtained with **IRKA** although it is an $\mathcal{H}_2$-oriented algorithm. Indeed, the cases where the reduced order models are unstable is coherent with what fluid mechanics experts were expecting. This property has also been exploited later in Pontes Duff et al. (2015b), to analyse the stability of a large-scale model and is the ground idea exposed in Chapter 5.

[8] See also Poussot-Vassal and Sipp (2015) for application of **IRKA** on an (unstable) model of dimension $n = 650,000$.

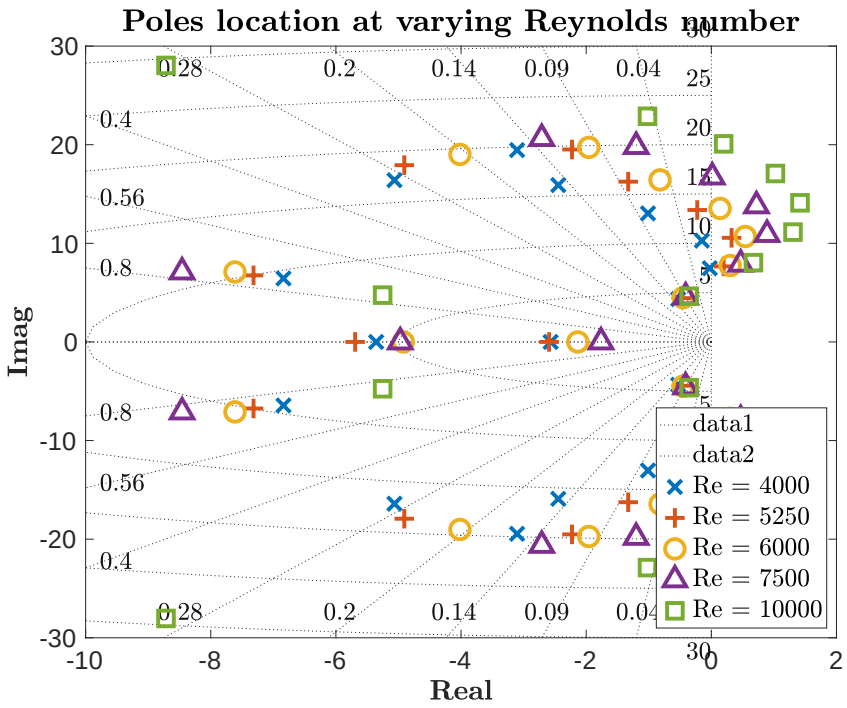


Figure 2.9: Eigenvalues of the reduced order model for varying Reynolds number configurations.

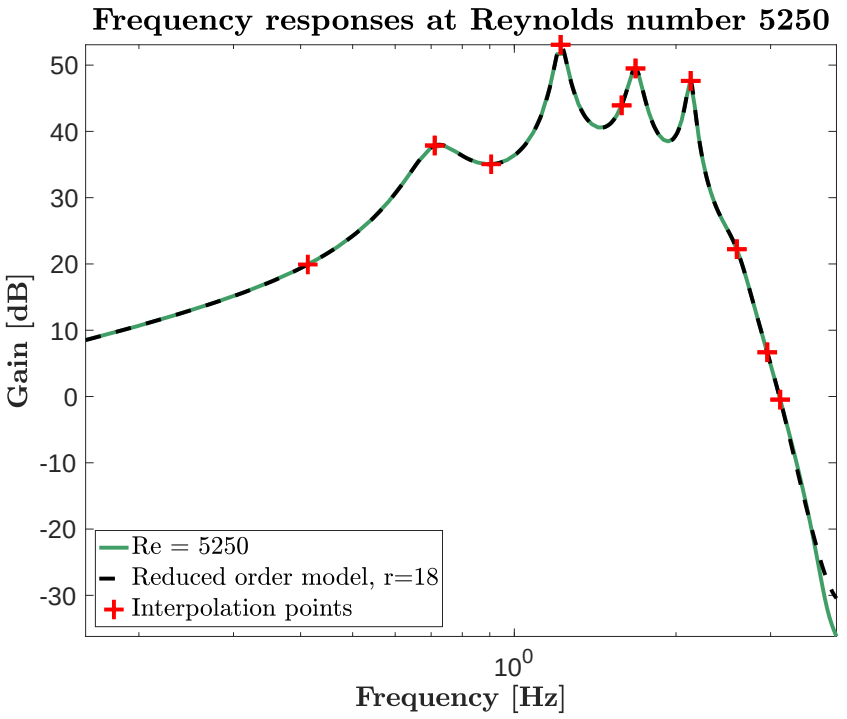


Figure 2.10: Frequency responses of the open cavity model at Reynolds number 5250. Original large-scale model with $n = 680,974$ (solid coloured line), reduced order with $r = 18$ (black dashed line) and evaluation of $\hat{\mathbf{H}}(|\sigma_l|)$ the reduced order model at the optimal interpolatory points (red crosses).

Then, in Poussot-Vassal and Sipp (2015), these reduced models are used to generate a parametric reduced order model by interpolating the realisations coefficients (in the canonical bais). Finally, an other interesting point is illustrated on Figure 2.10: the red crosses represents the frequency responses of $\mathbf{H}$ are the optimal shift points $\{\sigma_l\}_{l=1}^{r=18}$ absolute values. One may observe that these shift points are representative of the frequencies where the largest pick values appear. This might also be a nice angle of attack for treating $\mathcal{H}_\infty$-norm computation of large-scale systems (see also Vuillemin et al. (2014c) and perspectives in Chapter 7).

[a] Note that for each model, the approximated model has been obtained in only 1h on a standard laptop.

Later in Beattie and Gugercin (2009a), authors extended it to **MIMO** systems, associated with a complex-domain trust region algorithm guaranteeing $\mathcal{H}_2$ mismatch error monotonic descent and stability preservation. Still author believe that this improvement is more methodological rather that practical since, at the price of some numerical manipulation, it is always possible to preserve stability (see *e.g.* Kohler (2014), which provides a way to find an optimal $\mathcal{RH}_\infty$ approximant). Moreover, in Gugercin (2008) then Poussot-Vassal (2011), the stability preservation can be guaranteed in exchange for a loss of accuracy, or at least a loss of optimality conditions, by involving a single gramian computation in the process instead of a Krylov subspace. This slight modification is gathered in the following theorem for the case of the observability gramian. Note that, similarly to the **IRKA** procedure, an algorithm denoted as **ISTIA** for Iterative SVD tangential Interpolation Algorithm is defined first in Gugercin (2008) in the **SIMO** and **MISO** case, and in Poussot-Vassal (2011) in the **MIMO** one, involving tangential interpolatory directions.

Theorem 2.7: SVD-Tangential interpolation

Let $V \in \mathbb{C}^{n\times r}$ and $W \in \mathbb{C}^{n\times r}$ be full rank matrices such that $W^TV = I_r$. Let $\{\sigma_l, \mathbf{r}_l\}_{l=1}^r \in \{\mathbb{C} \times \mathbb{C}^{n_u}\}$ be a given set of interpolation points and right tangential directions, respectively. Assume that points σ_l are selected such that $(\sigma_l E - A)$ are invertible. If,

$$\begin{array}{rcccc} \left[(\sigma_1 E - A)^{-1}B\mathbf{r}_1, \ldots, (\sigma_r E - A)^{-1}B\mathbf{r}_r\right] & \subseteq & \mathcal{V} & = & \mathbf{span}\,(V) \\ A^T\mathcal{Q} + \mathcal{Q}A + C^TC & = & 0 & & \\ \mathcal{Q}V(V^TE\mathcal{Q}V)^{-1} & \subseteq & \mathcal{W} & = & \mathbf{span}\,(W) \end{array}$$

then, the reduced order system $\hat{\mathbf{H}}$, satisfies the following conditions, for $l = 1, \ldots r$,

$$\begin{array}{c} \mathbf{H}(\sigma_l)\mathbf{r}_l = \hat{\mathbf{H}}(\sigma_l)\mathbf{r}_l \\ (\hat{E}, \hat{A}) \text{ pencil is stable} \end{array}.$$

The above theorem then simply provides a framework that trades optimality with stability preservation. Note that this theorem will be the basis for the first frequency-limited procedure proposed in Chapter 3.

Back to Theorem 2.6, it is now clear that the interpolatory conditions are suitable for any original model $\mathbf{H}$. As a consequence, Beattie and Gugercin (2012) did extend it to the realisation-free case, leading to the Algorithm 2. This latter now involves the Loewner and shifted Loewner matrices $\mathbb{L}$ and $\mathbb{L}_\sigma$ given in (2.15) and detailed in the next subsection, instead of the Krylov subspaces.

To the author's knowledge, it is the first method allowing approximating any meromorphic transfer $\mathbf{H}$ with a finite order model $\hat{\mathbf{H}}$ equipped with a realisation $\hat{\mathcal{S}}$ while ensuring first order $\mathcal{H}_2$ optimality conditions. This algorithm has been exploited in the example for use-case #2 at the beginning of this chapter. Author also believes that, from a didactical viewpoint, it is also a fancy way to link Loewner matrices, generalised observability and controllability gramians and generalised rational Krylov subspaces. Therefore, it can clearly be used by an unfamiliar reader before getting into data-driven model approximation.

Algorithm 2 MIMO TF-IRKA - **MIMO** Transfer Function Iterative Rational Krylov Algorithm

Require: Transfer function $\mathbf{H}(s)$ and its derivative $\mathbf{H}'(s)$, an objective order r, $\{\sigma_l^{(0)}, \mathbf{r}_l^{(0)}, \mathbf{l}_l^{(0)}\}_{l=1}^r \in \{\mathbb{C} \times \mathbb{C}^{n_u} \times \mathbb{C}^{n_y}\}$

1: Set $i = 0$
2: **while** not convergence **do**
3: Set $i \leftarrow i + 1$
4: Build $\hat{E}$, $\hat{A}$, $\hat{B}$ and $\hat{C}$ matrices using the $\mathbb{L}$ and $\mathbb{L}_\sigma$ Loewner matrices as in (2.15)
5: Solve the eigenvalue problem $(\hat{E}, \hat{A})$ and obtain eigenvalues $\{\hat{\lambda}_l\}_{l=1}^r$ and left $L \in \mathbb{C}^{r\times r}$ and right $R \in \mathbb{C}^{r\times r}$ eigenvectors
6: Set $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r = \{\hat{\lambda}_l, (R\hat{B})^H \mathbf{e}_l, \hat{C}L\mathbf{e}_l\}_{l=1}^r$
7: Set $\{\sigma_l^{(i)}, \mathbf{r}_l^{(i)}, \mathbf{l}_l^{(i)}\}_{l=1}^r = \{-\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r$
8: **end while**
9: Build $\hat{E}$, $\hat{A}$, $\hat{B}$, $\hat{C}$ and obtain $\hat{\mathbf{H}}$.

Ensure: (2.12) are satisfied

$\mathcal{H}_2$ optimality conditions with respect to general parametrisation

Before closing the model-based interpolatory conditions section, in Pontes (2017), author also extended these optimality conditions to a more general parametrisation of the solution $\hat{\mathbf{H}}$. Indeed, in the above theorem, both the original and reduced-order model share the same pole residues structure given as

$$\mathbf{H}(s) = \sum_{j=1}^{n} \frac{\mathbf{c}_j \mathbf{b}_j^T}{s - \lambda_j} \quad \text{and} \quad \hat{\mathbf{H}}(s) = \sum_{k=1}^{r} \frac{\hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T}{s - \hat{\lambda}_k}.$$

Instead of this assumption, let us now suppose that $\hat{\mathbf{H}}$ depends on a vector of parameters $\hat{\mathbf{p}} = \{\hat{p}_l\}_{l=1}^q \in \mathbb{C}$. Then, the gradient of $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}})$ with respect to $\hat{\mathbf{p}}$ (or $\mathcal{J}^2_{\mathcal{H}_2}(\hat{\mathbf{H}}(\hat{\mathbf{p}}))$) is given by,

$$\begin{aligned} \nabla_{\hat{\mathbf{p}}} ||\mathbf{H} - \hat{\mathbf{H}}||^2_{\mathcal{H}_2} &= \nabla_{\hat{\mathbf{p}}} \langle \mathbf{H} - \hat{\mathbf{H}}, \mathbf{H} - \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} \\ &= 2\langle \mathbf{H} - \hat{\mathbf{H}}, \nabla_{\hat{\mathbf{p}}} (\mathbf{H} - \hat{\mathbf{H}}) \rangle_{\mathcal{H}_2} \\ &= -2\langle \mathbf{H} - \hat{\mathbf{H}}, \nabla_{\hat{\mathbf{p}}} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} \end{aligned}$$

where $\langle \mathbf{H} - \hat{\mathbf{H}}, \nabla_{\hat{\mathbf{p}}} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2}$ stands for

$$\langle \mathbf{H} - \hat{\mathbf{H}}, \nabla_{\hat{\mathbf{p}}} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} = \left[\langle \mathbf{H} - \hat{\mathbf{H}}, \frac{\partial}{\partial \hat{p}_1} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} \quad \langle \mathbf{H} - \hat{\mathbf{H}}, \frac{\partial}{\partial \hat{p}_2} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} \quad \dots \quad \langle \mathbf{H} - \hat{\mathbf{H}}, \frac{\partial}{\partial \hat{p}_q} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} \right].$$

Thus, the local necessary optimisation conditions with respect to $\hat{\mathbf{p}} = [\hat{p}_1 \quad \hat{p}_2 \dots \hat{p}_q]$ are given by

$$\nabla_{\hat{\mathbf{p}}} ||\mathbf{H} - \hat{\mathbf{H}}||^2_{\mathcal{H}_2} = 0,$$

and the following result holds.

Theorem 2.8: $\mathcal{H}_2$ necessary optimality condition with respect to a parameter

Given $\mathbf{H} \in \mathcal{H}_2$ and let us consider for each parameter $\hat{\mathbf{p}} = \{\hat{p}_l\}_{l=1}^q \in \mathbb{C}$ a model $\hat{\mathbf{H}}(\hat{\mathbf{p}})$ and the function that associates $\hat{\mathbf{p}} \to \hat{\mathbf{H}}(\hat{\mathbf{p}})$ is differentiable. Then, if $\mathbf{p}^\star$ is the minimiser of $||\mathbf{H} - \hat{\mathbf{H}}(\hat{\mathbf{p}})||_{\mathcal{H}_2}$, it satisfies

$$\left\langle \mathbf{H}, \nabla_{\hat{\mathbf{p}}} \hat{\mathbf{H}} \Big|_{\hat{\mathbf{p}}=\mathbf{p}^\star} \right\rangle_{\mathcal{H}_2} = \left\langle \hat{\mathbf{H}}(\mathbf{p}^\star), \nabla_{\hat{\mathbf{p}}} \hat{\mathbf{H}} \Big|_{\hat{\mathbf{p}}=\mathbf{p}^\star} \right\rangle_{\mathcal{H}_2}.$$

Theorem 2.8 enables deriving the optimality conditions for a general parameter $\hat{\mathbf{p}}$. Notice that here, no structure was pre-supposed on $\mathbf{H}$. Thus, it enables obtaining the $\mathcal{H}_2$ optimality conditions in a simpler

way. As an illustration, let us assume that ($q = 3r$) and

$$\hat{\mathbf{H}}(s) = \sum_{l=1}^{r} \frac{\hat{\mathbf{c}}_l \hat{\mathbf{b}}_l^T}{s - \hat{\lambda}_l} \quad \text{and} \quad \hat{\mathbf{p}} = [\hat{\mathbf{b}}_1 \dots \hat{\mathbf{b}}_r \quad \hat{\mathbf{c}}_1 \dots \hat{\mathbf{c}}_r \quad \hat{\lambda}_1 \dots \hat{\lambda}_r].$$

Thus, if one computes the gradient of $\hat{\mathbf{H}}$ with respect to $\hat{\mathbf{b}}_l$, we have

$$\begin{aligned} \nabla_{\hat{\mathbf{b}}_l} \hat{\mathbf{H}} &= \nabla_{\hat{\mathbf{b}}_l} \left(\frac{\hat{\mathbf{c}}_l \hat{\mathbf{b}}_l^T}{s - \hat{\lambda}_l} \right) \\ &= \left[\frac{\hat{\mathbf{c}}_l \mathbf{e}_1^T}{s - \hat{\lambda}_l} \quad \frac{\hat{\mathbf{c}}_l \mathbf{e}_2^T}{s - \hat{\lambda}_l} \cdots \frac{\hat{\mathbf{c}}_l \mathbf{e}_{n_u}^T}{s - \hat{\lambda}_l} \right] \end{aligned}$$

Hence, by writing the first order optimality conditions with respect to $\hat{\mathbf{b}}_l$ by invoking Theorem 2.8, one obtains

$$\begin{aligned} & \langle \mathbf{H}, \nabla_{\hat{\mathbf{b}}_l} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} = \langle \hat{\mathbf{H}}, \nabla_{\hat{\mathbf{b}}_l} \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} \\ \Leftrightarrow \quad & \langle \mathbf{H}, \frac{\hat{\mathbf{c}}_l \mathbf{e}_i^T}{s - \hat{\lambda}_l} \rangle_{\mathcal{H}_2} = \langle \hat{\mathbf{H}}, \frac{\hat{\mathbf{c}}_l \mathbf{e}_i^T}{s - \hat{\lambda}_l} \rangle_{\mathcal{H}_2} \quad \text{for } i = 1, \dots, n_u \\ \Leftrightarrow \quad & \mathbf{c}_l^T \mathbf{H}(-\hat{\lambda}_l) \mathbf{e}_i = \mathbf{c}_l^T \hat{\mathbf{H}}(-\hat{\lambda}_l) \mathbf{e}_i \quad \text{for } i = 1, \dots, n_u \\ \Leftrightarrow \quad & \mathbf{c}_l^T \mathbf{H}(-\hat{\lambda}_l) = \mathbf{c}_l^T \hat{\mathbf{H}}(-\hat{\lambda}_l). \end{aligned}$$

This last term recovers the second equation of the first order $\mathcal{H}_2$ interpolation conditions of Theorem 2.6, and given by (2.12). In the Ph.D. manuscript of Pontes (2017), author derived the same expression with respect to $\hat{\mathbf{c}}_l$ and $\hat{\lambda}_l$, recovering all the three expressions of (2.12). Interestingly, this formulation then opens to new extended $\mathcal{H}_2$ interpolatory conditions.

Before moving the data-driven case concerning the use-case #3, let us close this part with an illustration of the application of interpolatory techniques to construct a parametric model, applied on an industrial use-case provided by **Airbus** and exposed hereafter. This use-case also illustrates the flexibility embedded in the interpolatory framework, thanks to the smart shift points σ_l selection.

Use-case 2 - Airbus long range aircraft

This use-case comes from an **Airbus** long range aircraft model set, for which a parametric reduced-order model construction was intended (see Poussot-Vassal and Roos, 2012, Clean Sky 1 - SFWA project). Here $\{\mathbf{H}_i\}_{i=1}^{n_s=9}$, a family of nine models have been considered. Each of these models represent a long range aircraft model linearised at three given center tank $\mathbf{CT}$ and outer tank $\mathbf{OT}$ filling configurations ($n_s = 3 \times 3$). Each of them resulting in a medium scale model of order around $n_i \approx 500$. The objective, in view of robust and parametric controller design, was to construct a reduced order model $\hat{\mathbf{H}}(s, \mathbf{OT}, \mathbf{CT})$, where $\mathbf{OT}$ and $\mathbf{CT}$ stand as the outer and center tank filling coefficient. In Poussot-Vassal and Roos (2012), authors propose a simple approach based on reduction using **ISTIA** followed by a common base change (here the eigenvector one) and interpolation of the realisation's coefficients. One difficulty on the interpolation in the eigenvalue basis is that the eigen-modes should be "close" to each other to simplify the rational interpolation. To do so authors did compare the balanced truncation **BT** with the **ISTIA** and multi-**ISTIA** interpolation methods. The latter trades the $\mathcal{H}_2$ partial interpolatory conditions with a modal content "easy" to interpolate. This is illustrated on Figure 2.11 where eigenvalues of the approximated models using different methods is shown.

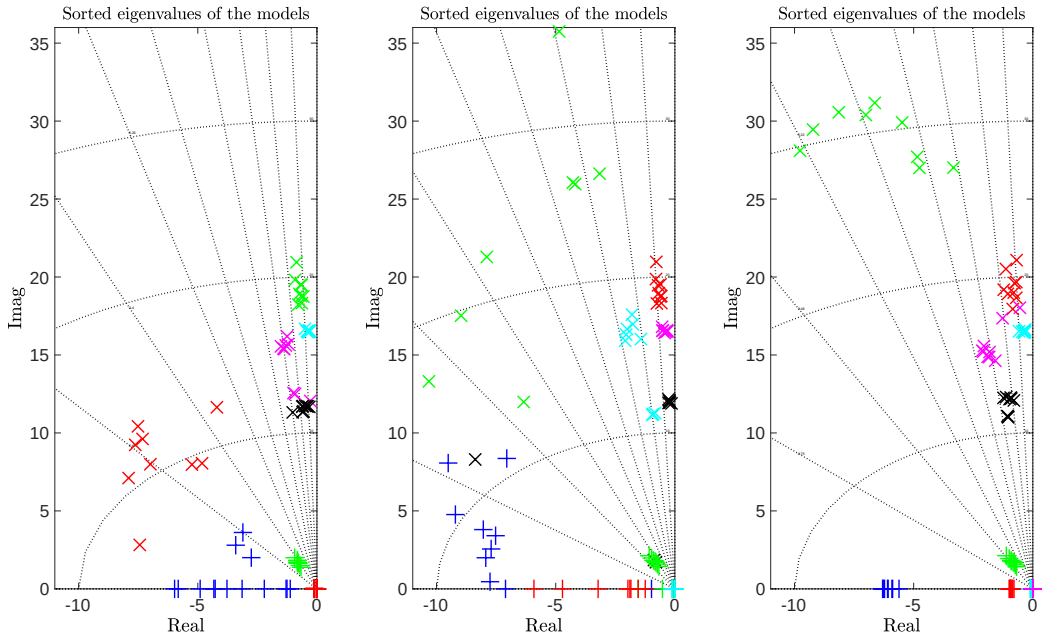


Figure 2.11: Eigenvalues of the approximated models using balanced truncation (left), **ISTIA** (middle) and multi-**ISTIA** (right).

In this example, the interest of the interpolatory methods with respect to the balanced ones is that it enables to fix some shift points locations and tangential directions to enforce the eigenvalues of the reduced order models $\{\hat{\mathbf{H}}_i\}_{i=1}^{n_s=9}$ to be close to each other, simplifying then the generation of the parametric model $\hat{\mathbf{H}}(s, \mathbf{OT}, \mathbf{CT})$ (see Poussot-Vassal and Roos, 2012, for details).

This last observation and use-case closes the brief overview of the model-based model approximation, viewed from the interpolatory framework angle. Let us now move to the data-driven approximation by interpolation.

2.4.7 The data-driven approximation method by interpolation

Forewords on the tangential interpolatory framework

It is now clear that tangential interpolation is a flexible framework for model reduction. Indeed, with reference to Problem 2.3, note that if $\mathbf{w}_i = \mathbf{H}(\lambda_i)\mathbf{r}_i$, then $e^{\lambda_i t}\mathbf{w}_i$ is precisely the response of the full order model to a pure input given by $\mathbf{u}(t) = e^{\lambda_i t}\mathbf{r}_i$, then the tangential interpolation conditions that characterise $\hat{\mathbf{H}}$ could (at least in principle) be obtained from measured input-output data drawn directly from observations on the original system. For example, if $\lambda_i = \imath\omega_0$, one is observing in $\mathbf{w}_i$ the sinusoidal response of the system to a pure tone input of pulsation ω_0. Similarly, the dual dynamical model also holds true. Then, the alternative Problem 2.4 that we consider is entirely based on observed input-output response data, and no other a priori information about the system (as defined in previous sub-sections through $\mathbf{H}$), is required. Note that a tutorial on this data-driven method is given in Antoulas et al. (2016).

> The main ingredient of the data-driven interpolation approach is the Loewner matrix, which was developed in a series of papers (see *e.g.* Mayo and Antoulas (2007) for its first application the derivation of a state-space model). In the sequel, we provide a quick overview of the Loewner framework, in the **MIMO** case. Author is convinced that this framework stands as a very strong and powerful one for many applications. This assertion is specifically illustrated in Chapters 5 and 7.

The Loewner framework

Following the same notations as before, let us be given respectively, the left or row data and the right or column data:

$$\left.\begin{array}{c}(\mu_j, \mathbf{l}_j^T, \mathbf{v}_j^T)\\ \text{for } j = 1, \ldots, q\end{array}\right\} \text{ and } \left\{\begin{array}{c}(\lambda_i, \mathbf{r}_i, \mathbf{w}_i)\\ \text{for } i = 1, \ldots, k\end{array}\right.$$

Let us assume that λ_i and μ_j are distinct and that the left and right interpolation data are organised as:

$$\left.\begin{array}{rcll}\mathbf{M} &=& \mathbf{diag}(\mu_1, \ldots, \mu_q) & \in \mathbb{C}^{q\times q}\\ \mathbf{L}^T &=& [\mathbf{l}_1 \ \ldots \ \mathbf{l}_q] & \in \mathbb{C}^{n_y\times q}\\ \mathbf{V}^T &=& [\mathbf{v}_1 \ \ldots \ \mathbf{v}_q] & \in \mathbb{C}^{n_u\times q}\end{array}\right\} \text{ and } \left\{\begin{array}{rcll}\mathbf{\Lambda} &=& \mathbf{diag}(\lambda_1, \ldots, \lambda_k) & \in \mathbb{C}^{k\times k}\\ \mathbf{R} &=& [\mathbf{r}_1 \ \ldots \ \mathbf{r}_k] & \in \mathbb{C}^{n_u\times k}\\ \mathbf{W} &=& [\mathbf{w}_1 \ \ldots \ \mathbf{w}_k] & \in \mathbb{C}^{n_y\times k}\end{array}\right. \tag{2.14}$$

The associated Loewner $\mathbb{L} \in \mathbb{C}^{q\times k}$ and shifted Loewner $\mathbb{L}_\sigma \in \mathbb{C}^{q\times k}$ matrices, also referred to as the Loewner pencil, are constructed as follows, for $i = 1, \ldots, k$ and $j = 1, \ldots, q$:

$$\begin{array}{rcccl}[\mathbb{L}]_{j,i} &=& \dfrac{\mathbf{v}_j^T\mathbf{r}_i - \mathbf{l}_j^T\mathbf{w}_i}{\mu_j - \lambda_i} &=& \dfrac{\mathbf{l}_j^T\big(\mathbf{H}(\mu_j) - \mathbf{H}(\lambda_i)\big)\mathbf{r}_i}{\mu_j - \lambda_i}\\ [\mathbb{L}_\sigma]_{j,i} &=& \dfrac{\mu_j\mathbf{v}_j^T\mathbf{r}_i - \lambda_i\mathbf{l}_j^T\mathbf{w}_i}{\mu_j - \lambda_i} &=& \dfrac{\mathbf{l}_j^T\big(\mu_j\mathbf{H}(\mu_j) - \lambda_i\mathbf{H}(\lambda_i)\big)\mathbf{r}_i}{\mu_j - \lambda_i}\end{array}. \tag{2.15}$$

Interestingly, the above Loewner $\mathbb{L}$ and shifted Loewner $\mathbb{L}_\sigma$ matrices satisfy the following Sylvester equations:

$$\mathbb{L}\mathbf{\Lambda} - \mathbf{M}\mathbb{L} = \mathbf{LW} - \mathbf{VR},$$

and

$$\mathbb{L}_\sigma\mathbf{\Lambda} - \mathbf{M}\mathbb{L}_\sigma = \mathbf{LW}\mathbf{\Lambda} - \mathbf{MVR}.$$

Then, the following lemma, which proof is given in Mayo and Antoulas (2007), provides one of the most important property of the Loewner matrices.

Lemma 2.1: Loewner matrix McMillian degree and rank

Given (tangential) samples of a rational function defined in terms of a minimal descriptor realisation through its transfer function $\mathbf{H}_\delta(s) = C_\delta(sE_\delta - A_\delta)^{-1}B_\delta$, construct the associated Loewner $\mathbb{L}$ and shifted Loewner $\mathbb{L}_\sigma$ matrices. Assuming that we have enough samples, and that the left, right tangential directions $\mathbf{l}_j$ and $\mathbf{r}_i$ are chosen so that no rank loss is observed, the following holds:

$$\mathbf{rank}\,\mathbb{L} = \mathbf{rank}\,E_\delta = n$$

where n is the McMillian degree of the underlying rational function $\mathbf{H}_\delta(s)$. Moreover,

$$\mathbf{rank}\,\mathbb{L}_\sigma = \mathbf{rank}\,A_\delta$$

where $E_\delta, A_\delta \in \mathbb{R}^{n\times n}$.

As rooted on the Lemma 2.1, one is now ready to state the main result concerning the construction of interpolants using the Loewner pencil. The following theorem summaries the interpolatory model construction from data.

Theorem 2.9: Data-driven interpolation though the Loewner framework

Given right and left interpolation data as in (2.14), and assume that $k = q$ and let $(\mathbb{L}, \mathbb{L}_\sigma)$ be a regular pencil where λ_i or μ_j are not eigenvalues. The rational transfer function $\mathbf{H}_\delta(s) = C_\delta(sE_\delta - A_\delta)^{-1}B_\delta$, with realisation $\mathcal{S} : (E_\delta, A_\delta, B_\delta, C_\delta, 0)$ constructed as

$$E_\delta = -\mathbb{L},\ A_\delta = -\mathbb{L}_\sigma,\ B_\delta = \mathbf{V} \quad \text{and} \quad C_\delta = \mathbf{W},$$

is a minimal descriptor realisation of an interpolant of the data, *i.e.*

$$\mathbf{H}_\delta(s) = \mathbf{W}(\mathbb{L}_\sigma - s\mathbb{L})^{-1}\mathbf{V}$$

satisfies the left and right interpolation constraints (2.6).

If now we consider the case where more data than absolutely necessary is provided, which is realistic and indeed practically the case in many applications[9]. In this case, the problem has a solution provided that, for all $\alpha \in \{\lambda_i\} \cup \{\mu_j\}$,

$$\mathbf{rank}(\alpha\mathbb{L} - \mathbb{L}_\sigma) = \mathbf{rank}\begin{bmatrix} \mathbb{L} & \mathbb{L}_\sigma \end{bmatrix} = \mathbf{rank}\begin{bmatrix} \mathbb{L} \\ \mathbb{L}_\sigma \end{bmatrix} = n^\star$$

A minimal realisation of an interpolant of the data is given by the system:

$$\mathcal{S} : \begin{cases} -\mathbf{Y}^T\mathbb{L}\mathbf{X}\dot{\mathbf{x}}(t) &= -\mathbf{Y}^T\mathbb{L}_\sigma\mathbf{X}\mathbf{x}(t) + \mathbf{Y}^T\mathbf{V}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{W}\mathbf{X}\mathbf{x}(t) + D\mathbf{u}(t) \end{cases}$$

where $\mathbf{X} \in \mathbb{C}^{k\times n^\star}$ and $\mathbf{Y} \in \mathbb{C}^{k\times n^\star}$ are the orthogonal factors of the short **SVD** factorisation of (where $\Sigma \in \mathbb{C}^{n^\star\times n^\star}$):

$$\alpha\mathbb{L} - \mathbb{L}_\sigma = \mathbf{Y}\Sigma\mathbf{X}^T.$$

In brief, the rank of the Loewner matrices encodes the McMillian degree of a realisation. From the author's feeling, this last property is actually a one, in control theory, but also in complex functions analysis. It will be extensively invoked in Chapter 5, for the analysis of meromorphic functions.

[9] Note that in this case we do not intend the noisy data one, which is treated in an other way (see *e.g.* Kergus et al. (2018))

Theorem 2.10: Data-driven approximation though the Loewner framework

The quintuple $(E, A, B, C, 0)$, given by:

$$E = -\mathbf{Y}^T\mathbb{L}\mathbf{X},\ A = -\mathbf{Y}^T\mathbb{L}_\sigma\mathbf{X},\ B = \mathbf{Y}^T\mathbf{V} \text{ and } C = \mathbf{W}\mathbf{X},$$

is a descriptor realization of an (approximate) interpolant of the data with McMillan degree $n^\star = \mathbf{rank}\,\mathbb{L}$.

Now the interpolatory framework, connected to the Loewner one has been clarified, it is obvious that, starting from data either obtained from experiments or simulations, it is simple to construct a rational interpolant $\hat{\mathcal{S}}$ and its associated (approximate) transfer function $\hat{\mathbf{H}}(s)$ with a user defined complexity. Starting from the left and right interpolation data (2.14) organised as:

$$\left.\begin{array}{rcll} \mathbf{M} & = & \mathbf{diag}(\mu_1, \ldots, \mu_q) & \in \mathbb{C}^{q\times q} \\ \mathbf{L}^T & = & [\mathbf{l}_1\ \ldots\ \mathbf{l}_q] & \in \mathbb{C}^{n_y\times q} \\ \mathbf{V}^T & = & [\mathbf{v}_1\ \ldots\ \mathbf{v}_q] & \in \mathbb{C}^{n_u\times q} \end{array}\right\} \text{ and } \left\{\begin{array}{rcll} \mathbf{\Lambda} & = & \mathbf{diag}(\lambda_1, \ldots, \lambda_k) & \in \mathbb{C}^{k\times k} \\ \mathbf{R} & = & [\mathbf{r}_1\ \ldots\ \mathbf{r}_k] & \in \mathbb{C}^{n_u\times k} \\ \mathbf{W} & = & [\mathbf{w}_1\ \ldots\ \mathbf{w}_k] & \in \mathbb{C}^{n_y\times k} \end{array}\right.$$

and by applying a rank revealing factorisation (such as the **SVD**):

$$\mathbb{L} = \left[\begin{array}{cc} \mathbf{Y}_1 & \mathbf{Y}_2 \end{array}\right] \left[\begin{array}{cc} \Sigma_1 & \\ & \Sigma_2 \end{array}\right] \left[\begin{array}{c} \mathbf{X}_1^T \\ \mathbf{X}_2^T \end{array}\right],$$

where $\Sigma_1 \in \mathbb{R}^{r\times r}$, $\Sigma_2 \in \mathbb{R}^{(n-r)\times(n-r)}$ and $\mathbf{Y}_1$, $\mathbf{Y}_2$, $\mathbf{X}_1$, $\mathbf{X}_2$ are of appropriate dimensions, the reduced order model $\hat{\mathbf{H}}$ with realisation $\hat{\mathcal{S}}$ is simply obtained by the Petrov-Galerkin projection:

$$\begin{array}{rcl} \hat{\mathcal{S}} & : & (\hat{E}, \hat{A}, \hat{B}, \hat{C}, 0) \\ & : & (-\mathbf{Y}_1^T\mathbb{L}\mathbf{X}_1, -\mathbf{Y}_1^T\mathbb{L}_\sigma\mathbf{X}_1, \mathbf{Y}_1^T\mathbf{V}, \mathbf{W}\mathbf{X}_1, 0). \end{array} \tag{2.16}$$

Consequently, one can construct a model $\hat{\mathcal{S}} : (\hat{E}, \hat{A}, \hat{B}, \hat{C}, 0)$ with transfer function $\hat{\mathbf{H}}(s)$ that has any degree of $r \leq n^\star$, the minimal McMillian degree of the data-based linear model, and which tangentially interpolates the data with a given error level (see more in detailed survey by Antoulas et al., 2016).

Remark 2.7 (About optimality conditions) *The truncation step performed when applying the **SVD** in* (2.16) *does not guarantee the so-called $\mathcal{H}_2$-optimality conditions of the projected model as in* e.g. *Gugercin et al. (2008). These conditions can be achieved through an adequate interpolation data μ_j and λ_i selection, as explained in Drmac et al. (2015b,a).*

Some extensions of this framework were developed in he recent research works. Here is a non-exhaustive list of extensions and relative works:

- Parametric systems: in Ionita (2013); Ionita and Antoulas (2014), authors extend the Loewner framework to parametric systems. This framework then allows treating parametrised data and transfer functions.
- Time-delay systems: in Pontes Duff et al. (2015a, 2016a) and Schulze and Unger (2016), the Loewner framework for a special class of time-delay systems is presented. See also the Ph.D. manuscript of Pontes (2017) for an insight on approximation with time delayed structures.
- Structured realisations: in Schulze et al. (2018), authors extend the Loewner framework to the general transfer with coprime structure, illustrated on time-delayed systems, but which provides a very exciting perspectives for this method class.

Before closing the data-driven part, let us provide a use-case extracted from a project conducted in collaboration with **Dassault-Aviation** and which considered the aircraft anti-vibration control validation on a ground test setup. It illustrates the efficiency of the Loewner framework in an experimental industrial context.

Use-case 3 - Dassault-Aviation business jet ground test

This use-case illustrates the **Dassault-Aviation** business jet ground vibration tests, performed at Istres, France (see Meyer et al., 2016, 2017, Clean Sky 1 - SFWA project). After having designed an anti-vibration controller[a] aiming at attenuating the amplifications at the cabin and passenger levels in response to aerodynamics turbulences occurring within 7 and 10Hz, **Dassault-Aviation** engineers did implement the control law on the real Business Jet aircraft, as shown on Figure 2.12 (left). Then, using shakers on the horizontal tail of the aircraft, it is possible to excite the structure, which simulates some aerodynamical disturbances. Using the kinematic effects of the movable control surfaces only, the objective was to validate the control design methodology on the Falcon 7X. To this end, sensors were installed over the aircraft (right frame of Figure 2.12) and data collected for a sine chirp excitation.

Figure 2.12: Left: Ground vibration experimental test-bed. Right: illustration of the sensors location.

In practice, **Dassault-Aviation** aeroelastic and tests engineers did collect the frequency-domain data (along among 100 sensors) and fed the **MOR** Toolbox as follows:

```
optLoe              = [];
red                 = 30;
optLoe.verbose      = true;
optLoe.extraInfo    = true;
optLoe.ensureStab   = true;     % Ensure stability of the reduced order model
optLoe.ioNormalize  = true;     % Normalize input-output transfer to catch all dynamics
optLoe.freqBand     = [0 inf]; % Focus on a specific banwidth
optLoe.filter       = 5;        % Savinsky-Sogolev digital filter window size
[sysLoe,info]       = mor.lti({data.W,data.H},red,optLoe);
```

Listing 2.1: Call of the **`mor.lti`** function, for data-driven model approximation using Loewner framework with stability enforcement.

With reference to the above code, one seeks for a model of dimension $r = 30$, with the optional arguments ensuring first the stability of the reduced order model, second, by applying input-output normalisation (to equally match the transfer whatever the gain are), third, to match data over the entire spectrum and by finally applying a pre-filter (non causal zeros phase digital filter) to smooth the data.
Then, the following Figure 2.13 is obtained, showing the data collected (dotted blue) on the aircraft and its approximation (solid red), as well as the modal content. This plot clearly shows a very good matching. Thanks to the **MOR** Toolbox software, engineers are now able to evaluate their model with respect to the real aircraft data and to double check the modal information, crucial in aero-elastic engineering.

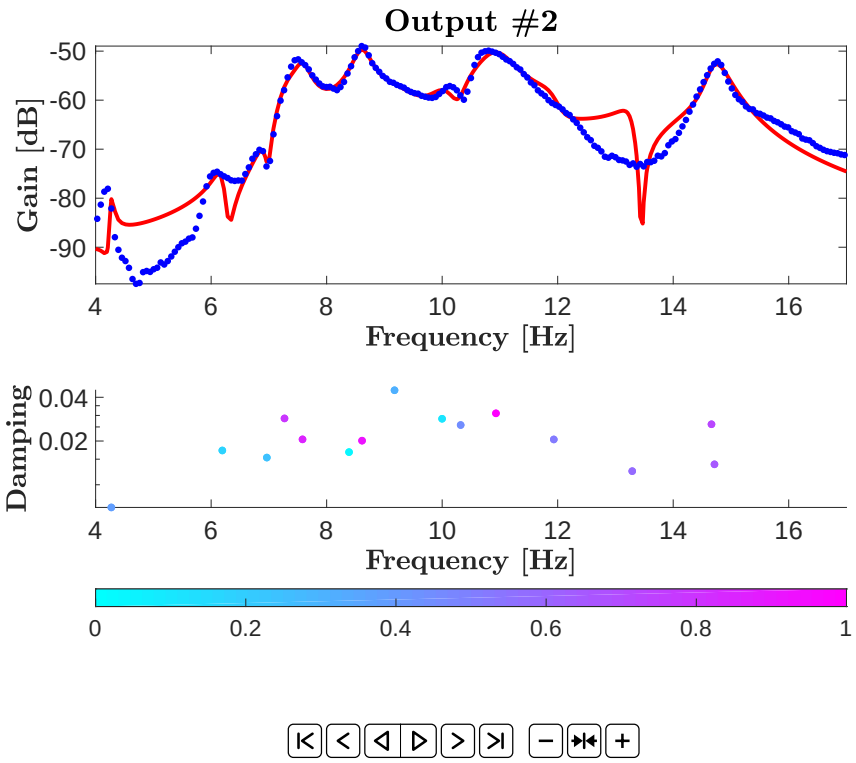


Figure 2.13: Approximated model of order $n = 30$, without anti-vibration control law. Top frame: comparison of the frequency response obtained during the **GVT** (blue dots) and the approximated model (solid red line). Bottom frame: damping map with varying colour as a function of the residue magnitude.

[a] The anti-vibration controller is designed thanks to a model-based techniques, involving an aero-servoelastic aircraft model of the **Dassault-Aviation** Falcon 7X (which have been approximated), followed by an $\mathcal{H}_\infty$-norm minimisation controller design done with `hinfstruct` by Apkarian and Noll (2006). See the connected papers from Poussot-Vassal et al. (2013); Meyer et al. (2016).

2.5 Conclusions

In this chapter we tried to present some of the most relevant results for linear large-scale dynamical model approximation. By relevant, one intends from the author's point of view and in resonance with the research activities presented along the manuscript. Obviously, it is clear that this chapter is absolutely not representative of all the numerous works and results available neither in the approximation community nor in the numerical and linear algebra approximation fields. Still, relevant frameworks have been given and suggested to the reader. In addition, the interest of model approximation has been illustrated through exotic academic and industrial applications. The main tool emphasised in this chapter, shared by models and data-based approximation, is the interpolatory one. This framework is actually the basis of the results presented in the coming Chapters 3, 4 and 5, constituting the main contributions of my research activities.

In the following chapters, we will now focus on the main contributions I have been developing in collaboration with colleagues along my research activities. All these results are rooted on the model approximation tools and framework presented in this chapter

Chapter 3

Model approximation over frequency-limited support

Choisir c'est renoncer.

André Gide

Contents

So far and in Chapter 2, unbounded frequency support model approximation was mainly addressed. In this chapter, two methods and procedures to approximate a dynamical model over a limited frequency range, are presented. After a quick introduction, motivation and problem formulation in Section 3.1, Section 3.2 introduces **FL-ISTIA** as a first approach to perform model approximation over a limited frequency band (*i.e.* as an $\mathcal{H}_{2,\Omega}$-norm oriented method). Then, Section 3.3 actually considers the frequency-limited problem directly and both describes the $\mathcal{L}_{2,\Omega}$ and $\mathcal{H}_{2,\Omega}$-norm model approximation optimality conditions. Then, **DARPO** is introduced as a practical and numerical way to reach these conditions. Section 3.4 links this last result with the interpolatory framework, connecting and extending this new solution to the standard interpolatory setup described in Chapter 2, in an un-standard fashion. Conclusions are reported in Section 3.5. This chapter is largely based on the Ph.D. thesis of Vuillemin (2014) and on the contributive papers of Vuillemin et al. (2013a, 2019) (note also that the original idea was already published in **`arXiv:1211.1858`** on November 8th, 2012 by the same authors).

3.1 Motivating example and problem formulation

3.1.1 Forewords

Depending on their complexity and of the desired representativeness, physical dynamical systems are represented by complex mathematical models. Generally these latter are defined over the entire frequency range since the effects and physical informations can hardly be a priori decoupled or isolated without affecting the rest of the phenomena and dynamics of interest. As presented in Chapter 2 and more deeply in Antoulas (2005), various approaches exist for the approximation of large-scale **LTI** models. Classically the model approximation problem is recast as an optimisation one in which the objective is to minimise

the $\mathcal{H}_2$-norm. Yet in many cases, focusing on a bounded frequency interval only may be more relevant. Indeed, the large-scale model might be less representative of the underlying physical system in some frequency domains and discarding them enables to increase the approximation accuracy where the initial model is actually representative. Also, control laws are generally designed to act over a limited frequency interval, due to actuators bandwidth or to prevent them from disturbing (high frequency) badly modelled dynamics. Optimal bounded frequency interval approximation elegantly enables translating these practical considerations and is addressed in this chapter.

3.1.2 Motivating example

A practical case encountered in collaboration with **Dassault-Aviation** engineers, is first given to illustrate the importance and potential gain earned when using frequency-limited model approximation.

Use-case 4 - Dassault-Aviation vibration control

This use-case illustrates the benefit in frequency-limited model approximation on a **Dassault-Aviation** generic business jet, in view of active vibration control (see Poussot-Vassal et al., 2013, Clean Sky 1 - SFWA project). We consider dynamical models describing the aircraft behaviour, linearised around $n_s = 48$ different flight and mass configurations. The control by feedback of the movable surfaces (*e.g.* ailerons, elevators) is generally used to enhance the aircraft performances and efficiency. Among the different phenomena to control, one can mention, the flight mechanics, the loads, the vibrations and the flutter[a]. For practical and safety reasons, all these phenomena appearing at varying frequency ranges, are treated separately (and often by different teams), based on the same original - large-scale - model. Consequently, approximating such a model over some frequency ranges may be interesting to focus on the phenomena of interest while approximating with a much lower order model.

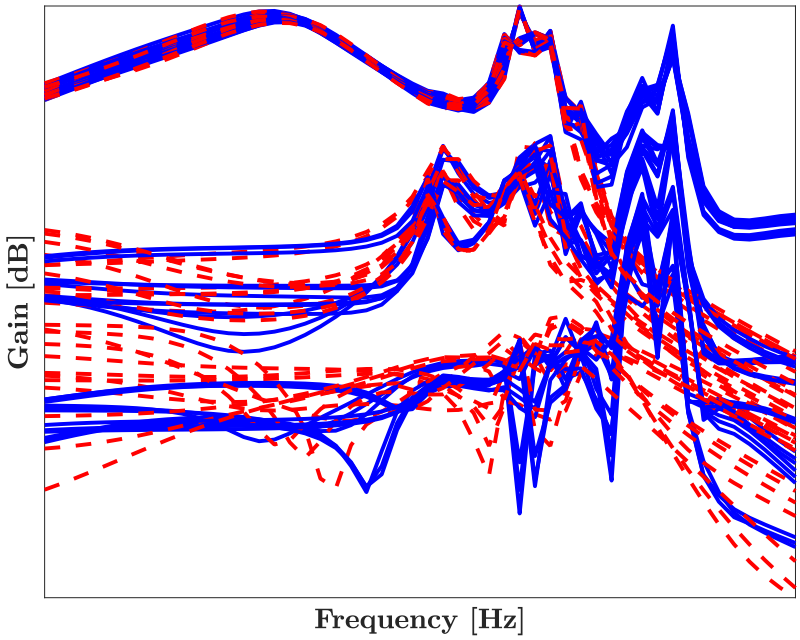


Figure 3.1: Sigma plot of different models obtained at varying flight configurations using **FLISTIA** where $\Omega = [0, 15]$Hz. Original model (solid blue) and reduced one (dashed red).

During the collaboration with **Dassault-Aviation** within the Clean Sky 1 - SFWA European project, an objective was to perform vibration attenuation over the frequency range $[7, 10]$Hz, without affecting low frequencies $[0, 3]$Hz, tuned by experts for flight performances. To this aim, both $\mathcal{H}_2$ and $\mathcal{H}_{2,\Omega}$ approximation of the n_s large-scale models representing a business jet aircraft at varying configurations, has been done. On Figure 3.1, the singular values plot of the original (with dimension $n_i \approx 700$,$i = 1, \ldots, n_s$) and reduced (with $r = 16$) models obtained with frequency-limited model approximation, are shown.

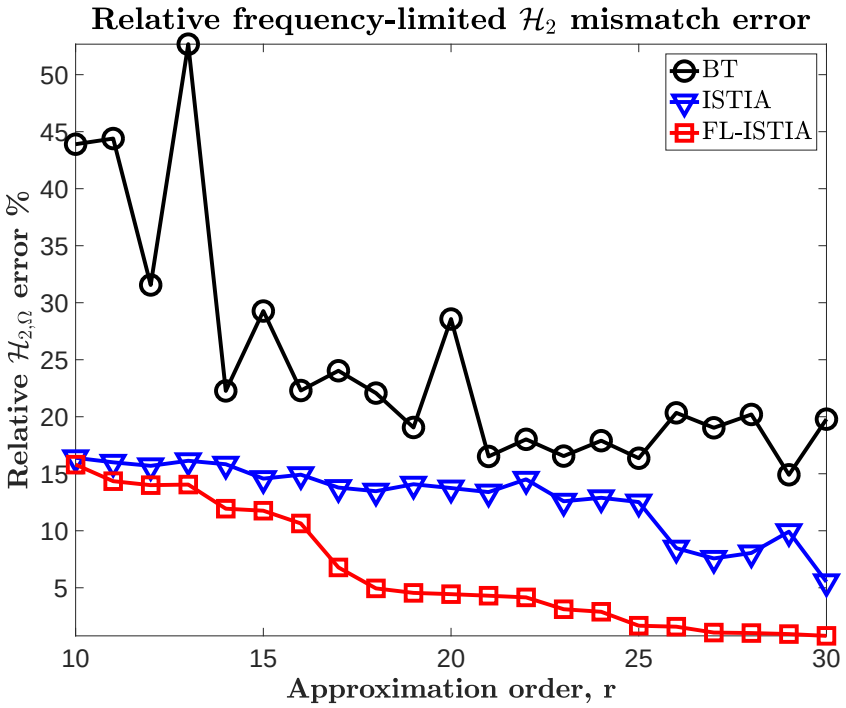


Figure 3.2: Mean relative mismatch $\mathcal{H}_{2,\Omega}$-norm applied on a **Dassault-Aviation** use-cases when the n_s reduced model are obtained with the **BT** , **ISTIA** and **FL-ISTIA** where $\Omega = [0, 15]$Hz.

Reader may note that an approximation over the entire frequency range would have required an order $r = 40$ to achieve the same accuracy level over $\Omega = [0, 15]$Hz. As a more measurable illustration, assertion of the relevance of the approach, one can observe on Figure 3.2, the good relative mismatch $\mathcal{H}_{2,\Omega}$-norm decay when using **FL-ISTIA** instead of the standard **ISTIA** rocedure.
One direct benefit of the frequency-limited model approximation is that the n_s multiple models involved in the anti-vibration control design can now be even more reduced. As a direct consequence, when the robust control law is designed and robustness is analysed, this complexity simplification directly leads to an easier problem to manipulate, especially in an iterative industrial context.
In Poussot-Vassal et al. (2015), this frequency limited approximation has been used to approximate a set of multiple models and transform them into a parametric form. Then, in Meyer et al. (2016) and Meyer et al. (2017), these approximations have been used for controller design and validation in **GVT** then in real flight on a Falcon 7X. The obtained measurements lead to a vibration attenuation around 40% in the frequency domain of interest.

[a] Which represent behaviours identified and appearing at increasing frequencies. Aerodynamical engineers usually consider them separately.

This use-case provides a fancy justification for frequency-limited model approximation, by directly linking the approximation to a control and physical engineers needs (details on the control tuning are given in Poussot-Vassal et al. (2015)). Now, let us describe the problem mathematical formulation.

3.1.3 Frequency-limited $\mathcal{H}_2$ model approximation

Let be given $\mathbf{H} \in \mathcal{H}_\infty$ equipped with the n-th order realisation $\mathcal{S} : (E, A, B, C, D)$. The frequency-limited model approximation problem considered here consists in finding a **LTI** model $\hat{\mathbf{H}} \in \mathcal{H}_\infty$ of smaller order r that solves the following problem.

Problem 3.1: $\mathcal{H}_{2,\Omega}$ model approximation

Given $\mathbf{H} \in \mathcal{H}_\infty$, the $\mathcal{H}_{2,\Omega}$ model approximation problem consists of seeking an approximation $\hat{\mathbf{H}} \in \mathcal{H}_\infty$ of $\mathbf{H}$, such that

$$\hat{\mathbf{H}} := \arg \min_{\substack{\mathbf{G} \in \mathcal{H}_\infty \\ \mathbf{dim}(\mathbf{G}) = r \ll n}} ||\mathbf{H} - \mathbf{G}||_{\mathcal{H}_{2,\Omega}}. \tag{3.1}$$

As the frequency-limited norm is also defined for unstable models, similarly to the $\mathcal{H}_{2,\Omega}$, the $\mathcal{L}_{2,\Omega}$ one can be defined as follows.

Problem 3.2: $\mathcal{L}_{2,\Omega}$ model approximation

Given $\mathbf{H} \in \mathcal{L}_\infty$, the $\mathcal{L}_{2,\Omega}$ model approximation problem consists of seeking an approximation $\hat{\mathbf{H}} \in \mathcal{L}_\infty$ of $\mathbf{H}$, such that

$$\hat{\mathbf{H}} := \arg \min_{\substack{\mathbf{G} \in \mathcal{L}_\infty \\ \mathbf{dim}(\mathbf{G}) = r \ll n}} ||\mathbf{H} - \mathbf{G}||_{\mathcal{L}_{2,\Omega}}.$$

In Problem 3.2, $||\cdot||_{\mathcal{L}_{2,\Omega}}$ stands as the $\mathcal{L}_2$-norm restriction over the frequency-interval $\Omega = [0, \omega]$, *i.e.*

$$||\mathbf{H}||_{\mathcal{L}_{2,\Omega}} \triangleq \left(\frac{1}{2\pi} \int_{-\omega}^{\omega} \|\mathbf{H}(\imath\nu)\|_F^2 d\nu \right)^{\frac{1}{2}}. \tag{3.2}$$

In the sequel, we denote by $\mathcal{L}_{2,\Omega}$ the set of analytic functions for which the integral in (3.2) is finite. Clearly, elements of $\mathcal{H}_2$ and $\mathcal{H}_\infty$ are also in $\mathcal{L}_{2,\Omega}$. In what follows, the frequency-limited approximation Problem 3.1 is addressed in two different ways: *(i)* first, indirectly, using the frequency-limited **ISTIA** approach of Vuillemin et al. (2013b), involving $\mathcal{Q}_\Omega$ (or $\mathcal{P}_\Omega$), a frequency limited gramian instead of the infinite one $\mathcal{Q}$ (or $\mathcal{P}$), then *(ii)* through **DARPO** , a dedicated descent method for the $\mathcal{L}_2$-norm minimisation (3.2), applied to the mismatch error (3.1), initially presented in Vuillemin et al. (2014a).

It should be noted that Problem 3.1 has also been considered in Petersson and Lofberg (2014). The difference lies in the framework used to express the approximation error and the first-order optimality conditions. In Petersson and Lofberg (2014), an approach based on the state-space representation is used while here, both in Sections 3.2 and 3.3, the partial fraction decomposition of the transfer function is exploited. We will see how this formulation may provide some advantages.

Remark 3.1 (Weighted $\mathcal{H}_2$ model approximation) *Instead of frequency-limited model approximation, as it is done the robust control community with the generalised plant and $\mathcal{H}_2/\mathcal{H}_\infty$-norm control-oriented synthesis, weighted model approximation has also been considered, instead of the frequency-limited one. The problem then turns to find $\hat{\mathbf{H}}$ such that,*

$$\hat{\mathbf{H}} := \arg \min_{\substack{\mathbf{G} \in \mathcal{H}_2 \\ \mathbf{dim}(\mathbf{G}) = r \ll n}} ||\mathbf{W}_o \mathbf{H} \mathbf{W}_i - \mathbf{G}||_{\mathcal{H}_2}.$$

In practice this approach is quite efficient, but suffers from different elements. First, one has to design the weighting filters $\mathbf{W}_o$ and $\mathbf{W}_i$, which might be tedious for practitioners, second, once the reduced order model found, inversion of the filters is mathematically necessary to come back to the actual solution.

As introduced above, two dedicated methods and algorithm have been developed for approximating a dynamical model over a finite frequency range. The first one, presented in Vuillemin et al. (2013b), is a modification of the **ISTIA** procedure orientating the objective toward Problem 3.1, while the second one, the **DARPO** procedure, which directly treats Problems 3.1 and 3.2, is described in Vuillemin et al. (2019).

3.2 $\mathcal{H}_{2,\Omega}$ oriented model approximation

3.2.1 General idea

As proposed by Gawronski (2004), the frequency-limited gramians, denoted $\mathcal{P}_\Omega$ and $\mathcal{Q}_\Omega$ in equations (1.11), can directly be used for model reduction through the **FL-BT** method. It consists in using frequency-limited gramians instead of classical ones to perform a balanced truncation (see Antoulas (2005) for more details). As shown in Gugercin and Antoulas (2004), using this approach is equivalent to apply a frequency-weighted balanced truncation with perfect filters. However, unlike the classical (infinite) balanced truncation, it does not guarantee the stability preservation any longer. Moreover, computing the frequency-limited gramians require solving two large-scale Lyapunov equations and to evaluate the logarithm of a large-scale matrix, making this approach numerically more complex to achieve than its standard unbounded version. The **FL-ISTIA** procedure trades a part of the numerical burden with Krylov subspaces.

3.2.2 The FL-ISTIA procedure

The frequency-limited gramians are implicitly playing the role of filters in the frequency-limited balanced truncation. In a similar way, using one frequency-limited gramian in the **ISTIA** procedure instead of one infinite ones, make the algorithm more efficient in terms of $\mathcal{H}_{2,\Omega}$-norm. This effect can be accentuated by choosing the initial interpolation points $\{\sigma_l\}_{l=1}^r$ so that their modulus remain in the concerned frequency interval. The Frequency-Limited **ISTIA** or **FL-ISTIA** is presented in Algorithm 3.

Algorithm 3 FL-ISTIA - Frequency-Limited Iterative SVD-Tangential Interpolation Algorithm

Require: $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$, $C \in \mathbb{R}^{n_y\times n}$, $\left\{\sigma_l^{(0)}, \mathbf{r}_l^{(0)}\right\}_{l=1}^r \in \{\mathbb{C}\times\mathbb{C}^{n_u}\}$

1: Construct $V \in \mathbb{R}^{n\times r}$,

$$\left[(\sigma_1^{(0)}E - A)^{-1}B\mathbf{r}_1^{(0)}, \ldots, (\sigma_r^{(0)}E - A)^{-1}B\mathbf{r}_r^{(0)}\right] \quad = \quad \mathbf{span}\,(V)$$

2: Compute $\mathcal{O}_\Omega$, the frequency-limited observability Gramian solving (1.11)
3: Compute $W = \mathcal{O}_\Omega V(V^T\mathcal{O}_\Omega V)^{-1} \in \mathbb{R}^{n\times r}$
4: **while** not converged **do**
5: Set $i \leftarrow i+1$
6: Project the model: $\hat{E} = W^TEV$, $\hat{A} = W^TAV$, $\hat{B} = W^TB$, $\hat{C} = CV$
7: Solve the eigenvalue problem $(\hat{E}, \hat{A})$ and obtain eigenvalues $\{\hat{\lambda}_l\}_{l=1}^r$ and right R eigenvectors
8: Set $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l\}_{l=1}^r = \{\hat{\lambda}_l, (R\hat{B})^H\mathbf{e}_l\}_{l=1}^r$
9: Set $\{\sigma_l^{(i)}, \mathbf{r}_l^{(i)}\}_{l=1}^r = \{-\hat{\lambda}_l, \hat{\mathbf{b}}_l\}_{l=1}^r$
10: Construct $V \in \mathbb{R}^{n\times r}$ s.t.

$$\left[(\sigma_1^{(i)}E - A)^{-1}B\mathbf{r}_1^{(i)}, \ldots, (\sigma_r^{(i)}E - A)^{-1}B\mathbf{r}_r^{(i)}\right] \quad = \quad \mathbf{span}\,(V)$$

11: Compute $W = \mathcal{O}_\Omega V(V^T\mathcal{O}_\Omega V)^{-1} \in \mathbb{R}^{n\times r}$
12: **end while**
13: Apply projectors V and W

Ensure: $V, W \in \mathbb{R}^{n\times r}$ and part of the interpolatory conditions (2.12) are satisfied

With reference to Algorithm 3, the following comments can be formulated:

- The frequency-limited controllability gramian $\mathcal{P}_\Omega$ can be indifferently used instead of the observability one $\mathcal{Q}_\Omega$ (1.11), in Algorithm 3. In that case, the left projector W is built as a basis of a Krylov subspace, *i.e.*

$$\left[(\sigma_1^{(i)}E - A)^{-H}C\mathbf{l}_1^{(i)}, \ldots, (\sigma_r^{(i)}E - A)^{-H}C\mathbf{l}_r^{(i)}\right] \quad = \quad \mathbf{span}\,(W)$$

  where $\{\mathbf{l}_l\}_{l=1}^r = \{\hat{\mathbf{c}}_l\}_{l=1}^r = \{\hat{C}L\mathbf{e}_l\}_{l=1}^r$. The controllability gramian is then involved in the construction of the right projector $V = \mathcal{P}_\Omega \left(W^T\mathcal{P}_\Omega W\right)^{-1}$.

- The main numerical cost of this algorithm comes from the computation of the frequency-limited gramian and the resolution of r linear systems at each iterations. As computing a frequency-limited gramian requires, in addition of solving a Lyapunov equation, to evaluate the matrix logarithm, the **FL-ISTIA** is much more complex than the **ISTIA** and for the same reasons as those presented in Gugercin (2008), it is cheaper than the **FL-BT** (depending, of course, on the number of iterations).

- The algorithm stops when the interpolation points σ_l do not evolve anymore. Other stopping criteria could be considered. One could for instance stops when the $\mathcal{H}_{2,\Omega}$-norm of the error does not evolve anymore (see Vuillemin, 2014, for details and illustrations).

- To alleviate the computational complexity of the algorithm in the very large-scale settings, a low-rank approximation of the gramian may be used. For instance, one can refer to (Antoulas, 2005, Chap. 12) and references therein. This kind of Lyapunov approximation approach is also widely explored in the works of Simoncini (2007); Shank et al. (2016).

- Note that as in other Krylov subspaces-based methods, to obtain real valued projectors V and/or W, the interpolation points must either be real or closed under complex conjugation. Indeed, in the latter case, if two vectors are complex conjugate $\mathbf{v}_2 = \mathbf{v}_1^*$, then they span the same subspace as their real and imaginary part, *i.e.* $\mathbf{span}\,(\mathbf{v}_1, \mathbf{v}_2) = \mathbf{span}\,(\Re(\mathbf{v}_1), \Im(\mathbf{v}_1))$. This also implies that only one single linear system has to be solved for each pair of complex conjugate interpolation points, which also reduces the computational complexity.

- Since it is based on a projection framework, direct feedthrough are not taken into account in this algorithm. A simple way to handle non strictly proper models is to set $\hat{D} = D$. However, as exposed later in Section 3.3, this is not the optimal choice considering the gradient of the approximation error with respect to $\hat{D}$.

3.2.3 Model stability issues

The properties of the **FL-ISTIA** concerning stability preservation and interpolation of the large-scale model are discussed below. To prove the stability of the reduced-order models built by **ISTIA** (see Gugercin and Antoulas, 2006), the Lyapunov equation of the initial large-scale model is projected thus leading to the reduced-order model Lyapunov equation. And, due to inertia results, the reduced-order model is proved to be stable. Here however, following the same reasoning as in Gugercin and Antoulas (2006) is not applicable any longer. Indeed, let us assume that the basis is such that $\mathcal{Q}_\Omega = I_n$, then, the large-scale Lyapunov equation is

$$A^T + A + W_o\,(\Omega) = 0.$$

By applying the projectors $W^T = V^T$ and V to the equation, it comes that

$$\hat{A}^T + \hat{A} + V^T W_o\,(\Omega)\,V = 0.$$

The last term of the reduced-order Lyapunov equation is not necessarily positive semi-definite, hence stability cannot be proved. In order to do so, the same modification that has been developed to ensure the stability with the **FL-BT** could be applied (see Gugercin and Antoulas, 2004). However, in the case of **FL-BT** one observes that this modification also leads to a loss of performances in terms of approximation

error in the considered frequency interval. Since instability has rarely been observed in the reduced-order model built by **FL-ISTIA** , this modification is not applied to the algorithm above presented. Moreover, in that case, the rational $\mathcal{RH}_\infty$ projection presented in Chapter 1 may be applied instead. To the author's experience and feeling, this is usually the best way to take.

3.2.4 About the interpolatory conditions

The Krylov subspace used in **FL-ISTIA** is similar to the one used in **IRKA** or **ISTIA** procedures. It implies that at convergence, the reduced-order model tangentially interpolates at $\{-\hat{\lambda}_l\}_{l=1}^r$ the initial large-scale one $\mathbf{H}$, from either the left or the right (depending on the Krylov subspace that has been chosen). For instance, if at convergence, $\left[(\sigma_1^{(i)}E - A)^{-1}B\mathbf{r}_1^{(i)}, \ldots, (\sigma_r^{(i)}E - A)^{-1}B\mathbf{r}_r^{(i)}\right] = \mathbf{span}\,(V)$, then for all $l = 1, \ldots, r$,

$$\mathbf{H}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l = \hat{\mathbf{H}}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l. \tag{3.3}$$

Note that equation (3.3) is one of the first-order optimality conditions for the optimal $\mathcal{H}_2$ case, as given in Theorem 2.6. For **SISO** models, just like **ISTIA** this implies that the reduced-order model built by **FL-ISTIA** is the best (in the $\mathcal{H}_2$-sense) among all the models which share the same eigenvalues. The same property holds in **SIMO** and **MISO** cases as well (depending on which side the Krylov subspace is built). In the **MIMO** case however, the optimality property is weaker since only one part of the first-order optimality conditions is fulfilled. In all cases, fulfilling this optimality condition implies that at convergence, the $\mathcal{H}_2$-norm of the approximation error between the large-scale and reduced-order models is equal to the difference of their $\mathcal{H}_2$-norm, *i.e.*

$$||\mathbf{H} - \hat{\mathbf{H}}||_{\mathcal{H}_2}^2 = ||\mathbf{H}||_{\mathcal{H}_2}^2 - ||\hat{\mathbf{H}}||_{\mathcal{H}_2}^2.$$

Still, fulfilling the $\mathcal{H}_2$ optimality conditions is not really relevant in the case of model approximation over a bounded frequency range and the choice of the Krylov subspace could be improved. This stands true also for the shifts point selection strategy which is a relaxation of a Newton's scheme based on the optimality conditions of the $\mathcal{H}_2$ problem. However, as it will appear later in this chapter, even if tangential interpolatory optimality conditions of the $\mathcal{H}_{2,\Omega}$ problem can be written, their expression as practical and Krylov-like subspace seems far to be trivial, and no solution is known, yet. Therefore, **FL-ISTIA** remains a relevant approach for frequency-limited model reduction.

Remark 3.2 (Link with TSIA procedure) *Similarly to the **FL-ISTIA** procedure, one can follow the approach of Xu and Zeng (2010) and design the **FL-TSIA** approach involving Sylvester equations instead of Krylov subspaces (see Vuillemin et al. (2013b) for details).*

3.3 $\mathcal{H}_{2,\Omega}$ model approximation

Approximation of **LTI** models over a finite frequency interval is now addressed in a more direct way, through the restriction of the $\mathcal{H}_2$-norm to a bounded interval, called $\mathcal{L}_{2,\Omega}$-norm (and with a slight language abuse, $\mathcal{H}_{2,\Omega}$-norm). A framework for optimal $\mathcal{L}_{2,\Omega}$ model approximation relying on the partial fraction decomposition of the transfer function $\hat{\mathbf{H}}$, is presented. It enables deriving a spectral expression for the $\mathcal{L}_{2,\Omega}$-norm of **LTI** models that can be used to express the norm of the approximation error which can in turn be differentiated to formulate the first-order optimality conditions with respect to the variables. These optimality conditions are then analysed to simplify the optimisation problem and are finally rewritten as interpolation conditions, similar to those existing in the $\mathcal{H}_2$ case. In addition, a procedure treating the problem using a dedicated descent scheme, called **DARPO** (see *e.g.* Vuillemin et al., 2014a, 2019), is given.

3.3.1 Spectral formulation of the approximation error

The spectral formulation of the $\mathcal{L}_{2,\Omega}$-norm presented in Chapter 1 and in Theorem 1.5 is now used to express the $\mathcal{L}_{2,\Omega}$-norm of the approximation error $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}}) = ||\mathbf{H} - \hat{\mathbf{H}}||_{\mathcal{L}_{2,\Omega}}^2$. Indeed, let us consider two

LTI models $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{L}_\infty$ of order n and r, respectively, that can be written as,

$$\begin{aligned} \mathbf{H}(s) &= \sum_{j=1}^{n} \frac{\mathbf{c}_j \mathbf{b}_j^T}{s-\lambda_j} + D \\ &= \sum_{j=1}^{n} \frac{\Phi_j}{s-\lambda_j} + D \end{aligned} \quad \text{and} \quad \begin{aligned} \hat{\mathbf{H}}(s) &= \sum_{k=1}^{r} \frac{\hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T}{s-\hat{\lambda}_k} + \hat{D} \\ &= \sum_{k=1}^{r} \frac{\hat{\Phi}_k}{s-\hat{\lambda}_k} + \hat{D}, \end{aligned} \tag{3.4}$$

then $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}}) = ||\mathbf{H} - \hat{\mathbf{H}}||^2_{\mathcal{L}_{2,\Omega}}$ can be expressed with respect to the poles and residues of both models as shown in Theorem 3.1. Note that this expression is fully expanded to ease the differentiation process in the next section. Yet, a more condensed expression can also be written as presented in Corollary 3.1.

Theorem 3.1: $\mathcal{L}_{2,\Omega}$ mismatch error

Given $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{L}_\infty$ of order n and r, respectively, that can be written as in (3.4). By denoting

$$a_\omega(s) \triangleq \frac{2}{\pi} \mathbf{atan}\left(\frac{\omega}{s}\right),$$

the $\mathcal{L}_{2,\Omega}$-norm of the approximation error between $\mathbf{H}$ and $\hat{\mathbf{H}}$ can be expressed as

$$\begin{aligned} \mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}}) &= \sum_{j,k=1}^{n} \frac{\mathbf{tr}\left(\Phi_j \Phi_k^T\right)}{\lambda_j + \lambda_k} a_\omega(\lambda_j) + \sum_{j,k=1}^{r} \frac{\mathbf{tr}\left(\hat{\Phi}_j \hat{\Phi}_k^T\right)}{\hat{\lambda}_j + \hat{\lambda}_k} a_\omega(\hat{\lambda}_j) \\ &\quad - \sum_{j=1}^{n} \sum_{k=1}^{r} \frac{\mathbf{tr}\left(\Phi_j \hat{\Phi}_k^T\right)}{\lambda_j + \hat{\lambda}_k} \left(a_\omega(\lambda_j) + a_\omega(\hat{\lambda}_k)\right) \\ &\quad + \sum_{k=1}^{r} \mathbf{tr}\left(\hat{\Phi}_k \tilde{D}^T\right) a_\omega(\hat{\lambda}_k) - \sum_{j=1}^{n} \mathbf{tr}\left(\Phi_j \tilde{D}^T\right) a_\omega(\lambda_j) + \frac{\omega}{\pi} \mathbf{tr}\left(\tilde{D}\tilde{D}^T\right) \end{aligned} \tag{3.5}$$

where $\Phi_j = \mathbf{c}_j \mathbf{b}_j^T$ $(j = 1, \ldots, n)$, $\hat{\Phi}_k = \hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T$ $(k = 1, \ldots, r)$ and $\tilde{D} = D - \hat{D}$.

Proof 3.1 (of Theorem 3.1) *is available in the Ph.D. manuscript of Vuillemin (2014), Chapter 7.*

One may notice that when the optimal $\mathcal{H}_2$ approximation problem is considered, the $\mathcal{H}_2$ approximation error is given as (see also Chapter 2 and relation (2.11))

$$\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}) = ||\mathbf{H}||^2_{\mathcal{H}_2} + ||\hat{\mathbf{H}}||^2_{\mathcal{H}_2} - 2\sum_{k=1}^{r} \mathbf{tr}\left(\mathbf{H}(-\hat{\lambda}_k)\hat{\Phi}_k^T\right).$$

Here, a similar expression can be obtained for the $\mathcal{L}_{2,\Omega}$ case, by reformulating (3.5) as presented in Corollary 3.1.

Corollary 3.1: $\mathcal{L}_{2,\Omega}$ mismatch error

Given $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{H}_\infty$ of order n and r, respectively, that can be written as in (3.4). Relation (3.5) can be factored as

$$\begin{aligned} \mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}}) &= ||\mathbf{H}||^2_{\mathcal{L}_{2,\Omega}} + ||\hat{\mathbf{H}}||^2_{\mathcal{L}_{2,\Omega}} \\ &\quad + \sum_{k=1}^{r} \mathbf{tr}\left(\hat{\Phi}_k \mathbf{H}(-\hat{\lambda}_k)^T\right) a_\omega(\hat{\lambda}_k) + \sum_{j=1}^{n} \mathbf{tr}\left(\Phi_j \hat{\mathbf{H}}(-\lambda_j)^T\right) a_\omega(\lambda_j) \\ &\quad - 2\frac{\omega}{\pi} \mathbf{tr}\left(D\hat{D}^T\right). \end{aligned} \tag{3.6}$$

In an optimisation context, the objective is to characterise the stationary points of Problem 3.2. For that purpose, the spectral formulation of the approximation error $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}})$ presented in Theorem 3.1 must be differentiated with respect to $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r$, the poles and residues of the reduced-order model. Later, in Section 3.4, these first-order optimality conditions are reformulated as interpolatory ones.

3.3.2 $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}})$ first-order optimality conditions

The approximation error $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}})$ is a real scalar function of the complex variables $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l\}_{l=1}^r$ and of the real variable $\hat{D}$. The corresponding derivatives are obtained by differentiating the expression (3.6) and are presented in Result 3.1.

Note that as $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}})$ involves both complex variables and their conjugates, it does not satisfy the Cauchy-Riemann conditions and is therefore not analytic. Hence, differentiation must be performed in a specific framework, known as the Wirtinger calculus (see *e.g.* Brandwood (1983); Remmert (1991)). From a practical point of view, the latter consists in treating two conjugate complex variables, *e.g.* $\hat{\lambda}_l$ and $\hat{\lambda}_l^*$, as independent.

Result 3.1: Gradient computation of $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}})$

Let $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{L}_\infty$ be systems with transfer functions as in (3.4). The gradients of the $\mathcal{L}_{2,\Omega}$ gap $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}}) = \mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\lambda}_k, \hat{\mathbf{b}}_k, \hat{\mathbf{c}}_k)$, shortly denoted $\mathcal{J}_{\mathcal{L}_{2,\Omega}}$ with respect to $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l, \hat{D}\}_{l=1}^r$ are given by the equations (3.7) below,

$$
\left\{
\begin{aligned}
\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \hat{\lambda}_l} &= \sum_{j=1}^{n} \hat{\mathbf{c}}_l^T \mathbf{c}_j \mathbf{b}_j^T \hat{\mathbf{b}}_l \left(\frac{a_\omega(\lambda_j) + a_\omega(\hat{\lambda}_l)}{\left(\lambda_j + \hat{\lambda}_l\right)^2} + \frac{2\omega}{\pi\left(\hat{\lambda}_l^2 + \omega^2\right)\left(\lambda_j + \hat{\lambda}_l\right)} \right) \\
&\quad - \sum_{k=1}^{r} \hat{\mathbf{c}}_l^T \hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T \hat{\mathbf{b}}_l \left(\frac{a_\omega(\hat{\lambda}_k) + a_\omega(\hat{\lambda}_l)}{\left(\hat{\lambda}_k + \hat{\lambda}_l\right)^2} + \frac{2\omega}{\pi\left(\hat{\lambda}_l^2 + \omega^2\right)\left(\hat{\lambda}_k + \hat{\lambda}_l\right)} \right) \\
&\quad - \frac{2\omega}{\pi\left(\hat{\lambda}_l^2 + \omega^2\right)} \hat{\mathbf{c}}_l^T \left(D - \hat{D}\right) \hat{\mathbf{b}}_l \\
\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \hat{\mathbf{b}}_l} &= -\sum_{j=1}^{n} \frac{\mathbf{b}_j \mathbf{c}_j^T \hat{\mathbf{c}}_l}{\hat{\lambda}_l + \lambda_j} \left(a_\omega(\lambda_j) + a_\omega(\hat{\lambda}_l) \right) \\
&\quad + \sum_{k=1}^{r} \frac{\hat{\mathbf{b}}_k \hat{\mathbf{c}}_k^T \hat{\mathbf{c}}_l}{\hat{\lambda}_l + \hat{\lambda}_k} \left(a_\omega(\hat{\lambda}_k) + a_\omega(\hat{\lambda}_l) \right) + (D - \hat{D})^T \hat{\mathbf{c}}_l a_\omega(\hat{\lambda}_l) \\
\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \hat{\mathbf{c}}_l} &= -\sum_{j=1}^{n} \frac{\mathbf{c}_j \mathbf{b}_j^T \hat{\mathbf{b}}_l}{\hat{\lambda}_l + \lambda_j} \left(a_\omega(\lambda_j) + a_\omega(\hat{\lambda}_l) \right) \\
&\quad + \sum_{k=1}^{r} \frac{\hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T \hat{\mathbf{b}}_l}{\hat{\lambda}_l + \hat{\lambda}_k} \left(a_\omega(\hat{\lambda}_k) + a_\omega(\hat{\lambda}_l) \right) + (D - \hat{D}) \hat{\mathbf{b}}_l a_\omega(\hat{\lambda}_l) \\
\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \hat{D}} &= \sum_{j=1}^{n} \mathbf{c}_j \mathbf{b}_j^T a_\omega(\lambda_j) - \sum_{k=1}^{r} \hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T a_\omega(\hat{\lambda}_k) - \frac{2}{\pi}\omega \left(D - \hat{D}\right)
\end{aligned}
\right. \tag{3.7}
$$

This result is the main basis for the derivation of a descent algorithm and may now be exploited in the **DARPO** procedure.

3.3.3 The DARPO procedure

Now one has the expression of $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\hat{\mathbf{H}})$ and its derivative with respect to the $\{\hat{\lambda}_l, \hat{\mathbf{b}}_l, \hat{\mathbf{c}}_l, \hat{D}\}_{l=1}^r$ parameters as given in Result 3.1, it is possible to derive a descent algorithm with objective of minimizing the mismatch error. This decent scheme has been proposed by Vuillemin et al. (2014a) and is known as the **DARPO** algorithm. The procedure is recalled in Algorithm 4.

Algorithm 4 DARPO - Descent Algorithm for Residues and Poles Optimization

Require: $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$, $C \in \mathbb{R}^{n_y\times n}$, $\Omega = [0, \omega]$ with $\omega > 0$ and $r \in \mathbb{N}^*$

1: Compute the eigenvalues and associated eigenvectors of (E, A) to determine $\{\lambda_j, \mathbf{c}_j, \mathbf{b}_j\}_{j=1}^n$

2: Choose an initial (stable) point z_0 composed of $\{\hat{\lambda}_l^{(0)}, \hat{\mathbf{b}}_l^{(0)}, \hat{\mathbf{c}}_l^{(0)}, \hat{D}^{(0)}\}_{l=1}^r$

3: Set $k = 0$.

4: **while not converged do**

5: Set

$$\mathbf{z}_k = \left[\hat{\lambda}_1^{(k)} \dots \hat{\lambda}_r^{(k)} \quad \hat{\mathbf{b}}_1^{(k)} \dots \hat{\mathbf{b}}_r^{(k)} \quad \hat{\mathbf{c}}_1^{(k)} \dots \hat{\mathbf{c}}_r^{(k)} \quad \hat{D}^{(k)}\right]$$

6: Compute the error $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\mathbf{z}_k)$ and the associated gradient following equations (3.7)

$$\left.\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \mathbf{z}^*}\right|_{\mathbf{z}=\mathbf{z}_k}.$$

7: Choose the descent direction (BFGS in practice)

$$\mathbf{p}_k = -2\left.\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \mathbf{z}^*}\right|_{\mathbf{z}=\mathbf{z}_k}.$$

8: Choose the step length α_k such that $\mathcal{J}_{\mathcal{L}_{2,\Omega}}(\mathbf{z}_k + \alpha_k \mathbf{p}_k)$ satisfies the strong Wolfe conditions (and such that the poles do not cross the imaginary axis)

9: Set $\mathbf{z}_{k+1} = \mathbf{z}_k + \alpha_k \mathbf{p}_k$

10: Set $k \leftarrow k + 1$

11: **end while**

12: Use $\{\hat{\lambda}_l^{(k)}, \hat{\mathbf{b}}_l^{(k)}, \hat{\mathbf{c}}_l^{(k)}, \hat{D}^{(k)}\}_{l=1}^r$ to construct $\hat{E}$, $\hat{A}$, $\hat{B}$, $\hat{C}$ and $\hat{D}$.

DARPO is a descent method performing in the complex domain. It is based on specific parametrisation, the pole residue one. Obviously other parametrisation may be chosen such as the Schur one.

3.4 Connection with interpolatory conditions

As exposed in Chapter 2, in the specific case where $\hat{\mathbf{H}} \in \mathcal{H}_2$ is an optimal $\mathcal{H}_2$ approximant of $\mathbf{H} \in \mathcal{H}_2$, then the following tangential interpolation conditions are satisfied (see *e.g.* Gugercin et al. (2008); Van Dooren et al. (2008))

$$\begin{aligned} \mathbf{H}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l &= \hat{\mathbf{H}}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \\ \hat{\mathbf{c}}_l^T\mathbf{H}(-\hat{\lambda}_l) &= \hat{\mathbf{c}}_l^T\hat{\mathbf{H}}(-\hat{\lambda}_l) \\ \hat{\mathbf{c}}_l^T\mathbf{H}'(-\hat{\lambda}_l)\hat{\mathbf{b}}_l &= \hat{\mathbf{c}}_l^T\hat{\mathbf{H}}'(-\hat{\lambda}_l)\hat{\mathbf{b}}_l. \end{aligned} \tag{3.8}$$

In the frequency-limited support $\mathcal{L}_{2,\Omega}$ case, similar interpolatory conditions as (3.8) may actually be obtained, as presented in Theorem 3.2.

Theorem 3.2: $\mathcal{L}_{2,\Omega}$ first order tangential optimality conditions

Let us define the following $\mathbf{T}_\omega$ and $\hat{\mathbf{T}}_\omega$ irrational functions,

$$\mathbf{T}_\omega(s) = \sum_{j=1}^{n} \mathbf{c}_j \mathbf{b}_j^T \left(\frac{a_\omega(\lambda_j) - a_\omega(s)}{s - \lambda_j}, \right) - D a_\omega(s)$$

and

$$\hat{\mathbf{T}}_\omega(s) = \sum_{k=1}^{r} \hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T \left(\frac{a_\omega(\hat{\lambda}_k) - a_\omega(s)}{s - \hat{\lambda}_k} \right) - \hat{D} a_\omega(s)$$

Then the three first equations of (3.7) are equivalent to

$$\begin{array}{rcl} \mathbf{T}_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{T}}_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \\ \hat{\mathbf{c}}_l^T \mathbf{T}_\omega(-\hat{\lambda}_l) & = & \hat{\mathbf{c}}_l^T \hat{\mathbf{T}}_\omega(-\hat{\lambda}_l) \\ \hat{\mathbf{c}}_l^T \mathbf{T}'_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{c}}_l^T \hat{\mathbf{T}}'_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \end{array} \tag{3.9}$$

and last equation of (3.7) is equivalent to (for $j = 1, \ldots, n_y$ and $k = 1, \ldots, n_u$)

$$\int_{-\omega}^{\omega} \mathbf{H}_{j,k}(j\nu) d\nu = \int_{-\omega}^{\omega} \hat{\mathbf{H}}_{j,k}(j\nu) d\nu.$$

Unlike the $\mathcal{H}_2$ interpolation conditions, the frequency-limited ones (3.9) do not involve directly the transfer functions $\mathbf{H}$ and $\hat{\mathbf{H}}$ but irrational functions $\mathbf{T}_\omega(\mathbf{H})$ and $\hat{\mathbf{T}}_\omega(\hat{\mathbf{H}})$, parametrised by of $\mathbf{H}$ and $\hat{\mathbf{H}}$ defined as

$$\mathbf{T}_\omega(s) = \mathbf{H}_\omega(s) + \mathbf{H}(s) a_\omega(-s)$$

and

$$\hat{\mathbf{T}}_\omega(s) = \hat{\mathbf{H}}_\omega(s) + \hat{\mathbf{H}}(s) a_\omega(-s),$$

where

$$\mathbf{H}_\omega(s) = \sum_{j=1}^{n} \frac{\mathbf{c}_j \mathbf{b}_j^T}{s - \lambda_j} a_\omega(\lambda_j) \quad \text{and} \quad \hat{\mathbf{H}}_\omega(s) = \sum_{k=1}^{r} \frac{\mathbf{c}_k \mathbf{b}_k^T}{s - \hat{\lambda}_k} a_\omega(\lambda_k).$$

Here again, these function should match at the opposite of the poles of the reduced-order model. Beside, these conditions are completed with an integral condition due to the optimality condition on the direct feedthrough $\hat{D}$. Note that if one assumes that $\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \hat{D}} = 0$, then the interpolation conditions (3.9) still holds but the expression of $\mathbf{T}_\omega(s)$ and $\hat{\mathbf{T}}_\omega(s)$ are modified as given in Remark 3.3.

Remark 3.3 (Specific case of gradient in $\hat{D}$) *Assuming that $\frac{\partial \mathcal{J}_{\mathcal{L}_{2,\Omega}}}{\partial \hat{D}} = 0$ then (3.9) holds true but $\mathbf{T}_\omega(s)$ and $\hat{\mathbf{T}}_\omega(s)$ are simplified with functions of the form,*

$$\mathbf{T}_\omega(s) = \sum_{j=1}^{n} \mathbf{c}_j \mathbf{b}_j^T \left(-\frac{a_\omega(\lambda_j) - a_\omega(s)}{s - \lambda_j} + \frac{\pi}{2\omega} a_\omega(\lambda_j) a_\omega(s) \right)$$

and

$$\hat{\mathbf{T}}_\omega(s) = \sum_{k=1}^{r} \hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T \left(-\frac{a_\omega(\hat{\lambda}_k) - a_\omega(s)}{s - \hat{\lambda}_k} + \frac{\pi}{2\omega} a_\omega(\hat{\lambda}_k) a_\omega(s) \right).$$

As the reduced-order model does not appear explicitly in (3.9), a fixed-point scheme as the one developed for the $\mathcal{H}_2$ case Beattie and Gugercin (2012) does not clearly appear here. Moreover, connections with any Krylov-like subspace, which is the justification for fast and scalable algorithm, is not accessible anymore. Still, the idea of mixing Loewner, Krylov and optimisation leaves space for perspectives.

3.5 Conclusions

In this chapter, the frequency limited model approximation problem is addressed through two numerical optimisation procedures. First, the **FL-ISTIA** as a direct extension of the **ISTIA** from Gugercin and Antoulas (2006) is presented. This latter reaches a subset of the $\mathcal{H}_2$ interpolatory conditions, losses the stability preservation, but, to the author's best of knowledge, provides very good performances in practice when shift points are well initialised. Indeed, its efficiency has been highlighted in many industrial applications encountered in the aeronautical domain (see *e.g.* Poussot-Vassal et al., 2013; Meyer et al., 2017). Secondly, the **DARPO** procedure, a real $\mathcal{H}_{2,\Omega}$ (indeed $\mathcal{L}_{2,\Omega}$) oriented model reduction algorithm, is also presented. This latter is described as a descent procedure which converges to the $\mathcal{H}_{2,\Omega}$ first order optimality conditions. These last conditions have been shown to be equivalent to some interpolatory ones, as in the unbounded $\mathcal{H}_2$ case, but including more complex (actually irrational) transfer functions. This last point is, to the author's feeling, a nice and interesting research topic for further developments in order to link $\hat{\mathbf{H}}$ with $\hat{\mathbf{T}}_\omega$ in a practical manner, and, if possible, involving Krylov-like subspaces in order to scale-up this method to very large-scale and to close the theoretical contribution.

Chapter 4

Model approximation by input-output delay structured reduced order model

> La chose la plus difficile est de n'accorder aucune importance aux choses qui n'ont aucune importance.
>
> Charles De Gaulle

Contents

Up to now, the approximated model $\hat{\mathbf{H}}$ we were looking for shared the same structure as $\mathbf{H}$, the original one, *i.e.* starting from an **ODE** set given by $\mathcal{S}$, one was looking for a reduced **ODE** set with the same structure and realisation $\hat{\mathcal{S}}$. In this chapter instead, one aims at constructing a reduced order model $\hat{\mathbf{H}}_d$ (equipped with a realisation $\hat{\mathcal{S}}_d$) potentially embedding input-output delays, extending the standard framework. After an introduction and motivation illustrated with an applicative example in Section 4.1, Section 4.2 derives the mismatch error in the specific input-output delay case, putting in evidence an alternate $\mathcal{H}_2$ inner product for delayed models. The fundamental aspect of this new expression of the $\mathcal{H}_2$-inner product in the presence of input and output delays is derived in Section 4.3. Section 4.4 then establishes the $\mathcal{H}_2$ optimality conditions as interpolatory ones, solving the so-called input-output delay dynamical model approximation problem. Section 4.5 suggests an iterative algorithm permitting to practically compute such an approximation. Section 4.6 gives some hints to connect these extended tangential interpolatory conditions to Sylvester equations. Conclusions are discussed in Section 4.7. This chapter is largely based on the Ph.D. thesis of Pontes (2017) and the contributive paper of Pontes Duff et al. (2018) (note also that the original idea was already published in **`arXiv:1511.05252`**, on November 17th, 2015 by the same authors).

4.1 Motivating example and problem formulation

4.1.1 Forewords

For linear time invariant systems, a standard model order reduction approach is to approximate the transfer function of the system under investigation with a rational function. In general, a reduced-order approximation is considered to be a finite dimensional model. This representation is quite general and a wide range of linear dynamical systems can be converted in this form. However, in some cases, it may be more relevant to find reduced-order models having some more complex structures, such as input and output delay ones. Approximating a large-scale dynamical model by a low order one including (a priori unknown) input-output delays is of particular interest when the original full order model represents a transport phenomenon where the information needs a finite amount of time to be transmitted from a point to another. Examples of this kind of models are present in the domains of fluid mechanics, electronics, thermodynamics, *etc.* For these cases, a reduced model containing an input-output delay structure should well capture the transport behaviour and thus allow finding less complex and more accurate reduced models.

The presence of input-output delays in the approximation model was tackled in Halevi (1996), exploiting both Sylvester and Lyapunov equations and gramians properties (derived in Hyland and Bernstein (1995) for the case without delay). The bottleneck of this approach is that it requires solving Lyapunov equations which might be costly in the large-scale context. From the moment matching side, Scarciotti and Astolfi (2014) proposed a problem formulation that enables the construction of an approximation which contains very rich delay structure, including state delay (see also Remark 4.1 below), but where the delays and the interpolation points are supposed to be a priori known. From the Loewner framework viewpoint, Pontes Duff et al. (2015a); Schulze and Unger (2016) generalise the principle of Mayo and Antoulas (2007) to the state delay case enabling data-driven interpolation. However, as for the moment matching case, the delays and the interpolation points are also assumed to be a priori fixed.

It is why an alternative pole / residue-based approach has been developed, which enables to derive firstly, the $\mathcal{H}_2$-inner product expression in the presence of input and output delays, and then, the $\mathcal{H}_2$ optimality conditions, treated as interpolation ones.

As it will appear more clearly in the rest of the chapter, the results presented here and based on Pontes Duff et al. (2018), consist in extending the interpolation results of Gugercin et al. (2008) to the case of approximate models with an extended structure, namely, including non-zero input-output delays.

Remark 4.1 (State delayed models) *The state delay structured reduced order model is a direct extension of the standard delay-free case and of the input-output delay one. This problem is largely more complex than the one treated in this chapter. As an illustration, the case of $\mathcal{H}_2$-optimal approximation of a model $\mathbf{H}$ with a first order reduced-order model $\hat{\mathbf{H}}$ and single internal delay, leads to interpolatory conditions expressed as a interpolation conditions of system series, as show in Pontes Duff et al. (2016a).*

4.1.2 Motivating example

Let us provide as motivation, an industrial use-case from **EDF** (French electricity provider) hydro-electrical engineers, illustrating the impact of a delay-based reduced modelling rather than a pure rational one.

Use-case 5 - EDF hydroelectric open-channel model

In Dalmas et al. (2016), an hydro-electrical open-channel model has been described through an irrational model linking the inflow (q_e) and outflow (q_s) to the measured output of the water's height (h). This model is given by the following irrational transfer function

$$\begin{aligned} h(x,s) &= G_e(x,s)q_e(s) - G_s(x,s)q_s(s) \\ G_e(x,s) &= \frac{\lambda_1(s)e^{\lambda_2(s)L+\lambda_1(s)x} - \lambda_2(s)e^{\lambda_1(s)L+\lambda_2(s)x}}{B_0 s(e^{\lambda_1(s)L} - e^{\lambda_2(s)L})} \\ G_s(x,s) &= \frac{\lambda_1(s)e^{\lambda_1(s)x} - \lambda_2(s)e^{\lambda_2(s)x}}{B_0 s(e^{\lambda_1(s)L} - e^{\lambda_2(s)L})} \end{aligned} \tag{4.1}$$

where x stands as the position of the water height measurement and where constant values and $\lambda(\cdot)$ functions are extracted from a section of the Rhin's river and are given in Dalmas et al. (2016). After applying the Loewner interpolation (from samples of the model (4.1)) or using the **TF-IRKA** framework, it is possible to obtain $\mathbf{H}$, a finite-order model that well reproduces the behaviour of (4.1). Obviously, according to the water height sensor location, delays naturally appears in the inputs q_e and q_s. It is then natural trying including it in a reduced order model to catch this delay behaviour. By applying the **IO-dITIA** (described later in this chapter), the following results can be obtained.

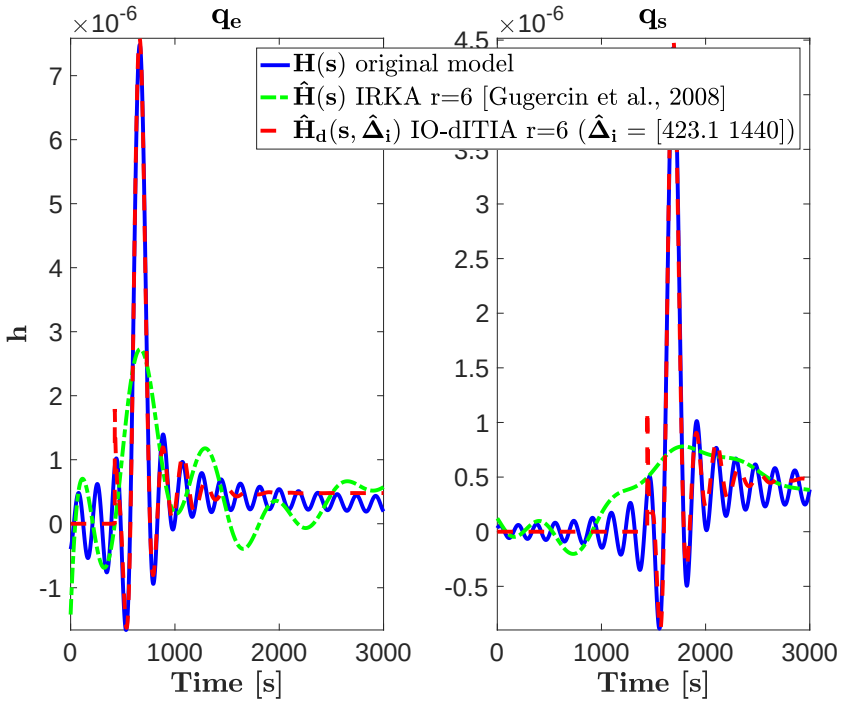


Figure 4.1: Impulse responses of the original model of order $n = 55$ and its approximation using **IRKA** and **IO-dITIA** with an order $r = 6$.

Figure 4.1 clearly shows the benefits of a $\hat{\tau}$ delay structured model compared to the standard one in the time-domain response.

This delay-based model structure has a great importance in many research fields, including modeling of networks, diseases (see *e.g.* Briat and Verriest, 2009), flow transfer (see *e.g.* Shaabani-Ardali et al., 2017) *etc.* Interested reader may also refer to the complete book of Briat (2015), the manuscript of Seuret (2017). Let us now formalise the problem considered in this chapter.

4.1.3 Input-output delayed structured reduced order problem

In what follows, let us consider $\mathbf{H} \in \mathcal{H}_2$, the delay-free original model

$$\mathbf{H}(s) = \hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B}, \tag{4.2}$$

which realisation can be written as $\mathcal{S} : (E, A, B, C, 0, I_{n_u}, I_{n_y})$[1], and the input-output delay reduced order model,

$$\hat{\mathbf{H}}_d(s) = \hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s)\hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B}\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s) = \hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s)\hat{\mathbf{H}}(s)\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s), \tag{4.3}$$

[1]This notation stands as the classical realization one $\mathcal{S} : (E, A, B, C, 0)$, extended with the matrices defining the input and output delays. Last terms given as I_{n_u}, I_{n_y} means that no input-output delay are considered.

where $\mathbf{\Delta_i}(s)$ and $\mathbf{\Delta_o}(s)$ are the input and output fixed delayed operators, defined as:

$$\left\{ \begin{array}{rcl} \hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s) & = & \mathbf{diag}(e^{-\hat{\tau}_1 s}, \ldots, e^{-\hat{\tau}_{n_u} s}) \in \mathcal{H}_\infty^{n_u \times n_u} \\ \hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s) & = & \mathbf{diag}(e^{-\hat{\gamma}_1 s}, \ldots, e^{-\hat{\gamma}_{n_y} s}) \in \mathcal{H}_\infty^{n_y \times n_y} \end{array} \right. , \tag{4.4}$$

where $\{\hat{\tau}_i\}_{i=1}^{n_u} \in \mathbb{R}_+$ and $\{\hat{\gamma}_o\}_{o=1}^{n_y} \in \mathbb{R}_+$. The associated realisation is written as $\hat{\mathcal{S}}_d : (\hat{E}, \hat{A}, \hat{B}, \hat{C}, 0, \hat{\mathbf{\Delta}}_{\boldsymbol{i}}, \hat{\mathbf{\Delta}}_{\boldsymbol{o}})$. Then, the considered input-output delay $\mathcal{H}_2$ model approximation problem is recalled hereafter.

Problem 4.1: Input-output delay $\mathcal{H}_2$-optimal model approximation

Given $\mathbf{H} \in \mathcal{H}_2$ with realization $\mathcal{S}$, a n-th order system without input-output delays $\mathcal{S} : (E, A, B, C, 0, I_{n_u}, I_{n_y})$ as in (4.2), find a reduced r-th order (with $r \ll n$) multiple input-output delays model $\hat{\mathbf{H}}$ with realisation $\hat{\mathcal{S}}_d : (\hat{E}, \hat{A}, \hat{B}, \hat{C}, \hat{\mathbf{\Delta}}_{\boldsymbol{i}}, \hat{\mathbf{\Delta}}_{\boldsymbol{o}})$ as in (4.3) such that

$$\hat{\mathbf{H}}_d = \underset{\substack{\mathbf{G}_d \in \mathcal{H}_2 \\ \mathbf{dim}(\mathbf{G}_d) \leq r}}{\mathbf{argmin}} \quad ||\mathbf{H} - \mathbf{G}_d||_{\mathcal{H}_2}.$$

This search for an optimal solution is carried out assuming that both $\mathbf{H}$ and $\hat{\mathbf{H}}$ from (4.2) and (4.3) are real systems and have semi-simple poles *i.e.* such that their respective transfer function matrix can be decomposed as

$$\mathbf{H}(s) = \sum_{j=1}^{n} \frac{\mathbf{c}_j \mathbf{b}_j^T}{s - \lambda_j} \qquad \text{and} \qquad \hat{\mathbf{H}}(s) = \sum_{k=1}^{r} \frac{\hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T}{s - \hat{\lambda}_k}, \tag{4.5}$$

where $\{\mathbf{b}_j\}_{j=1}^n, \{\hat{\mathbf{b}}_k\}_{k=1}^r \in \mathbb{C}^{n_u}$ and $\{\mathbf{c}_j\}_{j=1}^n, \{\hat{\mathbf{c}}_k\}_{k=1}^r \in \mathbb{C}^{n_y}$. The poles $\{\lambda_j\}_{j=1}^n, \{\hat{\lambda}_k\}_{k=1}^r$ are elements of $\mathbb{C}_-$ so that $\mathbf{H}$ and $\hat{\mathbf{H}}$ belong to $\mathcal{H}_2$ and $\mathbf{dim}(\mathbf{H}) = n$, $\mathbf{dim}(\hat{\mathbf{H}}) = r$.

Remark 4.2 (Redundancy of the delays) *In (4.3), the number of delay parameters is $n_u + n_y$, and the number of **SISO** sub-transfer functions defining the **MIMO** system is $n_u \times n_y$. If $n_u \times n_y < n_u + n_y$,* i.e. *the number of delay parameters is larger than the number of **SISO** subsystems, Problem 4.1 is over-parametrised with respect to the number of delays. This leads to a non-uniqueness of the optimal delay parameters. To resolve this issue, one may restrict the number of delay parameters in the reduced order model. This topic is discussed later in Remark 4.9.*

4.2 Mismatch error formulation

To treat Problem 4.1, let us derive the mismatch error. First, let us remind the following proposition.

Proposition 4.1: $\mathcal{H}_2$-norm invariance

Let $\hat{\mathbf{H}} \in \mathcal{H}_2$ be a stable dynamical system and M, N be two elements of $\mathcal{H}_\infty^{n_u \times n_u}$ and $\mathcal{H}_\infty^{n_y \times n_y}$ respectively such that,

$$\forall \omega \in \mathbb{R}, \quad \overline{M(\imath\omega)} M^T(\imath\omega) = I_{n_u}, \quad \overline{N(\imath\omega)} N^T(\imath\omega) = I_{n_y}. \tag{4.6}$$

It follows that, if $\hat{\mathbf{H}}_d = N \hat{\mathbf{H}} M$, then $||\hat{\mathbf{H}}_d||_{\mathcal{H}_2} = ||\hat{\mathbf{H}}||_{\mathcal{H}_2}$.

One can easily check that condition (4.6) appearing in Proposition 4.1 is satisfied by the delay matrices of (4.4) when $M = \hat{\mathbf{\Delta}}_{\boldsymbol{i}}$ and $N = \hat{\mathbf{\Delta}}_{\boldsymbol{o}}$. In other words, the $\mathcal{H}_2$-norm does not depend neither on the input, nor the output delays. The following result makes now explicit the calculation of the $\mathcal{H}_2$-norm associated with the dynamical mismatch gap $||\mathbf{H} - \hat{\mathbf{H}}_d||_{\mathcal{H}_2}$, which conditions Problem 4.1. Let $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{H}_2$ be real systems such that $\hat{\mathbf{H}}$ is given by (4.5). The $\mathcal{H}_2$-norm of the approximation gap (or mismatch error), denoted

by $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d)$, can be expressed as

$$\begin{aligned}\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d) &= ||\mathbf{H}-\hat{\mathbf{H}}_d||^2_{\mathcal{H}_2}\\ &= ||\mathbf{H}-\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}||^2_{\mathcal{H}_2}\\ &= ||\mathbf{H}||^2_{\mathcal{H}_2}-2\langle\mathbf{H},\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}\rangle_{\mathcal{H}_2}+||\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}||^2_{\mathcal{H}_2}\\ &= ||\mathbf{H}||^2_{\mathcal{H}_2}-2\langle\mathbf{H},\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}\rangle_{\mathcal{H}_2}+||\hat{\mathbf{H}}||^2_{\mathcal{H}_2}.\end{aligned}$$

With reference to the above criteria, one can consider that the $\mathcal{H}_2$-norm mismatch computation in presence of delays is available as in the $\mathcal{H}_2$ case. However, the $\mathcal{H}_2$ inner product, involving $\mathcal{H}_\infty$ operators ($\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}$) still must be carefully computed. This is the purpose of the following section.

4.3 Input-output delayed $\mathcal{H}_2$ inner product

For a didactical understanding, let us compute the input-output delayed $\mathcal{H}_2$ inner product first in the **SISO** case, then in the **MIMO** one.

Proposition 4.2: Single input delay $\mathcal{H}_2$ inner product

Let $\mathbf{H},\hat{\mathbf{H}}\in\mathcal{H}_2$ be two **SISO** real systems with transfer functions as

$$\mathbf{H}(s)=\sum_{j=1}^{n}\frac{\Phi_j}{s-\lambda_j}\quad\text{and}\quad\hat{\mathbf{H}}(s)=\sum_{k=1}^{r}\frac{\hat{\Phi}_k}{s-\hat{\lambda}_k},$$

and let $\tau\in\mathbb{R}_+$, $\hat{\mathbf{H}}_d=\hat{\mathbf{H}}e^{-\hat{\tau}s}$. The inner product is given as

$$\langle\mathbf{H},\hat{\mathbf{H}}_d\rangle_{\mathcal{H}_2}=\sum_{j=1}^{n}\Phi_j\hat{\mathbf{H}}(-\lambda_j)e^{\hat{\tau}\lambda_j}.$$

The inner product evaluation is then obtained as the evaluation of the reduced order model at the mirror images of the poles of the original one, weighted by the exponential (delay) operator. Similarly, the **MIMO** case can be formulated as follows.

Proposition 4.3: Multiple input-output delays $\mathcal{H}_2$ inner product

Let $\mathbf{H},\hat{\mathbf{H}}\in\mathcal{H}_2$ be two real systems with transfer functions as in (4.5). Moreover, let $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}\in\mathcal{H}_\infty^{n_y\times n_y}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\in\mathcal{H}_\infty^{n_y\times n_y}$ be real transfer functions satisfying $\sup_{s\in\mathbb{C}_+}\{||\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(s)||_F,||\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(s)||_F\}=c<+\infty$. By denoting $\hat{\mathbf{H}}_d=\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$, the inner product $\langle\hat{\mathbf{H}}_d,\mathbf{H}\rangle_{\mathcal{H}_2}$ is expressed as

$$\begin{aligned}\langle\mathbf{H},\hat{\mathbf{H}}_d\rangle_{\mathcal{H}_2} &= \sum_{j=1}^{n}\mathbf{tr}\left(\mathbf{res}\left[\hat{\mathbf{H}}_d(-s)\mathbf{H}^T(s),\lambda_j\right]\right)\\ &= \sum_{j=1}^{n}\mathbf{c}_j^T\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j)\hat{\mathbf{H}}(-\lambda_j)\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j)\mathbf{b}_j.\end{aligned}$$

Proof 4.1 (of Proposition 4.3) *is detailled in Pontes Duff et al. (2018).*

One may note that Proposition 4.3 is a generalisation of (Gugercin et al., 2008, Lemma 3.5) in the case of **MIMO** systems with multiple-input-output delays.

Remark 4.3 (Rational functions case symmetry) *An equivalent proposition was derived in the rational function case in the seminal paper of Gugercin et al. (2008). Obviously, it can be recovered from Proposition 4.3 by taking $\boldsymbol{\Delta}_{\boldsymbol{i}}=I_{n_u}$ and $\boldsymbol{\Delta}_{\boldsymbol{o}}=I_{n_y}$. The result corresponds to the symmetric expression of the*

inner product i.e. *the evaluation of* $\mathbf{H}$ *in the poles of* $\hat{\mathbf{H}}$ *and its associated residues* $\hat{\mathbf{c}}_k$ *and* $\hat{\mathbf{b}}_k$ *such that,*

$$\langle \mathbf{H}, \hat{\mathbf{H}} \rangle_{\mathcal{H}_2} = \sum_{k=1}^{r} \hat{\mathbf{c}}_k^T \mathbf{H}(-\hat{\lambda}_k)\hat{\mathbf{b}}_k = \sum_{j=1}^{n} \mathbf{c}_j^T \hat{\mathbf{H}}(-\lambda_j)\mathbf{b}_j = \langle \hat{\mathbf{H}}, \mathbf{H} \rangle_{\mathcal{H}_2}.$$

In the presence of input-output delays, as the $\mathcal{H}_2$*-norm cannot be approximated using one contour containing the poles of* $\hat{\mathbf{H}}_d$ *only, this result is no longer true. Indeed, it can be easily shown that in this case, the integral over* Γ_R[2] *will depend on a positive exponential argument which will not converge to* 0_+ *when* $R \to +\infty$*. This justifies the assumption* $\sup_{s\in\mathbb{C}_+}\{||\hat{\mathbf{\Delta}}_{\boldsymbol{o}}(s)||_F, ||\hat{\mathbf{\Delta}}_{\boldsymbol{i}}(s)||_F\} = c < +\infty$ *and importance of Proposition 4.3.*

This non-symmetric expression of the $\mathcal{H}_2$-inner product in the presence of input-output delays is the crucial point in the derivation of the optimality conditions[3]. Indeed, in the rational function case described by Gugercin et al. (2008), this symmetry exists and is used to derive the optimality conditions (see also Chapter 2). That is why Proposition 4.3 has to be invoked when deriving the optimality conditions for Problem 4.1. Now, let us recall the pole / residue $\mathcal{H}_2$-norm formula from Gugercin et al. (2008) and its immediate extension to the delayed case.

Corollary 4.1: Pole / residue $\mathcal{H}_2$-norm

Given $\hat{\mathbf{H}} \in \mathcal{H}_2$, expressed as

$$\hat{\mathbf{H}}(s) = \sum_{k=1}^{r} \frac{\hat{\mathbf{c}}_k \hat{\mathbf{b}}_k^T}{s - \hat{\lambda}_k}$$

then,

$$||\hat{\mathbf{H}}||_{\mathcal{H}_2}^2 = \sum_{k=1}^{r} \hat{\mathbf{c}}_k^T \hat{\mathbf{H}}(-\hat{\lambda}_k)\hat{\mathbf{b}}_k.$$

Using Proposition 4.1, for the delay operators $\hat{\mathbf{\Delta}}_{\boldsymbol{i}}$ and $\hat{\mathbf{\Delta}}_{\boldsymbol{o}}$ as in (4.4), we immediately establish

$$||\hat{\mathbf{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\mathbf{\Delta}}_{\boldsymbol{i}}||_{\mathcal{H}_2}^2 = \sum_{k=1}^{r} \hat{\mathbf{c}}_k^T \hat{\mathbf{H}}(-\hat{\lambda}_k)\hat{\mathbf{b}}_k.$$

Proof 4.2 (of Corollary 4.1) *is immediate from Gugercin et al. (2008).*

Now the $\mathcal{H}_2$ inner-product in the input-output delays context has been settled, let us state the $\mathcal{H}_2$ optimality conditions related to Problem 4.1, as an interpolation one.

4.4 $\mathcal{H}_2$ optimality conditions as interpolatory conditions

Considering the mathematical formulation of Problem 4.1 and the reduced order system structure $\hat{\mathbf{H}}_d = \hat{\mathbf{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\mathbf{\Delta}}_{\boldsymbol{i}}$, where $\hat{\mathbf{H}}$ is given as in (4.5), the underlying optimisation issue to be solved is parameterised by:

$$\begin{cases} \{\hat{\lambda}_k\}_{k=1}^{r} & \in \mathbb{C}_- & \text{the } r \text{ poles} \\ \left(\{\hat{\mathbf{b}}_k\}_{k=1}^{r}, \{\hat{\mathbf{c}}_k\}_{k=1}^{r}\right) & \in (\mathbb{C}^{n_u} \times \mathbb{C}^{n_y}) & \text{the } r \text{ bi-tangential directions} \\ \left(\{\hat{\tau}_i\}_{i=1}^{n_u}, \{\hat{\gamma}_o\}_{o=1}^{n_y}\right) & \in \left(\mathbb{R}_+^{n_u} \times \mathbb{R}_+^{n_y}\right) & \text{the } n_u + n_y \text{ delay values.} \end{cases}$$

Consequently, similarly to the $\mathcal{H}_2$ case presented in Chapter 2, or the $\mathcal{H}_{2,\Omega}$ (or $\mathcal{L}_{2,\Omega}$) case described in Chapter 3, our primary objective consists in rewriting the expression of the $\mathcal{H}_2$ gap $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d)$ as a function of these parameters, *i.e.* as $\mathcal{J}_{\mathcal{H}_2}(\hat{\lambda}_k, \hat{\mathbf{b}}_k, \hat{\mathbf{c}}_k, \hat{\tau}_i, \hat{\gamma}_o)$, which will subsequently facilitate the derivation of the

[2] Γ_R denotes the contour of the right complex plane, invoked when the residue Theorem is applied (see Pontes, 2017, Chapter 5).
[3] Note that the term "non-symmetric" is not dedicated to the $\mathcal{H}_2$ inner product, but rather to its expression.

$\mathcal{H}_2$ optimality conditions for Problem 4.1. From the preliminary results, $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d)$ can be equivalently rewritten as

$$\begin{aligned}\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d) &= ||\mathbf{H}||^2_{\mathcal{H}_2} + ||\hat{\mathbf{H}}||^2_{\mathcal{H}_2} - 2\langle \mathbf{H}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}\rangle_{\mathcal{H}_2} \\ &= ||\mathbf{H}||^2_{\mathcal{H}_2} + \sum_{k=1}^{r} \hat{\mathbf{c}}_k^T\hat{\mathbf{H}}(-\hat{\lambda}_k)\hat{\mathbf{b}}_k - 2\sum_{j=1}^{n}\mathbf{c}_j^T\,\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j)\hat{\mathbf{H}}(-\lambda_j)\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j)\mathbf{b}_j \\ &= \mathcal{J}_{\mathcal{H}_2}(\hat{\lambda}_k, \hat{\mathbf{b}}_k, \hat{\mathbf{c}}_k, \hat{\tau}_i, \hat{\gamma}_o).\end{aligned} \tag{4.7}$$

From equation (4.7), the first-order optimality conditions related to $\mathcal{J}_{\mathcal{H}_2}(\hat{\lambda}_k, \hat{\mathbf{b}}_k, \hat{\mathbf{c}}_k, \hat{\tau}_i, \hat{\gamma}_o)$ minimisation, can be analytically computed. The gradient expressions of the $\mathcal{H}_2$ gap with respect to each parameters (delays, tangential directions and poles) are presented in what follows.

Result 4.1: Gradient computation of $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d)$

Let $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{H}_2$ be systems with transfer functions as in (4.5) and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}} \in \mathcal{H}_\infty^{n_u\times n_u}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}} \in \mathcal{H}_\infty^{n_y\times n_y}$ be the delay operators defined in (4.4). The gradients of the $\mathcal{H}_2$ gap $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}) = \mathcal{J}_{\mathcal{H}_2}(\hat{\lambda}_k, \hat{\mathbf{b}}_k, \hat{\mathbf{c}}_k, \hat{\tau}_i, \hat{\gamma}_o)$, shortly denoted $\mathcal{J}_{\mathcal{H}_2}$ with respect to the delays $\hat{\tau}_i$ and $\hat{\gamma}_o$ are expressed as

$$\begin{cases} \dfrac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{\tau}_i} &= -2\dfrac{\partial\langle \hat{\mathbf{H}}_d, \mathbf{H}\rangle_{\mathcal{H}_2}}{\partial \hat{\tau}_i} \\ &= -2\displaystyle\sum_{j=1}^{n}\lambda_j\mathbf{c}_j^T\,\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j)\hat{\mathbf{H}}(-\lambda_j)\mathbf{D}_i\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j)\mathbf{b}_j, \\ \dfrac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{\gamma}_o} &= -2\dfrac{\partial\langle \hat{\mathbf{H}}_d, \mathbf{H}\rangle_{\mathcal{H}_2}}{\partial \hat{\gamma}_m} \\ &= -2\displaystyle\sum_{j=1}^{n}\lambda_j\mathbf{c}_j^T\mathbf{D}_o\,\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j)\hat{\mathbf{H}}(-\lambda_j)\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j)\mathbf{b}_j, \end{cases}$$

for $i = 1, \ldots, n_u$ and $o = 1, \ldots, n_y$, where elements of $\mathbf{D}_i \in \mathbb{R}^{n_u\times n_u}$, $\mathbf{D}_o \in \mathbb{R}^{n_y\times n_y}$, are defined as

$$[\mathbf{D}_k]_{ij} = \delta_{ijk} = \begin{cases} 1 & \text{if } i = j = k \\ 0 & \text{otherwise} \end{cases}.$$

The gradients of $\mathcal{J}_{\mathcal{H}_2}(\hat{\mathbf{H}}_d)$ with respect to parameters $\{\hat{\lambda}_k, \hat{\mathbf{b}}_k, \hat{\mathbf{c}}_k\}_{k=1}^r$, are expressed as

$$\begin{cases} \dfrac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{\lambda}_k} &= 2\hat{\mathbf{c}}_k^T\left(\mathbf{T}_d'(-\hat{\lambda}_k) - \hat{\mathbf{H}}'(-\hat{\lambda}_k)\right)\hat{\mathbf{b}}_k \\ \dfrac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{\mathbf{b}}_k} &= -2\hat{\mathbf{c}}_k^T\left(\mathbf{T}_d(-\hat{\lambda}_k) - \hat{\mathbf{H}}(-\hat{\lambda}_k)\right) \\ \dfrac{\partial \mathcal{J}_{\mathcal{H}_2}}{\partial \hat{\mathbf{c}}_k} &= -2\hat{\mathbf{b}}_k^T\left(\mathbf{T}_d(-\hat{\lambda}_k) - \hat{\mathbf{H}}(-\hat{\lambda}_k)\right)^T, \end{cases}$$

where

$$\mathbf{T}_d(s) = \sum_{j=1}^{n}\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j)\frac{\mathbf{c}_j^T\mathbf{b}_j}{s-\lambda_j}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j), \tag{4.8}$$

and where $\mathbf{T}_d'$ and $\hat{\mathbf{H}}'$ are the derivative of $\mathbf{T}_d$ and $\hat{\mathbf{H}}$, respectively.

Proof 4.3 (of Result 4.1) *see Pontes Duff et al. (2018).*

In the **SISO** case, as the tangential directions vanishes, all conditions provided in Result 4.1 appear much simpler. Then, the following main result for **SISO** systems is given.

Theorem 4.1: Single input delay structured $\mathcal{H}_2$ optimality conditions

Given $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{H}_2$ as,

$$\mathbf{H}(s) = \sum_{j=1}^{n} \frac{\Phi_j}{s - \lambda_j} \quad \text{and} \quad \hat{\mathbf{H}}(s) = \sum_{k=1}^{r} \frac{\hat{\Phi}_k}{s - \hat{\lambda}_k},$$

such that $\hat{\mathbf{H}}_d = \hat{\mathbf{H}} e^{-\hat{\tau} s}$ is a local optimum of Problem 4.1, then the following interpolatory and delay conditions, for all $l = 1, \ldots, r$, hold true:

$$\begin{cases} \hat{\mathbf{H}}(-\hat{\lambda}_l) &= \mathbf{T}_d(-\hat{\lambda}_l), \\ \hat{\mathbf{H}}'(-\hat{\lambda}_l) &= \mathbf{T}'_d(-\hat{\lambda}_l), \end{cases}$$

$$\sum_{j=1}^{n} \lambda_j \Phi_j \left(\sum_{k=1}^{r} \frac{\hat{\Phi}_k}{\lambda_j + \hat{\lambda}_k} \right) e^{\hat{\tau} \lambda_j} = 0,$$

and where

$$\mathbf{T}_d(s) = \sum_{j=1}^{n} \frac{\Phi_j}{s - \lambda_j} e^{\hat{\tau} \lambda_j}.$$

Then, Theorem 4.2 gathers all the first-order optimality conditions related to Problem 4.1 for the **MIMO** case.

Theorem 4.2: Multiple input-output delays structured $\mathcal{H}_2$ optimality conditions

Let us consider $\mathbf{H} \in \mathcal{H}_2$ whose transfer function is given by (4.5). Let $\hat{\mathbf{H}}_d = \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}} \hat{\mathbf{H}} \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ be a local optimum of Problem 4.1. It is assumed that $\hat{\mathbf{H}} \in \mathcal{H}_2$ is given by (4.5). Let $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}$ be elements of $\mathcal{H}_\infty^{n_u \times n_u}$ and $\mathcal{H}_\infty^{n_y \times n_y}$, respectively, such that Propositions 4.1 and 4.3 are verified. Then, the following equalities hold:

$$\begin{cases} \hat{\mathbf{H}}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l &= \mathbf{T}_d(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \\ \hat{\mathbf{c}}_l^T \hat{\mathbf{H}}(-\hat{\lambda}_l) &= \hat{\mathbf{c}}_l^T \mathbf{T}_d(-\hat{\lambda}_l) \\ \hat{\mathbf{c}}_l^T \hat{\mathbf{H}}'(-\hat{\lambda}_l)\mathbf{b}_l &= \hat{\mathbf{c}}_l^T \mathbf{T}'_d(-\hat{\lambda}_l)\hat{\mathbf{b}}_l, \end{cases} \tag{4.9}$$

$$\begin{cases} \sum_{j=1}^{n} \lambda_j \mathbf{c}_j^T \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j) \hat{\mathbf{H}}(-\lambda_j) \mathbf{D}_i \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j) \mathbf{b}_j &= 0 \\ \sum_{j=1}^{n} \lambda_j \mathbf{c}_j^T \mathbf{D}_o \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}(-\lambda_j) \hat{\mathbf{H}}(-\lambda_j) \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}(-\lambda_j) \mathbf{b}_j &= 0, \end{cases} \tag{4.10}$$

for all $l = 1, \ldots, r$, $i = 1, \ldots, n_u$ and $o = 1, \ldots, n_y$ and where $\mathbf{T}_d(s)$ is given by (4.8).

Proof 4.4 (of Theorem 4.2) *is immediate from Result 4.1 (see also Pontes Duff et al., 2018).*

Theorem 4.2 asserts that any solution of the $\mathcal{H}_2$ model approximation Problem 4.1, denoted as $\hat{\mathbf{H}}_d = \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}} \hat{\mathbf{H}} \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ is such that $\hat{\mathbf{H}}$ satisfies, at the same time, a set of $3r$ bi-tangential interpolation conditions detailed in (4.9) and another set of $n_u + n_y$ relations on the delays contained in the $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}$ diagonal matrices given by (4.10). Note that the interpolation conditions (4.9) are very similar to the optimality conditions for the $\mathcal{H}_2$ approximation with rational functions presented in Chapter 2. However, in the case with input-output delays, the new model $\mathbf{T}_d$ defined in (4.8) enters in the game, instead of $\mathbf{H}$. This new system possesses the same poles as $\mathbf{H}$, but differs from $\mathbf{H}$ by its residues, computed as (4.8), where delays play the role of weights in their impact. One should note, this new model is intrinsically related to the non-symmetric expression of the $\mathcal{H}_2$-inner product (see Remark 4.3).

Remark 4.4 (Optimality conditions equivalence) *By considering* $\mathbf{T}_d$ *as in* (4.8) *and the following rational function problem: find* $\hat{\mathbf{H}} \in \mathcal{H}_2$ *a reduced* r*-th order (without delay) approximation which minimises* $||\mathbf{T}_d - \hat{\mathbf{H}}||_{\mathcal{H}_2}$*, then, if* $\hat{\mathbf{H}}$ *is a local minimum of this problem, then it satisfies the interpolation conditions* (4.9)*. Thus, for fixed input and output delays, this problem and Problem 4.1 both lead to the same optimality conditions and have the same stationary points.*

Based on the above conditions and remark, let us now derive the **IO-dITIA** (for Input-Output delay Iterative Tangential Interpolation Algorithm), an interpolation-based algorithm, similar to the **MIMO IRKA** one, to numerically compute the approximation $\hat{\mathbf{H}}_d$.

4.5 The IO-dITIA procedure and computational considerations

4.5.1 Some computational considerations

Let us assume that $\hat{\mathbf{H}}_d = \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ is a local minimum of Problem 4.1, satisfying assumptions of Theorem 4.2. Then the following observations can be made.

Remark 4.5 (Fixed poles and delay values) *If the matrices* $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}$*,* $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ *and the reduced order model poles* $\hat{\lambda}_1, \ldots, \hat{\lambda}_r$ *are assumed to be known, Problem 4.1 is reduced to a much simpler problem that can be solved, for example, by using the Loewner framework (*e.g. *using Mayo and Antoulas, 2007).*

Remark 4.6 (Fixed delay values) *If the delay matrices* $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}$*,* $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}$ *are known, then Problem 4.2 can be solved by finding a model realization* $\hat{\mathbf{H}}$*, which satisfies the interpolation conditions* (4.9) *of Theorem 4.2, only. This can be done using the* ***MIMO IRKA*** *(see Gugercin et al., 2008).*

Remark 4.7 (Fixed realisation) *Assume that the system realisation* $\hat{\mathcal{S}}$ *has already been determined, it follows that Problem 4.2 is equivalent to look for optimal delays matrices* $(\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}) \in (\mathcal{H}_\infty^{n_y \times n_y} \times \mathcal{H}_\infty^{n_u \times n_u})$ *such that*

$$(\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}) = \underset{(\tilde{\boldsymbol{\Delta}}_o, \tilde{\boldsymbol{\Delta}}_i)}{\mathbf{argmax}} \langle \tilde{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\tilde{\boldsymbol{\Delta}}_{\boldsymbol{i}}, \mathbf{H} \rangle_{\mathcal{H}_2}. \tag{4.11}$$

Interestingly, since $\langle \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}\hat{\mathbf{H}}\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}, \mathbf{H} \rangle_{\mathcal{H}_2} \to 0$ *when the delays go to infinity, this problem can be restricted to a compact set and thus a global solution exists. The delay optimisation will be illustrated later in the chapter.*

4.5.2 The IO-dITIA procedure

Now all the necessary theoretical results and computational remarks have been made, an algorithm allowing to numerically compute a model $\hat{\mathbf{H}}_d$ satisfying the previous $\mathcal{H}_2$ optimality conditions is proposed in this subsection. It relies on Remarks 4.4, 4.5, 4.6 and 4.7 and the optimality conditions defined above. Therefore, the proposed approach corresponds to an iterative algorithm in which each iteration can be decomposed in two steps. The first one aims at computing a realisation $\hat{\mathbf{H}}$ which satisfies the interpolation conditions (4.9) while fixing the matrices $(\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}})$ at their values obtained from the previous iteration. This can be done using, for instance, an implementation of the original **MIMO IRKA** of Gugercin et al. (2008) (step 4). In the second step, $\hat{\mathbf{H}}$ is therefore fixed and the optimal values for the $(\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}})$ matrices elements are determined (step 5). This sequential procedure can be summarised in Algorithm 5.

This iterative algorithm is inspired from Gugercin et al. (2008); Van Dooren et al. (2008) and, upon convergence, the first-order necessary conditions of Theorem 4.2 will be satisfied. In practice, one can stop the while loop when the relative changes of the poles of $\hat{\mathbf{H}}^{(i)}$ and the delay blocks $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}^{(i)}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}^{(i)}$ are smaller than a given tolerance.

Remark 4.8 (Delay optimization) *In Algorithm 5, one needs to solve an optimisation problem involving the delay parameters. To this aim, two approaches can be considered:*

Algorithm 5 IO-dITIA - Input-Output delay Iterative Tangential Interpolation Algorithm

Require: $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$, $C \in \mathbb{R}^{n_y\times n}$, $\left\{\sigma_l^{(0)}, \mathbf{r}_l^{(0)}, \mathbf{l}_l^{(0)}\right\}_{l=1}^{r} \in \{\mathbb{C}\times\mathbb{C}^{n_u}\times\mathbb{C}^{n_y}\}$ and initial guesses for both $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}^{(0)}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}^{(0)}$.

1: Set $i = 0$
2: **while** not converged **do**
3: Build $\mathbf{T}_d^{(i)}$ as in (4.8)
4: Build $\hat{\mathbf{H}}^{(i)}$ satisfying the bi-tangential interpolation conditions (4.9) using **MIMO IRKA** (see *e.g.* Gugercin et al., 2008) with $\mathbf{T}_d^{(i)}$ and $\{\sigma_l^{(i)}, \mathbf{r}_l^{(i)}, \mathbf{l}_l^{(i)}\}_{l=1}^{r}$
5: Compute $(\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}^{(i)}, \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}^{(i)})$ which solves (4.11) using $\hat{\mathbf{H}}^{(i)}$
6: Set $\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}^{(i+1)} \leftarrow \hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}^{(i)}$ and $\hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}^{(i+1)} \leftarrow \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}^{(i)}$
7: Set $\{\sigma_l^{(i+1)}, \mathbf{r}_l^{(i+1)}, \mathbf{l}_l^{(i+1)}\}_{l=1}^{r} \leftarrow \{\sigma_l^{(i+1)}, \mathbf{r}_l^{(i+1)}, \mathbf{l}_l^{(i+1)}\}_{l=1}^{r}$
8: Set $i \leftarrow i + 1$
9: **end while**
10: Construct $\hat{\mathbf{H}}_d = \hat{\boldsymbol{\Delta}}_{\boldsymbol{o}}^{(i)}\mathbf{H}^{(i)}\hat{\boldsymbol{\Delta}}_{\boldsymbol{i}}^{(i)}$

Ensure: $\hat{\mathbf{H}}_d$ satisfies the interpolation conditions of Theorem 4.9.

- *One may implement a branch and bound algorithm to search the global optimum in a given interval. Although this approach provides accurate results, it has a high numerical cost and it might be not suitable in large-scale settings.*
- *Otherwise, one should provide a good initialisation of the delay parameters. This initialisation might come from some physical knowledge or by a simple time-domain simulation of the impulse response of* $\mathbf{H}$*, obviously, if this latter is not too large.*

Remark 4.9 (Structured input-output delays) *If one wants to restrict the input and output delays to act only on part of the input and output, one may set some of the* $\hat{\tau}_i^{(i)}$ *and* $\hat{\gamma}_o^{(i)}$ *to zero and optimise only with respect to the remaining variables. For these variables, the gradients derived in derived in Result 4.1 are still valid and hence Algorithm 5 can be employed also for this case.*

Before closing this chapter, let us now illustrate some of the properties and results presented so far using an academic example which mimics a delay behaviour.

Example 9 - Artificial delay

Let us consider a model $\mathbf{H}$ of order $n = 20$, given by the following transfer function

$$\mathbf{H}(s) = \prod_{j=1}^{n} \frac{\lambda_j}{s - \lambda_j},$$

where $\lambda_j \in \mathbb{R}_-$ $(j = 1, \ldots, n)$ are linearly spaced between $[-2, -1]$. The impulse response of $\mathbf{H}$ is given by the solid blue line in Figure 4.2. Interestingly, it naturally behaves as an input delay system. In order to fit the framework proposed in this chapter, input-delay $\mathcal{H}_2$ optimal model $\hat{\mathbf{H}}_d = \hat{\mathbf{\Delta}}_o \hat{\mathbf{H}} \hat{\mathbf{\Delta}}_i$ of order $r = 4$ (dashed red) was obtained by applying Theorem 4.2 and **IO-dITIA** procedure. This model is compared with a pure rational function-based approximation of the same order $r = 4$ obtained with the **IRKA** procedure (green dash dotted). The resulting impulse responses are reported on Figure 4.2.

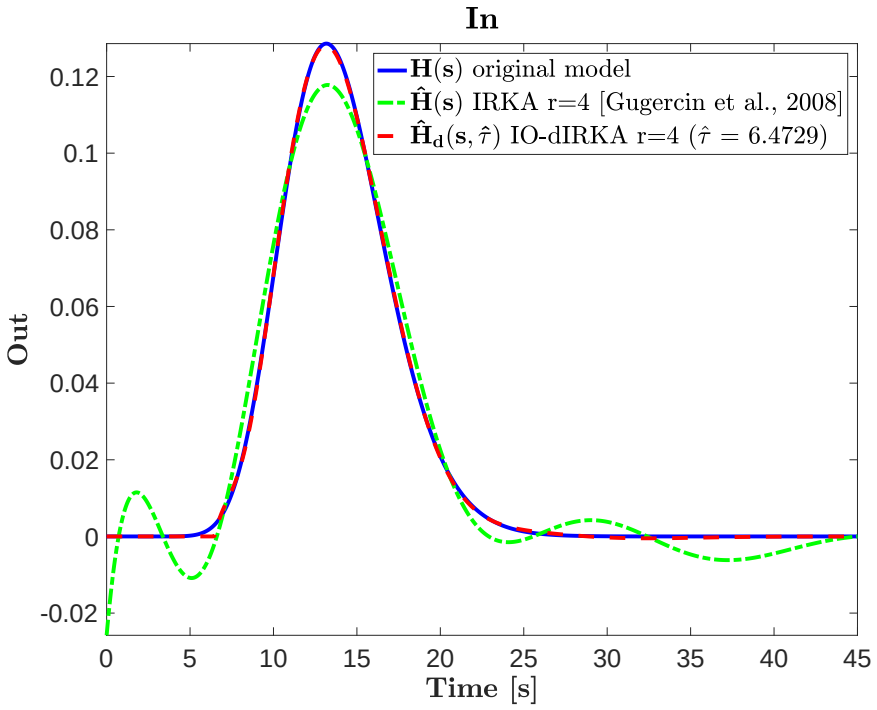


Figure 4.2: Impulse response of the original model $\mathbf{H}$ of order $n = 20$ (solid blue line), the input-delay $\mathcal{H}_2$-optimal model $\hat{\mathbf{H}}_d$ of order $r = 4$ (dashed red line) and the rational $\mathcal{H}_2$-optimal model $\hat{\mathbf{H}}$ of order $r = 4$ obtained with **IRKA** (dash dotted green line).

As clearly shown in Figure 4.2, the proposed methodology allows obtaining an input-delay $\mathcal{H}_2$ approximation of model $\mathbf{H}$ that clearly provides a better matching than the rational case. Note that even for higher orders (here, **IRKA** with $n = 6$ still has a bad matching and exhibits difficulties in accurately catching the delay and main dynamics). Indeed, the rational cases exhibits an oscillatory behaviour during the first seconds while the input-delay model $\hat{\mathbf{H}}_d$ takes benefit of the delay structure to focus on the main dynamical effect. The approximating model $\hat{\mathbf{H}}_d$ satisfies the conditions given in Theorem 4.2.

Then, Figure 4.3 shows the impulse response mismatch errors for these different configurations. For each reduced order models, the mean square absolute error ε of the impulse responses are also computed. The main observation that can be made is that the mismatch error obtained for $\hat{\mathbf{H}}_d$ is lower than the one obtained by a rational model $\hat{\mathbf{H}}$, motivating the use of the specific approximation model delay structure.

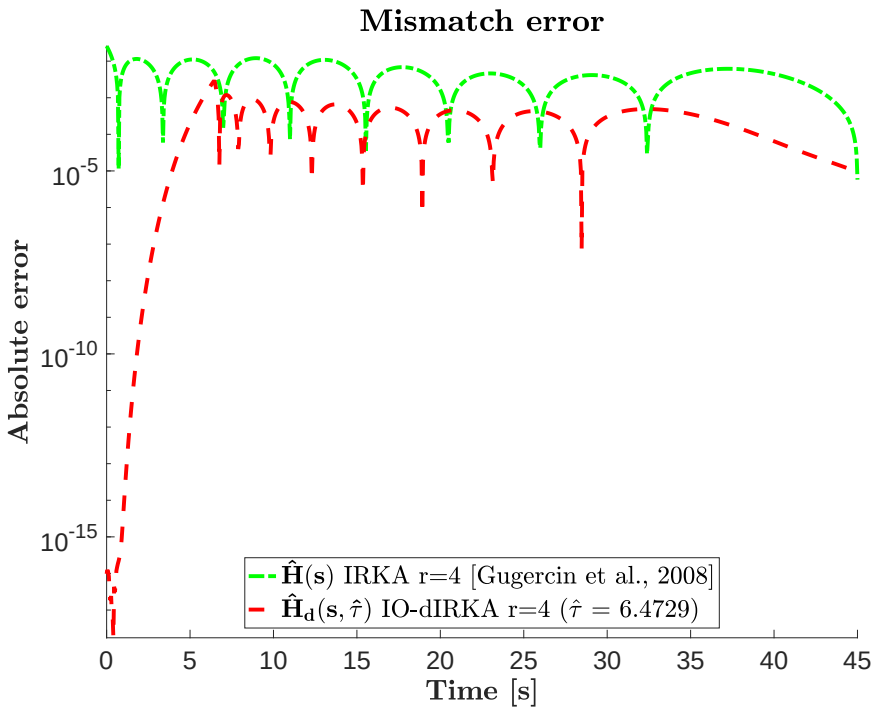


Figure 4.3: Impulse response error between the original model $\mathbf{H}$ of order $n = 20$ and the input-delay $\mathcal{H}_2$-optimal model $\hat{\mathbf{H}}_d$ of order $r = 4$ (dashed red line) and the rational $\mathcal{H}_2$-optimal models $\hat{\mathbf{H}}$ of order $r = 4$ obtained with **IRKA** (dash dotted green line).

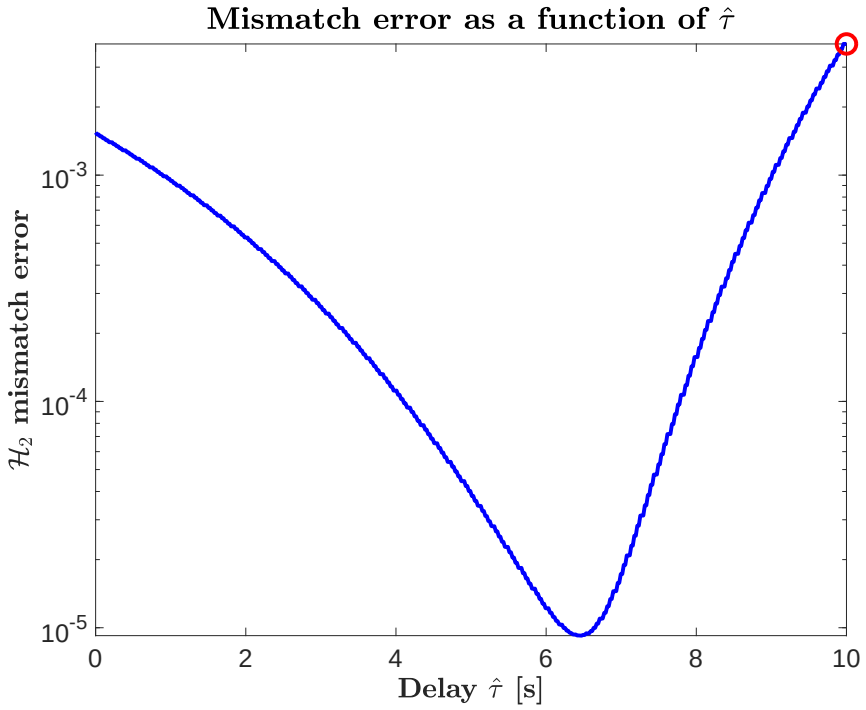


Figure 4.4: $\mathcal{H}_2$ mismatch error as a function of the input delay value.

Finally, as an interesting - still not yet well understood - property, the following Figure 4.4 illustrates

the $\mathcal{H}_2$ mismatch error as a function of the input delay value $\hat{\tau}$. Although the overall behaviour seems predictable with a decreasing error from delay 0 up to the optimal one ($\hat{\tau} \approx 6.4$) and an increase when the optimal one is overpassed, some non monotonic behaviour is also observed in between points, as illustrated on Figure 4.5, highlighting a periodical behaviour, *e.g.* between 4.01 and 4.05 seconds.

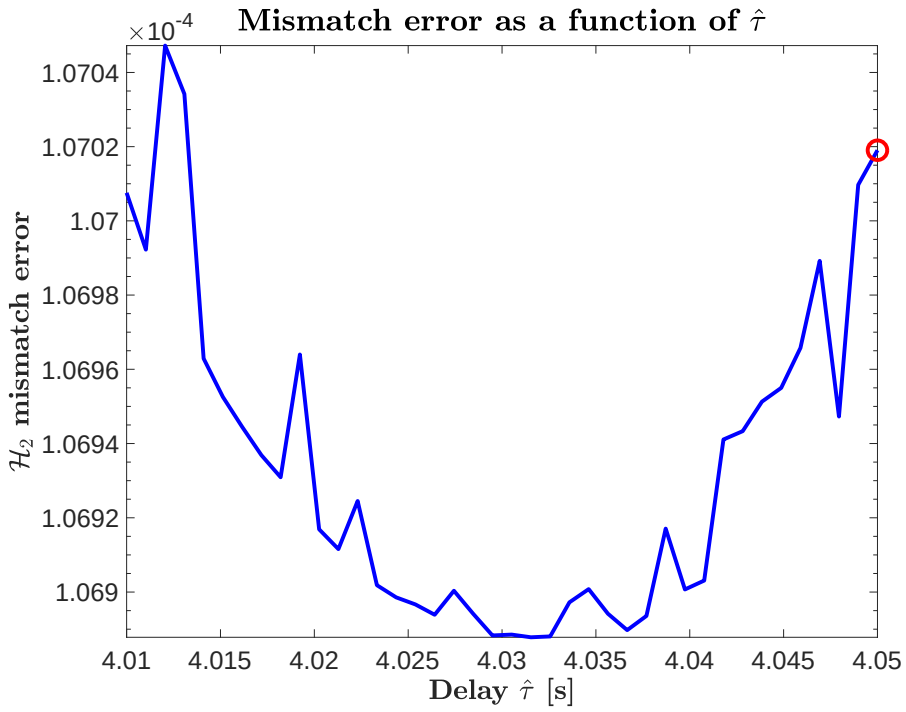


Figure 4.5: $\mathcal{H}_2$ mismatch error as a function of the input delay value for the academic model (zoom on a section).

4.6 Toward connections with the Lyapunov equations

The input-output delays structure in the approximation model was also presented in Halevi (1996) exploiting both Lyapunov equations and gramians properties. In this paper, author did provide the following relevant results.

Theorem 4.3: Gramian-based SISO $\mathcal{H}_2$ optimality conditions

Given $\mathbf{H}, \hat{\mathbf{H}} \in \mathcal{H}_2$ respectively equipped with realisation $\mathcal{S}$ a n-th order system without input-output delays $\mathcal{S} : (I_n, A, B, C, 0, I_{n_u}, I_{n_y})$ and $\hat{\mathcal{S}}$ a r-th order system with input-output delays $\hat{\mathcal{S}} : (I_r, \hat{A}, \hat{B}, \hat{C}, 0, I_{n_u}, \hat{\tau})$, where $\hat{\tau} \in \mathbb{R}_+$. Then, $\hat{\mathcal{S}}$ solves the approximation problem if it satisfies:

$$(\hat{A}, \hat{B}, \hat{C}) = (W_{\hat{\tau}}^T A V_{\hat{\tau}}, W_{\hat{\tau}}^T B, C V_{\hat{\tau}}),$$

where $W_{\hat{\tau}}, V_{\hat{\tau}} \in \mathbb{R}^{n \times r}$ and $\Pi_{\hat{\tau}} \in \mathbb{R}^{n \times n}$ are defined as,

$$W_{\hat{\tau}}^T V_{\hat{\tau}} = I_r \text{ , } \Pi_{\hat{\tau}} = V_{\hat{\tau}} W_{\hat{\tau}}^T \text{ and } \hat{P}_{\hat{\tau}} \hat{Q}_{\hat{\tau}} = V M W^T,$$

where $M \in \mathbb{R}^{r \times r}$ and the non-negative matrices $\hat{P}_{\hat{\tau}}$ and $\hat{Q}_{\hat{\tau}}$ satisfy

$$\begin{aligned} A\hat{P}_{\hat{\tau}} + \hat{P}_{\hat{\tau}}\hat{A}^T + BB^T &= 0 \\ \hat{Q}_{\hat{\tau}} A + \hat{A}^T \hat{Q}_{\hat{\tau}} + e^{A^T \hat{\tau}} C^T C e^{A\hat{\tau}} &= 0 \\ \mathbf{rank}(P_{\hat{\tau}}) = \mathbf{rank}(Q_{\hat{\tau}}) = \mathbf{rank}(P_{\hat{\tau}} Q_{\hat{\tau}}) &= r \end{aligned} \cdot$$

Interestingly, these conditions are not that far to the one presented by Wilson (1974) and reminded in Corollary 2.1. Then, as in the delay-free case presented in Gallivan et al. (2004a,b), where some links between Lyapunov and Sylvester equations with tangential interpolatory conditions were exposed, using the Loewner matrix framework, a similar extension to the delay case may be done in the close future. This last connection has not been derived yet, but may be an interesting issue to elegantly close the result and make the connection between interpolation and Sylvester-like equations.

4.7 Conclusions

In this chapter, the approximation by an input-output delay structured reduced order model has been addressed through the versatile interpolatory framework. The main result is presented in Theorem 4.2 (see also Pontes Duff et al., 2018). This latter involves a new formulation of the $\mathcal{H}_2$ inner product in presence of input and output delays. Besides this extension, one may notice that this result nicely extends the $\mathcal{H}_2$ optimality conditions obtained by Gugercin et al. (2008) for the delay-free case, to the input-output delay reduced order models. Interestingly, the optimal interpolatory conditions now involve a "shifted" transfer function $\mathbf{T}_d$ (weighted by the delay). Attentive reader may remember that $\hat{\mathbf{T}}_\omega$ was entering in the interpolatory frame for the $\mathcal{H}_{2,\Omega}$ case (presented in the precedent Chapter 3) and $\mathbf{H}$ in the $\mathcal{H}_2$ case without delay (recalled in Chapter 2) which points very nice connections, encouraging for future researches. Then, author proposed the **IO-dITIA** inspired from the **MIMO IRKA** and which, as a matter of consequence, shares barely the same strengths and weaknesses. The **IO-dITIA** has been successfully applied to a large number of examples including academic and industrial ones. Besides the algorithmic issues revealed in the present chapter and in Chapter 7, author believes that such result stand as a seminal one in the structured model approximation within the interpolatory framework.

Chapter 5

Model approximation for $\mathcal{L}_2$ functions input-output stability estimation

La difficulté n'est pas de comprendre les idées nouvelles, mais d'échapper aux idées anciennes.

John Maynard Keynes

Contents

Up to this chapter, model reduction and approximation was used in its original purpose, *i.e.* to find a low complexity rational (input-output delayed) model $\hat{\mathbf{H}}(_d)$ which reproduces the behaviour of the (very) large-scale original one $\mathbf{H}$ over (a part of) the frequency space. In this chapter, the concept of rational interpolation induced by the model approximation framework (in Chapters 2, 3 and 4) is used in a totally different objective. Instead, now one aims at establishing the (in)stability of any dynamical system described by a $\mathcal{L}_2$ meromorphic function, through model interpolation-based algorithms. We first motivate the purpose and illustrate the solution proposed on a time-delayed system in Section 5.1. Section 5.2 then both provides the proposed numerical procedures accompanied by theoretical arguments for assessing the stability of any $\mathcal{L}_2$ meromorphic functions. Section 5.3 provides a set of illustrative examples of different nature, and Section 5.4 closes the chapter with conclusions and discussions. Reader should consider this chapter as an opening one and preliminary result description rather than a fully complete result. The contribution still is on-process (on going result by Poussot-Vassal and Vuillemin, 2019). The aim of the chapter is rather to show how versatile and powerful model approximation tools methods can be for a large variety of problems "resolutions".

5.1 Motivating example and problem formulation

5.1.1 Forewords

Stability of a dynamical systems is clearly one of the main property to assess in control theory, numerical simulation, optimisation *etc.* Without loss of generalities, in the case of "classical"[1] **LTI** models either described by a set of **ODE** or **DAE** (including second order ones), the stability problem is recast as the eigenvalues one. In this specific case, the number of eigenvalues is finite and its computational complexity is only related to the (E, A) pencil calculation[2]. If instead, the **LTI** model $\mathbf{H}$ has an infinite number of singularities[3] or its realisation is not necessarily available, the stability assessment becomes much more tedious. As an illustration, one can mention delay invariant **TDS** models or linear **PDE** models (see later in Section 5.3, for a bundle of examples). In these cases, tailored solutions are usually invented to deal with these specificities (*e.g.* the time-delay stability analysis literature is very important and one may refer to Richard (2003); Sipahi et al. (2011); Seuret and Gouaisbaut (2015); Briat (2015)). Still, all these use-cases share a common point: they all are defined by a meromorphic (real-valued) complex function given as

$$\mathbf{H}: \quad \mathbb{C} \quad \mapsto \quad \mathbb{C}^{n_y \times n_u}.$$

Moreover, if $\mathbf{H}$ ensures

$$\int_{-\infty}^{\infty} ||\mathbf{H}(\imath\omega)||_F^2 d\omega < \infty,$$

then the meromorphic function is said to be of finite energy and $\mathbf{H} \in \mathcal{L}_2$. If instead $\mathbf{H}$ ensures

$$\max_{\omega \in \mathbb{R}} ||\mathbf{H}(\imath\omega)||_2 < \infty,$$

then the meromorphic function is said to be of finite energy and $\mathbf{H} \in \mathcal{L}_\infty$. Based on the above considerations and with reference to Chapter 1, let us define the following two input-output stability notions.

Definition 5.1: Input-output $L_\infty - L_2$ stability

A system represented by the transfer function $\mathbf{H}(s)$ is said to be input-output $L_\infty - L_2$ stable, if there exists a $c > 0$ such that:

$$||\mathbf{Hu}||_{L_\infty} = ||\mathbf{y}||_{L_\infty} \leq c||\mathbf{u}||_{L_2}.$$

In this case, the system is said $\mathcal{L}_2$ stable (or $\mathcal{H}_2$).

Definition 5.2: Input-output $L_2 - L_2$ stability

A system represented by the transfer function $\mathbf{H}(s)$ is said to be input-output $L_2 - L_2$ stable, if there exists a $c > 0$ such that:

$$||\mathbf{Hu}||_{L_2} = ||\mathbf{y}||_{L_2} \leq c||\mathbf{u}||_{L_2}.$$

In this case, the system is said $\mathcal{L}_\infty$ stable (or $\mathcal{H}_\infty$).

Here, one proposes to attack the $\mathcal{L}_2$ stability evaluation of meromorphic functions with finite energy, problem from a new angle. Mathematically, the considered problem is given as follows.

Problem 5.1: $\mathcal{L}_2$ meromorphic function stability

Given a meromorphic function $\mathbf{H} \in \mathcal{L}_2$, determine if $\mathbf{H}$ is input-output stable, *i.e.* if, $\mathbf{H} \in \mathcal{H}_2$.

[1] By "classical", ones means equipped with the (E, A, B, C, D) realisation, classically presented in class-room.
[2] In this case, very efficient tools already exist such as **LAPACK** (see also Chapter 1).
[3] In the finite dimensional case, singularities are finite and are the eigenvalues.

Note that this problem formulation is quite large, and as mentioned before, this chapter does not claim at completely solving it. The purpose of the chapter is rather to illustrate the potential of the interpolatory framework to go toward a potentially satisfactory solution. So far, the proposed framework appears promising but would require additional work. It is nothing to say that this problem is indeed quite an active one, and many works from both control and mathematics communities are considering it (see *e.g.* Jacob et al. (2016); Dashkovskiy and Mironchenko (2018)).

5.1.2 A motivating example

Let us motivate the problem and proposed solution through the following time-delay dynamical model, which stability is usually quite complicated to prove for non experts.

Example 10 - TDS stability chart approximation

In this use-case, we consider an example provided in Sipahi et al. (2011), representing a dynamical system highlighting the stabilising effect of delays. The model is described by the following delay dependent dynamical representation (where $w_0 = 3$, $\tau \in [0, 10]$ and $k \in [0, 4]$):

$$\mathbf{H}(s, k, \tau) = \frac{ke^{-\tau s}}{s^2 + w_0^2 - ke^{-\tau s}}.$$

By applying the $\mathcal{L}_2$-**MFSA** procedure later described, over a 100×100 uniformly spaced grid of delay τ and feedback gain k, one obtains the stability index $S \in \mathbb{R}^{100\times100}$ which values are reported on Figures 5.1 and 5.2.

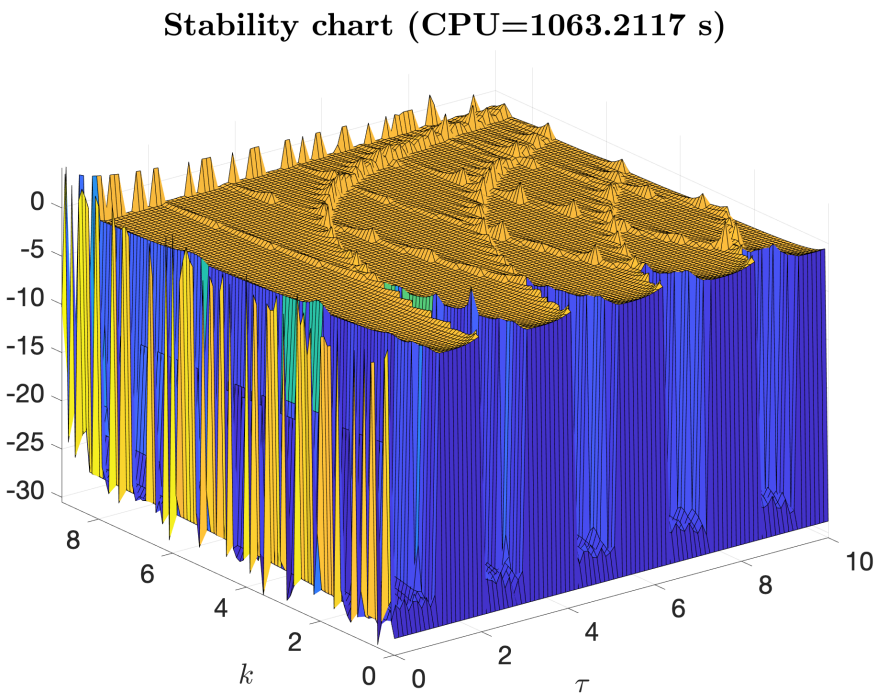


Figure 5.1: 3D representation of the stability index $\log(S)/\max(S)$ as a function of τ and k.

These latter have been obtained in approximatively 1063 seconds using a standard laptop (representing then 0.1s per point), which is quite reasonable. Besides, it is noteworthy that comparing Figure 5.1 with (Sipahi et al., 2011, Figure 10) shows a very accurate restitution of the stability regions. In practice, the above results has been obtained using the following script, involving the **`mor.stability`** interface.

```
% Sipahi et al., IEEE control system magazine, 2011
% Stabilizing effect of delays (p48 and Figure 10)
clear J
kSpace   = linspace(0,9,dx);      % dx=100
tauSpace = linspace(0,10,dx);     % dx=100
W        = logspace(-2.5,1,200); % frequency grid of evaluation
w0       = 3;
H        = @(s,p) 1/( s^2 + w0^2 - p(1)*exp(-p(2)*s)); % L2 meromorphic function
kk       = 0;
tic
for ii = 1:length(kSpace)
    for jj = 1:length(tauSpace)
        kk = kk + 1;
        fprintf('%0.3f %% \n',100*kk/(length(kSpace)*length(tauSpace)))
        p_ij     = [kSpace(ii) tauSpace(jj)];
        H_ij     = @(s) H(s,p_ij); % meromorphic function evaluation at p=[k \tau]
        J(ii,jj) = mor.stability(H_ij,W); % L2-MFSA procedure
    end
end
cpuTime = toc;
```

Listing 5.1: `demo_Chap5_startEvalBench` script: illustration of $\mathcal{L}_2$-**MFSA** procedure.

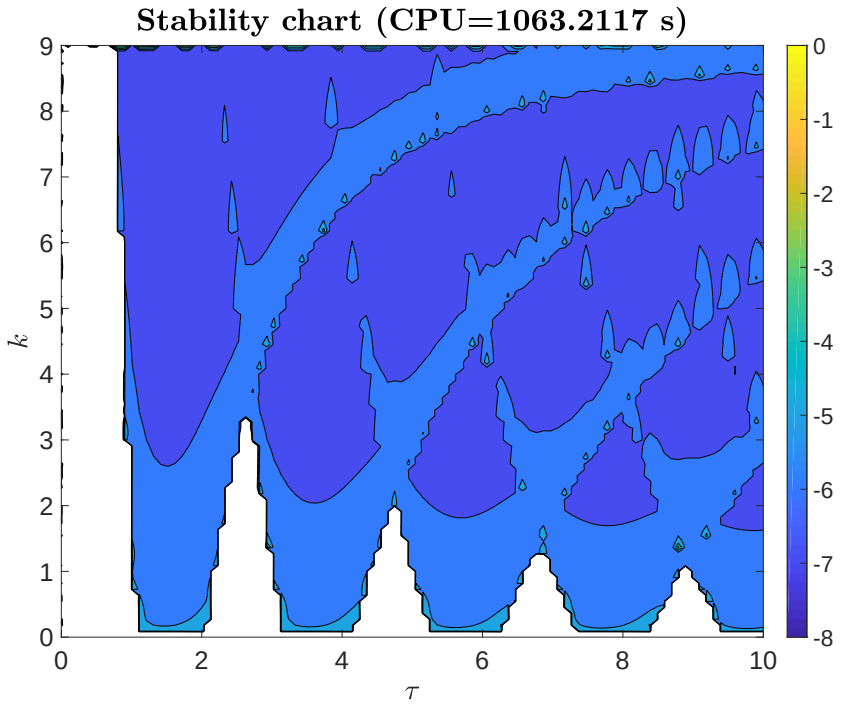


Figure 5.2: Projection of the stability index $\log(S)/\max(S)$ as a function of τ and k.

At this point, it can be pointed that in comparison with the **LMI**-based techniques or bifurcation tools developed in the time-delay literature, the velocity and relative simplicity of the proposed scheme seems appealing. Moreover, the accuracy of the algorithm seems good with respect to the exact solution. However, to be fair, it is important to keep in mind that **LMI**-based methods allow treating varying delay, while the one presented here, as well as the spectral methods (see *e.g.* Jarlebring et al., 2010) are restricted to fixed ones. Let us now describe the $\mathcal{L}_2$-**MFSA** procedure and provide some mathematical arguments.

5.2 Stability estimation of $\mathcal{L}_2$ meromorphic functions

Let us describe the proposed approach. We start with the derivation of the proposed numerical procedure (Algorithm 6). This algorithm is then justified in the following sub-sections, with some reminders, then, theoretical arguments to justify such a procedure, and finally, the steps for the construction of Algorithm 6.

5.2.1 The proposed $\mathcal{L}_2$-MFSA procedure

The proposed procedure is summed up as in Algorithm 6.

Algorithm 6 $\mathcal{L}_2$**-MFSA** - $\mathcal{L}_2$ Meromorphic Function Stability Approximation

Require: $\mathbf{H} \in \mathcal{L}_2$, $\{\omega_i\}_{i=1}^N \in \mathbb{R}_+$, $N \in \mathbb{N}$ and $\epsilon \in \mathbb{R}_+$ (typically twice machine precision)
1: Sample $\mathbf{H}$ and obtain $\{\omega_i, \mathbf{\Phi}_i\}_{i=1}^N$
2: Perform an exact Loewner interpolation (as described in Chapter 2) and obtain $\hat{\mathbf{H}}$ equipped with $\hat{\mathcal{S}} \in \mathbb{S}^0_{n,n_y,n_u}$ which ensures interpolatory conditions
3: Compute $\hat{\mathbf{H}}_s$ with realisation $\hat{\mathcal{S}}_+ \in \mathbb{S}^+_{n,n_y,n_u}$, the best stable approximation of $\hat{\mathbf{H}}$
4: Compute the stability index as $S = ||\hat{\mathbf{H}}_s - \hat{\mathbf{H}}||_{\mathcal{L}_2}$
5: **if** $S < \epsilon$ **then**
6: $\quad \mathbf{H}$ is stable
7: **else**
8: $\quad \mathbf{H}$ is unstable
9: **end if**

At this point, Algorithm 6 embeds a relative simple procedure, which will be show to be actually quite effective, fast and reliable in multiple cases. In few words, the idea consists in exactly matching the original input-output model by a rational model $\hat{\mathbf{H}}$, by guaranteeing interpolatory conditions. Then, to seek for the best stable approximation $\hat{\mathbf{H}}_s$ of the obtained model $\hat{\mathbf{H}}$. The $\mathcal{L}_2$ distance between the interpolated $\hat{\mathbf{H}}$ and stable $\hat{\mathbf{H}}_s$ models is then computed. If this latter is smaller than a given threshold, then we conclude that $\mathbf{H}$ is stable, and unstable otherwise. Obviously, such a simple procedure deserves some arguments to be fully satisfactory. In the following some arguments to justify such an approach are given. Still author stresses that all the necessary ones are not given, yet. In what follows we start by reminding standard results mentioned in Chapter 1.

5.2.2 Reminding remarks on stability by $\mathcal{L}_2$ inner product

Let us consider a **MIMO** linear dynamical system, denoted by $\mathbf{H}$ with n_u (*resp.* n_y) $\in \mathbb{N}^*$ inputs (*resp.* outputs), represented by its transfer function $\mathbf{H}(s) \in \mathbb{C}^{n_y \times n_u}$. Let $\mathcal{L}_2(\imath\mathbb{R})$ be the Hilbert space of holomorphic functions $\mathbf{F} : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$ which are analytic in the complex plane except on the imaginary axis and for which

$$\int_{-\infty}^{+\infty} \mathbf{tr}\left(\overline{\mathbf{F}(\imath\omega)}\mathbf{F}^T(\imath\omega)\right) d\omega < +\infty.$$

For given $\mathbf{G}, \mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})$, the associated inner-product reads:

$$\langle \mathbf{G}, \mathbf{H} \rangle_{\mathcal{L}_2} = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathbf{tr}\left(\overline{\mathbf{G}(\imath\omega)}\mathbf{H}^T(\imath\omega)\right) d\omega,$$

and the $\mathcal{L}_2(\imath\mathbb{R})$ norm can be explained:

$$||\mathbf{G}||_{\mathcal{L}_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{+\infty} \|\mathbf{G}(\imath\omega)\|_F^2 d\omega\right)^{1/2} = \langle \mathbf{G}, \mathbf{G} \rangle_{\mathcal{H}_2},$$

where $||\mathbf{G}||_F^2 = \langle \mathbf{G}, \mathbf{G} \rangle_F$ and $\langle \mathbf{G}, \mathbf{H} \rangle_F = \mathbf{tr}(\overline{\mathbf{G}}\mathbf{H}^T)$ are the Frobenius norm and inner-product, respectively. Since real dynamical systems are considered only, it is noteworthy that if $\mathbf{G}, \mathbf{H} \in \mathcal{L}_2$ are real, then $\langle \mathbf{G}, \mathbf{H} \rangle_{\mathcal{L}_2} = \langle \mathbf{H}, \mathbf{G} \rangle_{\mathcal{L}_2} \in \mathbb{R}_+$.

According to definitions given in Chapter 1, one notices that $\mathcal{H}_2(\mathbb{C}_-)$ is the left half-plane analog of $\mathcal{H}_2(\mathbb{C}_+)$, *e.g.* , $\mathbf{G} \in \mathcal{H}_2(\mathbb{C}_-)$ if and only if $\mathbf{G}(-s) \in \mathcal{H}_2(\mathbb{C}_+)$. Then $\mathcal{H}_2(\mathbb{C}_-)$ stands as the space of transfer function $\mathbf{H}(s)$ whose all poles lies in $\mathbb{C}_+$, *i.e.* the poles of $\mathbf{H}(s)$ are all unstable. The space $\mathcal{H}_2(\mathbb{C}_-)$ is called the space of anti-stable models. Now, let us remind the following important proposition, standing as the basis of the result derivation. The $\mathcal{H}_2(\mathbb{C}_-)$ and $\mathcal{H}_2(\mathbb{C}_+)$ spaces are closed subspaces of $\mathcal{L}_2(\imath\mathbb{R})$ and

$$\mathcal{L}_2(\imath\mathbb{R}) = \mathcal{H}_2(\mathbb{C}_-) \bigoplus \mathcal{H}_2(\mathbb{C}_+).$$

In addition, one can remind that by applying the Laplace transform, denoted as $\mathcal{L}(\cdot)$, over these two spaces, the following bijections are obtained:

$$\mathcal{L}(\cdot) : \mathcal{L}_2^n[0,\infty) \to \mathcal{H}_2(\mathbb{C}_+) \text{ and } \mathcal{L}(\cdot) : \mathcal{L}_2^n(-\infty,0] \to \mathcal{H}_2(\mathbb{C}_-), \tag{5.1}$$

which map the causal and anti-causal time-domain functions. Obviously, (5.1) shows that every element $\mathbf{H} \in \mathcal{H}_2(\mathbb{C}_+)$ (respectively $\mathbf{G} \in \mathcal{H}_2(\mathbb{C}_-)$) can be uniquely associated to an element $\mathbf{h} \in \mathcal{L}_2^n[0,\infty)$ (respectively $\mathbf{g} \in L_2^n(-\infty,0]$). In addition, the following functional analysis theorem shows that the Laplace transform preserves inner product and orthogonality.

Theorem 5.1: Plancherel

Let us consider $\mathbf{h}_1, \mathbf{h}_2 \in \mathcal{L}_2^n(-\infty,\infty)$, one has

$$\langle \mathbf{H}_1, \mathbf{H}_2 \rangle_{\mathcal{L}_2(\imath\mathbb{R})} = \langle \mathcal{L}(\mathbf{h}_1), \mathcal{L}(\mathbf{h}_2) \rangle_{\mathcal{L}_2(\imath\mathbb{R})} = \langle \mathbf{h}_1, \mathbf{h}_2 \rangle_{L_2}.$$

Moreover, since $\mathcal{H}_2(\mathbb{C}_-)$ is orthogonal to $\mathcal{H}_2(\mathbb{C}_+)$ with respect to the $\mathcal{L}_2(\imath\mathbb{R})$-inner product, if $\mathbf{H}_s \in \mathcal{H}_2(\mathbb{C}_+)$ and $\mathbf{H}_a = \mathcal{H}_2(\mathbb{C}_-)$, then

$$\langle \mathbf{H}_s, \mathbf{H}_a \rangle_{\mathcal{L}_2(\imath\mathbb{R})} = 0.$$

In other words, the above decomposition and Theorem 5.1 state that given a model $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})$, there is a stable model $\mathbf{H}_s \in \mathcal{H}_2(\mathbb{C}_+)$ and an anti-stable model $\mathbf{H}_a = \mathcal{H}_2(\mathbb{C}_-)$ such that $\mathbf{H} = \mathbf{H}_s + \mathbf{H}_a$ and $\langle \mathbf{H}_s, \mathbf{H}_a \rangle_{\mathcal{L}_2(\imath\mathbb{R})} = 0$. Then the following proposition holds.

Proposition 5.1: Input-output $L_\infty - L_2$ stability

A system $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})$ is input-output $L_\infty - L_2$ stable if and only if $\mathbf{H} \in \mathcal{H}_2(\mathbb{C}_+)$.

Theorem 5.1 and the $\mathcal{L}_2(\imath\mathbb{R})$ space decomposition are the main ingredient arguing for the proposed $\mathcal{L}_2$**-MFSA** procedure. These arguments are now provided in the following subsection.

5.2.3 Arguments for the $\mathcal{L}_2$-MFSA

Let us now provide some justifications for the use of interpolation methods in order to estimate the (in)stability of a given system $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})$. These arguments are based on the following results, stated in Pontes Duff et al. (2015b) and recalled in Propositions 5.2, 5.3 and 5.4, and finally in Theorem 5.2, providing the main argument for stability.

First of all, let us assume that the global minimiser $\hat{\mathbf{H}}$ of the $\mathcal{H}_2(\mathbb{C}_+)$ and $\mathcal{H}_2(\mathbb{C}_-)$ approximation problems exist[4]. Then, the following first result holds true.

Proposition 5.2: $\mathcal{L}_2$ orthogonality

If $\mathbf{H} \in \mathcal{H}_2(\mathbb{C}_+)$ and there exists a global minimizer $\hat{\mathbf{H}} \in \mathcal{L}_2(\imath\mathbb{R})$ of the $\mathcal{L}_2$ approximation problem, then $\hat{\mathbf{H}} \in \mathcal{H}_2(\mathbb{C}_+)$. Similarly, if $\mathbf{H} \in \mathcal{H}_2(\mathbb{C}_-)$ and there exists a global minimizer $\hat{\mathbf{H}} \in \mathcal{L}_2(\imath\mathbb{R})$ of the $\mathcal{L}_2$ approximation problem, then $\hat{\mathbf{H}} \in \mathcal{H}_2(\mathbb{C}_-)$.

[4]The $\mathcal{H}_2(\mathbb{C}_+)$ approximation problem is simply Problem 2.1 while the $\mathcal{H}_2(\mathbb{C}_-)$ stand as the same one but for $\mathbf{H}(-s)$.

Proof 5.1 (of Proposition 5.2) *is given in Pontes Duff et al. (2015b).*

In other words, the following statements holds true,

- if a system $\mathbf{H}$ is stable, then the global minimiser $\hat{\mathbf{H}}$ of the $\mathcal{L}_2$ approximation problem is stable too.
- if a system $\mathbf{H}$ is anti-stable, *i.e.* all its eigenvalues are unstable, then the global minimiser $\hat{\mathbf{H}}$ of the $\mathcal{L}_2$ problem is anti-stable as well.

This result directly comes from the orthogonality property of $\mathcal{H}_2(\mathbb{C}_-)$ and $\mathcal{H}_2(\mathbb{C}_+)$ spaces.

Let us now denote by $\hat{\mathbf{H}}_k$, the sequence of models of order $k \in \mathbb{N}^*$ and consider the case where the initial model $\mathbf{H}$ is stable, then the following proposition holds.

Proposition 5.3: Unstable approximate sequence of stable model

Given a stable model $\mathbf{H} \in \mathcal{H}_2(\mathbb{C}_+)$, there exists a sequence of k-th order unstable models $\hat{\mathbf{H}}_k \in \mathcal{L}_2(\imath\mathbb{R})\backslash\mathcal{H}_2(\mathbb{C}_+)$, $k \in \mathbb{N}^*$, such that, when $k \to \infty$, $||\mathbf{H} - \hat{\mathbf{H}}_k||_{\mathcal{L}_2} \to 0$.

Proof 5.2 (of Proposition 5.3) *is given in Pontes Duff et al. (2015b).*

In other words, the set $\mathcal{H}_2(\mathbb{C}_+)$ is not an open set of $\mathcal{L}_2(\imath\mathbb{R})$. As a consequence, it is always possible to approximate a stable model $\mathbf{H}$ by an unstable one of order k while decreasing the $\mathcal{L}_2$ mismatch error $||\mathbf{H} - \hat{\mathbf{H}}_k||_{\mathcal{L}_2}$. Similarly, let us now consider the case where the initial model $\mathbf{H}$ both has stable and unstable modes.

Proposition 5.4: Unstable approximate of unstable model

Given an unstable model $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})\backslash\mathcal{H}_2(\mathbb{C}_+)$, there exists $\varepsilon > 0$ such that the ball $B_\varepsilon(\mathbf{H})$ defined as

$$B_\varepsilon(\mathbf{H}) = \left\{\hat{\mathbf{H}}_k \in \mathcal{L}_2(\imath\mathbb{R}) \;\middle|\; ||\mathbf{H} - \hat{\mathbf{H}}_k||_{\mathcal{L}_2} < \varepsilon\right\},$$

satisfies $B_\varepsilon(\mathbf{H}) \subset \mathcal{L}_2(\imath\mathbb{R})\backslash\mathcal{H}_2(\mathbb{C}_+)$.

Proof 5.3 (of Proposition 5.4) *is given in Pontes Duff et al. (2015b).*

In other words, the set of unstable systems $\mathcal{L}_2(\imath\mathbb{R})\backslash\mathcal{H}_2(\mathbb{C}_+)$ is an open set of $\mathcal{L}_2(\imath\mathbb{R})$. Moreover, by fixing an arbitrarily small ε, it is always possible to find a $\hat{\mathbf{H}}_k$ that is unstable too. Based on the above three propositions, let us now formulate the following stability argument, which will be invoked in order to derive the proposed numerical procedure.

Theorem 5.2: Main stability argument

Given an unstable system $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})\backslash\mathcal{H}_2(\mathbb{C}^+)$, there exists $r \in \mathbb{N}^*$ for which the minimizer $\hat{\mathbf{H}}_k$ of order $k \in \mathbb{N}^*$, $k \geq r$, obtained from the $\mathcal{L}_2$-approximation problem is also unstable.

Since Proposition 5.4 states that if a system $\hat{\mathbf{H}}_k$ is sufficiently close to an unstable system in the $\mathcal{L}_2$-norm, it is also unstable. Since, the subspace of rational finite **LTI** systems function is dense in $\mathcal{L}_2$, for a given **LTI** unstable system $\mathbf{H} \in \mathcal{L}_2(\imath\mathbb{R})\backslash\mathcal{H}_2(\mathbb{C}^+)$, a sequence $\hat{\mathbf{H}}_k$ of systems of order $k \in \mathbb{N}$ which satisfies the $\mathcal{L}_2$ approximation problem, will converge to $\mathbf{H}$. Thus, due to Theorem 5.2, there exists an order $r \in \mathbb{N}^*$ such that if $k \geq r$, $\hat{\mathbf{H}}_k$ will be unstable as well.

In other words there exists an approximation order $k \geq r$ such that if the original system $\mathbf{H}$ is unstable, the approximated one $\hat{\mathbf{H}}$ is unstable too. Moreover, if one has found the global $\mathcal{L}_2$ minimiser of the approximation problem of order r, it will be stable if the original model is stable, due to Proposition 5.2, and it will be unstable if the original model is unstable, due to Theorem 5.2.

5.2.4 Toward the derivation of the $\mathcal{L}_2$-MFSA

In Pontes Duff et al. (2015b), these arguments were used to derive a procedure based on the **TF-IRKA** algorithm, combined with a greedy search of the approximation order r. This last procedure did provide quite good results but the search procedure for an adequate order r was complex and led to a time consuming procedure. Moreover, the **TF-IRKA** is an $\mathcal{H}_2(\mathbb{C}_+)$-oriented procedure and its validity in the $\mathcal{L}_2(\imath\mathbb{R})$ function space is limited to models where the stable and anti-stable part is known (see Magruder et al., 2010).

Consequently, in this chapter and in Poussot-Vassal and Vuillemin (2019), the $\mathcal{H}_2$-optimal interpolatory conditions are released and one now considers the interpolatory conditions embedded in the Loewner framework instead. One major benefit of such a trade stands in the selection of the approximating order r, which may be automatically done, thanks to Lemma 2.1. By coming back to the $\mathcal{L}_2$-**MFSA** procedure defined in Algorithm 6, step 2 provides a simple solution, where the dimension r is automatically determined by the rank of the Loewner matrices. Then, $\hat{\mathbf{H}}$ tangentially interpolates the data (without bi-tangential interpolation). Then, following Propositions 5.3 and 5.4, it appears quite naturally that, approximating the interpolated model $\hat{\mathbf{H}}$ with $\hat{\mathbf{H}}_s$ using the methodology given in Theorem 1.3 and proposed by Kohler (2014), followed by the $\mathcal{L}_2$ distance evaluation is appealing. Indeed, as

- From a stable model $\mathbf{H}$, it is always possible to find an unstable one which minimises the $\mathcal{L}_2$ mismatch problem, if, based on an unstable approximation a same complexity (order) stable model may be obtained, without affecting the $\mathcal{L}_2$ norm, then the interpolated model is stable, as the original one.
- From an unstable model $\mathbf{H}$, its global minimiser in the $\mathcal{L}_2$ sense should be unstable as well, then, applying a stable approximation will lead to a large $\mathcal{L}_2$ mismatch. Then, one may confirm that the original model is unstable, as its rational approximant.

The conjecture we claim is in twofolds:

1. One is always able to find a rational model $\hat{\mathbf{H}} \in \mathcal{RL}_2$ that well reproduces $\mathbf{H} \in \mathcal{L}_2$, whatever the complexity of $\mathbf{H}$ is, if we can arbitrarily increase r, the dimension of $\hat{\mathbf{H}}$.
   This can be achieved by increasing the Loewner matrix up to a numerical rank loss.
2. If, based on an unstable realisation of $\hat{\mathbf{H}} \in \mathcal{RL}_2$, the optimal stable approximant $\hat{\mathbf{H}}_s \in \mathcal{H}_2$ is close enough to $\hat{\mathbf{H}} \in \mathcal{RL}_2$, in the sense of the $\mathcal{L}_2$-norm, then $\hat{\mathbf{H}}$ is stable and, following previous statement (1.), $\mathbf{H}$ is stable too.
   This step can be achieved by a rational stable approximation followed by a norm computation which threshold is fixed to machine precision.

Remark 5.1 (About rational approximation of a meromorphic function) *In all the above considerations, the starting point aims at approximating any meromorphic function* $\mathbf{H}$ *by* $\hat{\mathbf{H}}$ *(equipped with* $\hat{\mathcal{S}}$*), a rational meromorphic model satisfying interpolatory conditions. One (huge?) underlying question is, how accurately a rational form can reproduce an irrational one? This is not clearly stated so far. With reference to Algorithm 6, this point may be by-passed by selecting a wide enough frequency range* ω_i *and enough points* N *(note that* N *may be increased up to the moment where the rank revealing factorisation reaches machine precisions).*

5.3 Numerical illustrations

Now the main idea and some theoretical arguments for the proposed $\mathcal{L}_2$-**MFSA** approach have been derived, let us illustrate it on some different problems. Indeed, as theoretical results are clearly missing to be fully mathematically complete, with this section, author aims at motivating and promoting the feature of this approach, which he believes is promising. First, let us start with a finite dimensional dynamical model.

Example 11 - Ordinary differential equation model

On this example, we consider the stable Clamped beam model provided in Leibfritz (2003), equipped with a state-space vector of order $n = 348$. Then, we consider the same system with an additional unstable pole in $+1$. To illustrate the process, the following code is executed and results obtained:

```
[A,B1,B,C1,C,D11,D12,D21,nx,nw,nu,nz,ny] = COMPleib('CBM');
W = logspace(-3,2,500);
H = ss(A,B,C,0);
mor.stability(H,W)
isstable(H)
H = series(H,tf(1,[1 -1])); % add an unstable pole in 1 rad/s
mor.stability(H,W)
isstable(H)
```

Listing 5.2: `demo_Chap5_ode_dae` script: illustration of $\mathcal{L}_2$-**MFSA** procedure.

The above code leads to the following results:

```
ans =
   1.6017e-15 % stability index by mor.stability, close to machine precision => stable
ans =
  logical
   1  % => stable according to Matlab
ans =
   61.0948  % stability index by mor.stability, greater than machine precision =>
       unstable
ans =
  logical
   0  % => unstable according to Matlab
```

Listing 5.3: Results of the above script.

By analysing the above numerical values, the first one, representing the stability index provides an index close to machine precision ($\approx 10^{-15} \approx 0$) meaning that the model is stable (MATLAB **`issable`** function confirms this conclusion). In the second result set, the stability index is far from machine precision (≈ 61) therefore, following the above arguments and conjecture, the model is assumed to be unstable (MATLAB **`issable`** function confirms it). One can repeat the experiment with different unstable pole values, *i.e.* **`0`**, **`1e-6`** or **`1e2`** and the process keep working. Still, to be completely fair, if the unstable eigenvalue is "too far" from the bounds of **`W`**, the frequency support provided as input and which will dictate the ability of the Loewner interpolation step to reproduce the behaviour in this frequency range (especially if the function rolls-off), the result may fail. Indeed, executing the following code:

```
[A,B1,B,C1,C,D11,D12,D21,nx,nw,nu,nz,ny] = COMPleib('CBM');
W = logspace(-3,2,500); % try also W = logspace(-3,3,500);
H = ss(A,B,C,0);
mor.stability(H,W)
isstable(H)
H = series(H,tf(1,[1/1e6 -2*.1/1e3 1])); % add an unstable pole in 1e3 rad/s
mor.stability(H,W)
isstable(H)
```

Listing 5.4: `demo_Chap5_ode_dae` script: illustration of $\mathcal{L}_2$-**MFSA** procedure.

Here, we clearly exhibit a limitation...

```
ans =
   1.6017e-15 % stability index by mor.stability, close to machine precision => stable
ans =
  logical
   1 % => stable according to Matlab
ans =
   1.2172e-15 % stability index by mor.stability, close to machine precision => stable
       !!!
ans =
  logical
   0 % => unstable according to Matlab
```

Listing 5.5: Results.

Here, the third number, **`1.2172e-15`**, indicates that according to the $\mathcal{L}_2$-**MFSA** the model is stable, which is obviously not true at all. By replacing the frequency grid **`W`** by **`W = logspace(-3,3,500)`** the problem is almost solved. Indeed it leads to a stability index of **`2.0394e-05`**, now closed to zero even if still far from machine precision.

Now, as it is of great interest for control practitioners (*e.g.* control engineers implementing control laws on target including networks and computational delay), let us now apply the proposed procedure, to a **TDS** dynamical model embedding multiple delays.

Example 12 - High speed network TDS model

Let us consider a congestion system of a high speed network, originally derived from Izmailov (1996) and analyzed in Niculescu (2002) and Sipahi et al. (2011). Its behaviour is driven by the following time-delayed dynamical equations:

$$\left\{ \begin{array}{rcl} \dot{\mathbf{x}}(t) & = & A_0\mathbf{x}(t) + A_1\mathbf{x}(t-\tau) + A_2\mathbf{x}(t-\tau-r) + \mathbf{b}u(t) \\ y(t) & = & \mathbf{c}^T\mathbf{x}(t) \end{array} \right. ,$$

where

$$A_0 = \left[\begin{array}{cc} 0 & 0 \\ 1 & 0 \end{array} \right] , A_1 = \left[\begin{array}{cc} 0 & -a \\ 0 & 0 \end{array} \right] , A_2 \left[\begin{array}{cc} 0 & -b \\ 0 & 0 \end{array} \right] , \mathbf{b} = \left[\begin{array}{c} 1 \\ 0 \end{array} \right] \text{ and } \mathbf{c} = \left[\begin{array}{c} 0 \\ 1 \end{array} \right] ,$$

and where $a = 2$, $b = -1.75$ and $\tau, r \in \mathbb{R}_+$ are the fixed delays. One is interested in deriving the so-called stability chart as a function of the $\{\tau, r\}$ couple delay values. More specifically, we are interested in the stability conditions for $\tau \in [0, 1.4]$s and $r \in [0, 1.8]$s for which the system is stable or not. This specific problem has been theoretically solved by Niculescu (2002), using complete analysis of its characteristic transcendaental equation leading to the exact stability chart as a function of $\{\tau, r\}$.
When comparing the solution obtained in (Niculescu, 2002, Figure 1) to the one based on the $\mathcal{L}_2$-**MFSA** procedure reported on Figure 5.3, the stable area (white area on Figure 5.3) almost perfectly coincide. More specifically, Figure 5.3 reports the stability index (which amplitude is in log-scale) as a function of τ and r, obtained by the $\mathcal{L}_2$-**MFSA** over a finite grid set of frozen $\{\tau, r\}$ values (here 50×50 grid points). The proposed algorithm well catches the (in)stability property while being quite fast (result obtained in 930s, only) and without any a-priori knowledge. Note also that this process may be connected to a boundary search process, to accelerate it even more, as done in Pontes Duff et al. (2015b).

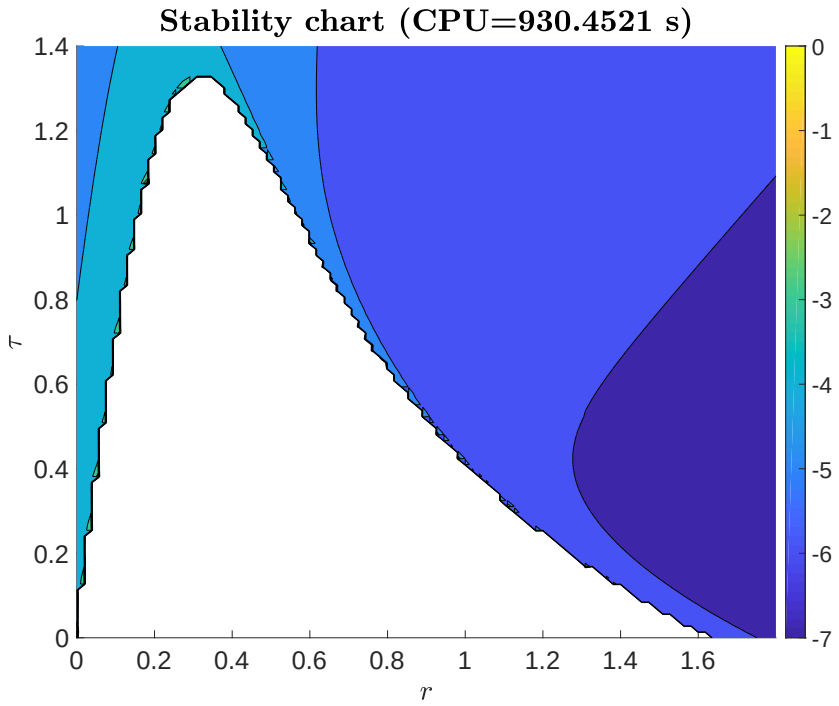


Figure 5.3: Stability index chart as a function of the delay values $\{\tau, r\}$ (values are in logarithmic scale).

User may consider that the above figure is simply obtained with following code, calling the **`mor.stability`** routine of the **MOR** Toolbox.

```
% Congestion error (p58 and Figure 20, values from [74])
tau1Space = linspace(0,1.8,dx); % dx=100
tau2Space = linspace(0,1.4,dx); % dx=100
W         = logspace(-2,1,200);
a         = 2;
b         = -1.75;
A0        = [0 0; 1 0];
A1        = [0 -a; 0 0];
A2        = [0 -b; 0 0];
B         = [1; 0];
C         = [0 1];
Id        = eye(length(A0));
H         = @(s,p) C*(Id/( s*Id - A0 - A1*exp(-p(2)*s) - A2*exp(-(p(2)+p(1))*s)))*B;
kk        = 0;
tic
for ii = 1:length(tau1Space)
    for jj = 1:length(tau2Space)
        kk = kk + 1;
        fprintf('%0.3f %% \n',100*kk/(length(tau1Space)*length(tau2Space)))
        p_ij      = [tau1Space(ii) tau2Space(jj)];
        H_ij      = @(s) H(s,p_ij);
        J(ii,jj) = mor.stability(H_ij,W);
    end
end
cpuTime = toc;
```

Listing 5.6: **`demo_Chap5_startEvalBench`** script: illustration of $\mathcal{L}_2$-**MFSA** procedure.

As for the previous case, the function call is also made simple and few parameters are required by the user.

Moving to an other class of infinite dimensional systems, let us now consider linear **PDE** which may lead to irrational transfer functions (as illustrated in Chapter 2). The following one falls to be also a **TDS** model but with an additional complexity related to the descriptor form with a strong rank loss.

Example 13 - Linear PDE

In Pilbauer et al. (2018), authors did present a study involving a drilling system. This later is controlled by PI-structured controller and authors investigate the impact of an input shaping on the performances. In the context of this chapter, it is interesting to notice that such system (parametrised by the controller gain k_p and integral gain k_i), may be represented by linear **PDE** which can be turned into a delayed-descriptor form as follows:

$$\mathbf{H}(s, k_p, k_i) = C\left(sE - A_0(k_p, k_i) - A_1 e^{-\tau_1 s} - A_2 e^{-\tau_2 s}\right)^{-1} B(k_p),$$

where matrices and delays are defined in the code given as follows:

```
%% #6
% D. Pilbauer et al., ECC 2018
% Oil drilling system
c0      = 2;
c1      = c0;
q0      = -.9;
q1      = q0;
a0      = .17;
a1      = 1.8;
b0      = a0;
b1      = a1;
lambda  = 1;
mu_     = lambda;
tau1    = 1/lambda;
tau2    = 1/mu_;
tau3    = 1/lambda + 1/mu_;

% Rational approximation
W       = logspace(-2,2,100)*2*pi;
nW      = numel(W);
kpSpace = linspace(-5,5,71);
kiSpace = linspace(-10,10,70);
kk      = 0;
tic
for ii = 1:length(kpSpace)
    kp = kpSpace(ii);
    for jj = 1:length(kiSpace)
        ki = kiSpace(jj);
        kk = kk + 1;
        fprintf('%d%%\n',100*kk/(length(kpSpace)*length(kiSpace)))
        nx      = 7;
        E       = diag([1 0 0 0 0 1 1]);
        A0      = zeros(nx,nx);
        A0(1,1) = -a0-kp;
        A0(1,3) = b0;
        A0(1,7) = ki;
        A0(2,1) = c0;
        A0(2,2) = -1;
        A0(2,3) = q0;
        A0(3,3) = -1;
        A0(4,4) = -1;
        A0(5,4) = q1;
        A0(5,5) = -1;
        A0(5,6) = c1;
        A0(6,4) = b1;
        A0(6,6) = -a1;
        A0(7,1) = -1;
        A1      = zeros(nx,nx);
        A1(3,5) = 1;
        A2      = zeros(nx,nx);
```

```
        A2(4,2) = 1;
        B       = [kp 0 0 0 0 0 1]';
        C       = ones(1,nx);
        H       = @(s) C*((s*E-A0-A1*exp(-tau1*s)-A2*exp(-tau2*s))\B);
        try
            J(ii,jj) = mor.stability(H,W);
        catch
            J(ii,jj) = inf;
        end
    end
end
cpuTime = toc;
```

Listing 5.7: `demo_Chap5_startEvalBench` script: illustration of $\mathcal{L}_2$-**MFSA** procedure.

On the above model one may notice that the E matrix is strongly rank deflective. Indeed, for delays $\{\tau_1, \tau_2\} = \{0, 0\}$, the (E, A) pencil shows four infinite eigenvalues, making the problem even more (numerically) complex. Still, Figure 5.4 shows the stability index values as function of the PI gains. When plugging the gains couple in the white area in a time-domain simulation leads to an input-output stable behaviour.

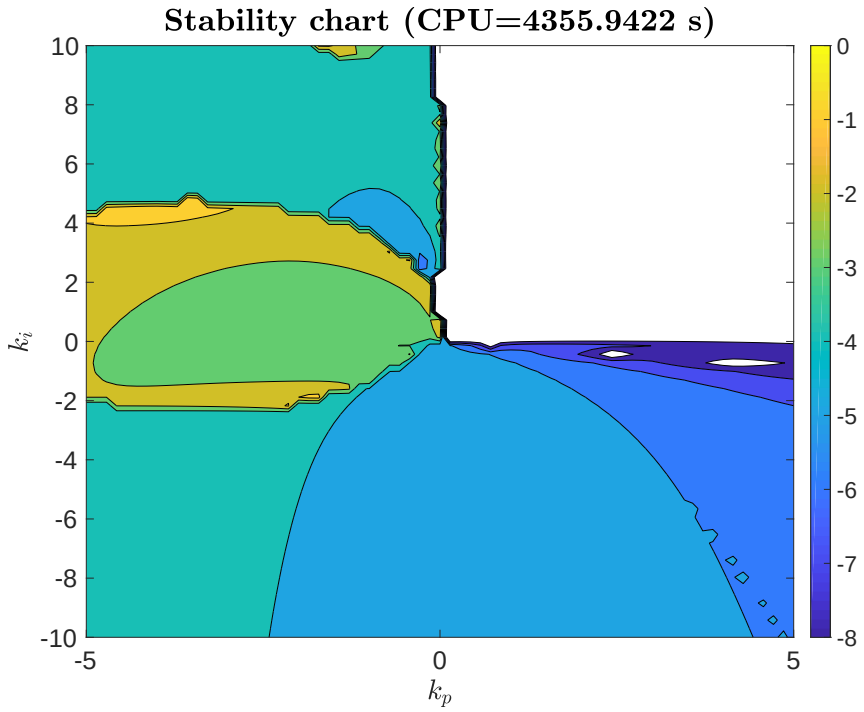


Figure 5.4: Stability index chart as a function of the PI gains $\{k_p, k_i\}$ (values are in logarithmic scale).

Still, attentive reader may notice some white areas for slightly negative k_i values. Here again, these latter are probably more related to numerical issues rather than realistic stabilising gain values, showing that the solution still require some additional developments.

5.4 Conclusions

In this chapter, a new paradigm for stability estimation, accompanied by the $\mathcal{L}_2$**-MFSA** procedure, a numerically simple but yet effective algorithm, has been presented. At this point, author must stress out that the chapter presents an embryonic idea. Indeed, despite the preliminary theoretical arguments and procedure, reader should be aware that a complete proof is still missing. However, author is convinced that such framework may represent a notable advance in the stability analysis of any $\mathcal{L}_2$ functions. So far, to the author's best efforts, such approach did fail in a very few cases, and seems to be quite robust with respect to the model nature. Clearly, if additional theoretical arguments can be stated, such a procedure may be a nice way to attack the complex function stability question, at least as an initial guess. To this aim, the works of Prokhorov (1994); Gonchard (2003) seem to be an interesting starting point.

As a direct continuation and following the **PDE** example given above, such a procedure may also be embedded in a structured controller design procedure *e.g.* by linking with norm estimation techniques, boundary search algorithms *etc.* (see *e.g.* Pontes Duff et al., 2015c, 2016b).

To the author viewpoint, one of the main interests of this approach stand in *(i)* its versatility and applicability to many models structures, including multiple input-output gain dependent, delays, irrational form *etc.* and *(ii)* in its scalability to large-scale dynamical models equipped with a large state-space vector. Without entering into details, such a scalability has already been observed when applying the methods to industrial aircraft dynamical models (see *e.g.* Ossmann and Poussot-Vassal, 2019).

Part II

Epilogue

Chapter 6

Conclusions and discussions

> Voici venu le temps d'affirmer, contre les économistes, que l'inutile crée de l'utilité, que la gratuité crée de la richesse, que l'intérêt ne peut exister sans le désintéressement.
>
> Bernard Maris

Contents

In this manuscript, author aimed at describing the large-scale linear dynamical model approximation problem and (some of) its many applications and implications. First, its "classical" functionality as a model simplifier was done, then, some un-standard side applications were illustrated. Here, we summarise the two main theoretical and methodological advances within the model reduction and approximation field, in Section 6.1. Then, Section 6.2 points the main successful industrial applications within the aeronautical field. Section 6.3 recalls the more exotic and strange side effects of model approximation, specifically for dynamical systems stability analysis. Then, Section 6.4 presents the **MOR** Toolbox, embedding some of the numerical tools and methods presented in this manuscript within a single easy-to-use engineer-oriented interface. Section 6.5 finally closes the chapter and opens the door to Chapter 7 for further developments and researches, from either the theoretical, methodological, numerical and applicative angles.

6.1 Highlights of the methodological contributions

The main methodological contributions and innovations within the model approximation field concern *(i)* the frequency-limited $\mathcal{H}_2$ approximation (Chapter 3), and *(ii)* the $\mathcal{H}_2$ approximation using structured delayed input-output reduced order models (Chapter 4). Both results, proof and detailed reasoning are embedded in the Ph.D. manuscript of Vuillemin (2014) (Chapters 7, 8 and 9) and Pontes (2017) (Chapter 5), respectively. Interestingly and almost casually, as in the seminal works of Gallivan et al. (2004a); Gugercin et al. (2008); Van Dooren et al. (2008); Van Dooren et al. (2010), both solutions share the interpolatory framework as a common basis. With reference to *(i)*, the first set of interpolatory condition boils down as follows.

$\mathcal{H}_{2,\Omega}$ frequency-limited model approximation interpolarory conditions

In the case of $\mathcal{H}_{2,\Omega}$ model approximation (or approximation over a frequency limited range), and following the notations adopted along this manuscript, the interpolatory conditions described in Chapter 3 can be written as ($l = 1, \ldots, r$):

$$\left\{\begin{array}{rcl} \mathbf{T}_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{T}}_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \\ \hat{\mathbf{c}}_l^H\mathbf{T}_\omega(-\hat{\lambda}_l) & = & \hat{\mathbf{c}}_l^H\hat{\mathbf{T}}_\omega(-\hat{\lambda}_l) \\ \hat{\mathbf{c}}_l^H\mathbf{T}'_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{c}}_l^H\hat{\mathbf{T}}'_\omega(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \end{array}\right.$$

where

$$\begin{aligned} \mathbf{T}_\omega(s) &= \sum_{j=1}^{n} \mathbf{c}_j\mathbf{b}_j^H \left(\frac{a_\omega(\lambda_j) - a_\omega(s)}{s - \lambda_j},\right) - Da_\omega(s) \\ \hat{\mathbf{T}}_\omega(s) &= \sum_{k=1}^{r} \hat{\mathbf{c}}_k\hat{\mathbf{b}}_k^H \left(\frac{a_\omega(\hat{\lambda}_k) - a_\omega(s)}{s - \hat{\lambda}_k}\right) - \hat{D}a_\omega(s), \end{aligned}$$

involving $\mathbf{T}_\omega$ and $\hat{\mathbf{T}}_\omega$ instead of $\mathbf{H}$ and $\hat{\mathbf{H}}$ in the $\mathcal{H}_2$ case.

As it, these interpolatory conditions are difficult to practically exploit. Indeed, no Krylov-like subspace have been clearly identified yet. Consequently, an **IRKA** like procedure as in Gugercin et al. (2008) is not straightforward to develop. So far, this $\mathcal{H}_{2,\Omega}$ problem has been attacked using the **DARPO** procedure Vuillemin et al. (2014a), a descent algorithm, without taking advantage of the interpolation framework. Therefore, improvements may consider developing a manner to exploit these conditions directly by *e.g.* mixing the Loewner framework (allowing approximating any irrational transfer function as $\mathbf{T}_\omega(s)$ by a rational one) with an iterative procedure (such as the **IRKA** one). As a hint, the interpolation step of $\mathbf{T}_\omega$ may consider an approximation by a function embedding a rational structure and possibly a richer structure, as proposed by Schulze et al. (2018), more adapted and tailored to the frequency limited specific one.

Similarly, following the delay structured approximation *(ii)*, the second interpolatory conditions set can be formulated as follows.

$\mathcal{H}_2$ structured input-output model approximation interpolatory conditions

In the case of $\mathcal{H}_2$ model approximation by an input-output delay structured reduced order model, and following the notations adopted along this manuscript, the interpolatory conditions described in Chapter 4 can be written as ($l = 1, \ldots, r$):

$$\left\{\begin{array}{rcl} \mathbf{T}_d(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{H}}(-\hat{\lambda}_l)\hat{\mathbf{b}}_l \\ \hat{\mathbf{c}}_l^T\mathbf{T}_d(-\hat{\lambda}_l) & = & \hat{\mathbf{c}}_l^H\hat{\mathbf{H}}(-\hat{\lambda}_l) \\ \hat{\mathbf{c}}_l^H\mathbf{T}'_d(-\hat{\lambda}_l)\hat{\mathbf{b}}_l & = & \hat{\mathbf{c}}_l^H\hat{\mathbf{H}}'(-\hat{\lambda}_k)\mathbf{b}_l \end{array}\right.$$

where

$$\mathbf{T}_d(s) = \sum_{j=1}^{n} \hat{\mathbf{\Delta}}_o(-\mu_j)\frac{\mathbf{c}_j^H\mathbf{b}_j}{s - \mu_j}\hat{\mathbf{\Delta}}_i(-\mu_j)$$

involving $\mathbf{T}_d$ instead of $\mathbf{H}$ in the unstructured $\mathcal{H}_2$ case. The additional constrain on the delay values are given in (4.10).

Here again, as in the $\mathcal{H}_2$ and $\mathcal{H}_{2,\Omega}$ cases, interpolarory conditions hold. Nevertheless in contrast with the $\mathcal{H}_{2,\Omega}$ context recalled above, an iterative procedure similar to the **IRKA** one, denoted **IO-dITIA** has been proposed. This latter allows tacking benefit of the Krylov subspaces and thus makes the approach scalable. Still, from an algorithmic point of view, the **IO-dITIA** suffers from two main limitations: first, the pole residue decomposition of the original model is required to construct $\mathbf{T}_d$ (which is rarely accessible in very large-scale context), and second, the delay optimisation is done for a fixed poles-residue set. Therefore, future developments may consider the algorithmic issues rather than theoretical one. Although, as pointed in Chapter 4, a link between the work of Halevi (1996) and Pontes Duff et al. (2018) would be interesting to elegantly close the theoretical aspects of this problem.

6.2 Approximation as pivot in civilian aircraft engineering

Along the manuscript, attention has been given to illustrate the applicability of the proposed methods both in the academic and industrial worlds. Indeed, multiple use-cases were used to either illustrate state of the art or new model approximation methods. Most of these use-cases were provided in the context of collaborative projects, by industrial partners within the aeronautical domain. More specifically these use-cases have been carried out in strong collaborations with **Dassault-Aviation** and **Airbus** engineers[1]. It is important to stress out that in most collaborative projects, model approximation is rarely the single objective. Still, it is undoubtedly a very important step which serves other purposes *e.g.* the control design, the simulation or the analysis and norm estimations. In all projects the author have been involved in, the model approximation step did stand as an unlocking tool, allowing improving control design and performance analysis, alleviating numerical limitations and providing accuracy and efficiency in an industrial iterative process. As a major highlight, the following recalls three successful aeronautical applications illustrating the key role of model approximation for control and analysis, in both industrial and research environments.

Model-based approximation for Dassault-Aviation Falcon 7X vibration control design
During the collaboration with **Dassault-Aviation** control engineers, the vibration attenuation around $[7, 10]$Hz for all flight and mass configurations and using movable surfaces, was targeted. To this aim, 48 dynamical models, representing the aircraft at different configurations were used. The selected approach to design such a control law was the $\mathcal{H}_\infty$-norm attenuation framework offered by the recent development of Apkarian and Noll (2006) and the MATLAB powered **`hinfstruct`** function.

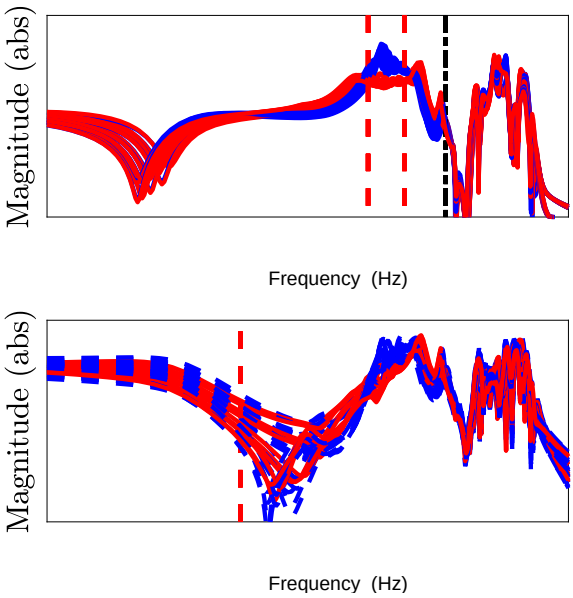


Figure 6.1: Vibration attenuation of some of the **Dassault-Aviation** models. Top: comfort criteria. Bottom: handling quality criteria. Right: controller characteristics and performances.

Still, as each single state-space model embeds a dimension close to 700 states, the model approximation over a finite frequency range using the **MOR** Toolbox (both **`mor.lti`** and **`mor.norm`** interfaces), played a crucial role in the success of the control design. Indeed, it allows strongly reducing the complexity, rending the design step almost easy. Figure 6.1 illustrates some of the obtained performances at a frozen flight altitude (here almost 50% of vibration attenuation were observed).

[1] Author also points out the huge efforts and confidence shown by these industrial partners. Indeed these models have a strategical importance in the competitive industrial world. Using these models also clearly help me so greatly improving the methods numerical reliability, and enhancing the user experience.

Following the obtained promising results, a similar control strategy has been propagated by **Dassault-Aviation** control expert on a model family describing the same aircraft, but stuck on ground (*i.e.* without flight mechanical equations). Then, the obtained controller were validated during ground vibration tests.

Data-driven approximation for Dassault-Aviation Falcon 7X aeroelastic model generation
On September 14th, 2015, ground test for vibration control demonstration has been applied on a real **Dassault-Aviation** Business Jet Falcon 7X test aircraft, excited by shakers at the aircraft rear (see Figure 6.2). During this work, **Dassault-Aviation** did illustrate the effective vibration attenuation achieved.

Figure 6.2: Falcon 7X s/n 001 during ground vibration test (with some **Dassault-Aviation** colleagues).

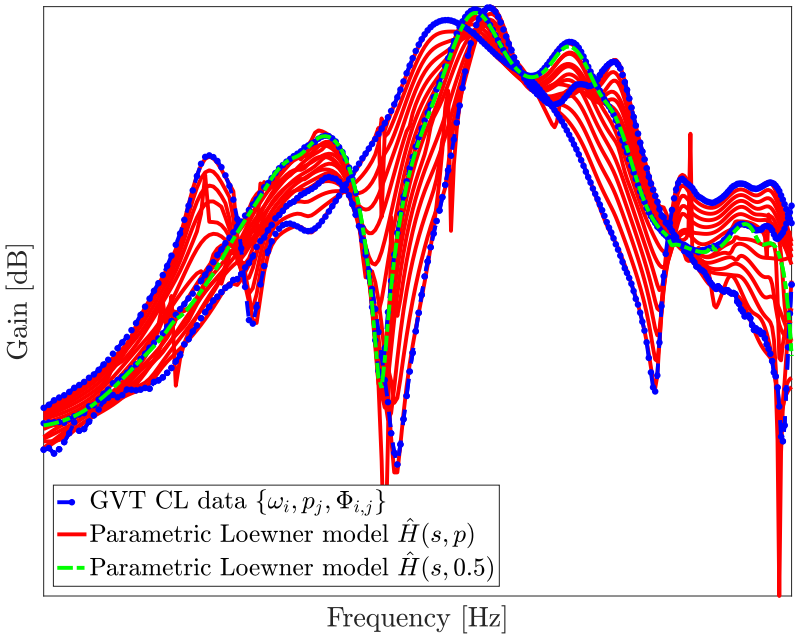


Figure 6.3: Parametric model construction from data. Collected data (blue dotted), normalized parametric model evaluated at frozen mass (red) and mean behaviour (dashed green).

During this phase, **Dassault-Aviation** aeroelastic, control and test engineers measured the acceleration, velocities and displacements at hundred points of the aircraft in response to shakers excitation. All the collected data are very precious in order to double check the control performances, but also to validate the original aircraft (finite-element) models.

During this test phase, in which I had the chance to participate, the Loewner framework, through the `mor.lti` interface, has been used to both recover the model and control characteristics (*i.e.* open- and closed-loop transfers). This method showed impressive results in its ability to recover the aircraft modal content as well as all the hundred transfers. As an extension, since experiments were performed at varying mass (tank filling) configurations, the parametric Loewner method was also successfully applied. An illustration is given on Figure 6.3, illustrating the ability to construct a parametric model from real data.

Let us now change topic and give an illustration of a research project obtained at **Onera** on the load control aspects. Within the aeronautical domain, minimisation of load envelope in response to discrete-time gusts is a challenging objective to reduce aircraft mass, and thus, consumption. To this aim, a wind tunnel experiment has been carried out at **Onera** to illustrate the gain brought by an active feedback control to limit the wing bending moments and vertical accelerations in response to generated gusts, from trans- to sub-sonic MACH numbers.

Data and model-based approximation for sub/transsonic gust load control in wind tunnel

On May 26th, 2015, a press article did spot the results obtained in **Onera** wind tunnel facility for both trans- and sub-sonic configurations to attenuate gust load. This result has been rendered possible thanks to the conjugation of many competences, including aeroelasticity, mechanics, fluid *etc.* and control. Figure 6.4 shows an artistic view of the experimental setup within the **Onera** S3Ch wind tunnel facility.

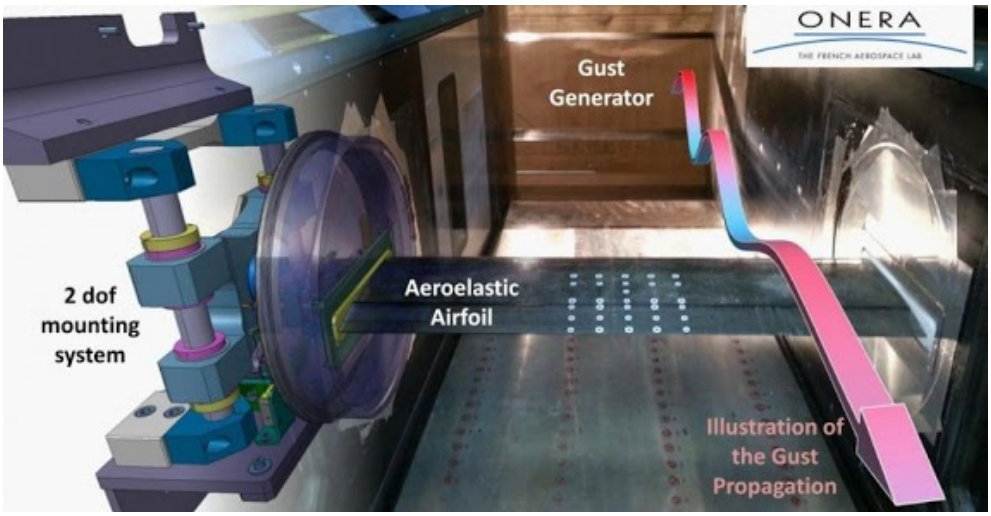


Figure 6.4: Experimental set-up in the **Onera** S3Ch wind tunnel: aeroelastic airfoil (foreground) and gust generator (background). Sensors on the wing are materialised by white dots.

After a calibration campaign for the gust generator, open-loop experiments were performed. As rooted on these experiments, sweeping the frequencies, a data-driven method (*i.e.* Loewner) using the `mor.lti` interface of the **MOR** Toolbox has been used to construct a dynamical model at varying wing angles and wind velocities.

Then, as rooted on these simple but representative models, an active closed-loop control has been designed using the model-based structured $\mathcal{H}_\infty$-norm oriented minimisation framework (`hinfstruct`). Interestingly, the experimental results obtained and reported on Figure 6.5, showed a huge reduction of the load gust impact on the wing loads. More interestingly and impressively, the experimental results were very close to the simulation ones, which claim in favour of this model-based approach and shows the effectiveness of the data-driven approximation methods, even in a very challenging problem (at the limit of the nonlinearity).

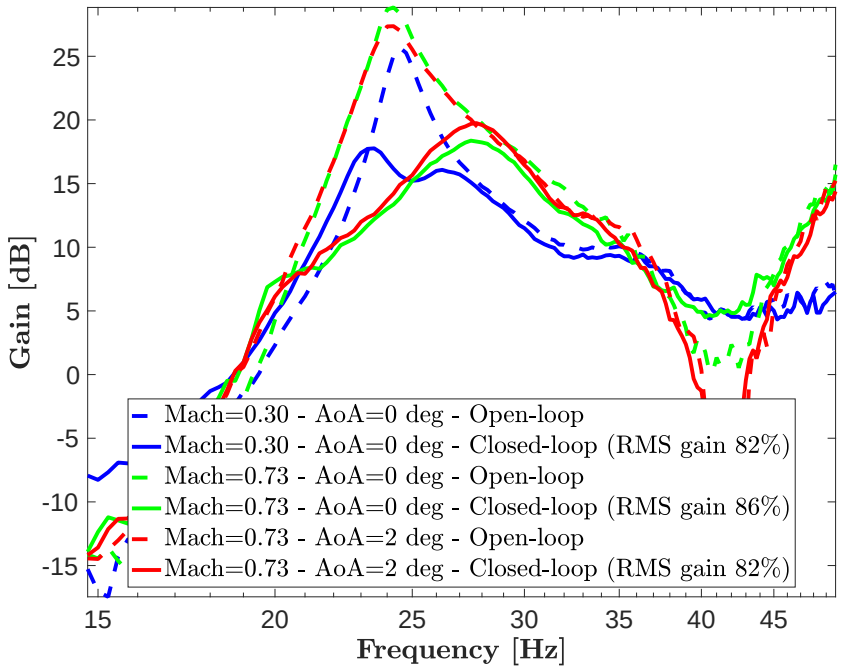


Figure 6.5: Experimental open-loop vs. closed-loop performances obtained in the S3Ch **Onera** wind tunnel facility.

6.3 Approximation as pivot for $\mathcal{L}_2$ systems analysis

The $\mathcal{L}_2$ meromorphic functions stability approximation, exposed in Chapter 5, is to the author's feeling, the most promising and exciting topic for future developments. Indeed, the proposed stability estimation process, even if mathematically not well closed, seems giving interesting preliminary results both in term of accuracy and scalability. The $\mathcal{L}_2$ meromorphic functions approximation by rational ones provides a powerful tool to attack multiple complex problems such as the stability (Chapter 5), the $\mathcal{H}_\infty$-, $\mathcal{H}_2$-norm estimation, and may be used as a controller tuning method for this larger model class. The following recalls the main idea.

Finite energy meromorphic function stability approximation

We show in Chapter 5 that it is possible to assess (or at least to estimate) the stability of any $\mathcal{L}_2$ meromorphic function. This includes among others the **TDS** systems on which many attention is given by the control community. As an example, assessing the stability of the following model as a function of gain k and delay τ,

$$\mathbf{H}(s,k,\tau) = \frac{1}{s + ke^{-\tau s}},$$

may be done using the **`mor.stability`** interface. Figure 6.6 illustrates the result obtained when evaluating this problem for multiple frozen $\{k,\tau\}$ couples (the white area stands as the stable part).

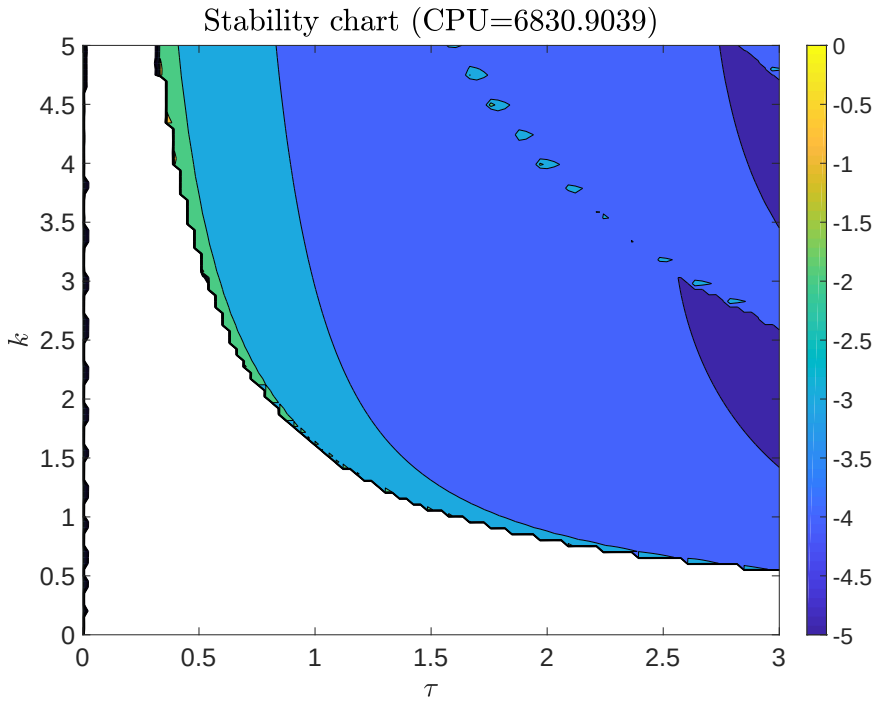


Figure 6.6: Stability estimation of a time-delay model (colours are in logarithmic scale).

Although the procedures shows a coloured area for τ-values close to 0, while the model is mathematically stable for any $k > 0$ (and $\tau = 0$), the overall stability chart is really close to the theoretical one described in Sipahi et al. (2011). By remembering that the main ingredients of the $\mathcal{L}_2$**-MFSA** is the interpolatory framework, connected with the $\mathcal{RH}_\infty$ approximation framework and to norm measures, it is quite impressive to figure out the possibilities of such a framework for systems analysis. To the author's point of view, there is no doubt that model approximation, for finite and infinite order models, may play a strategical role in either approximating some control engineers and practitioners metric, or even in initialising more "guaranteed" methods, in the context of large-scale complex problems.

6.4 Approximation and numerical tools

We saw along the manuscript that dealing with dynamical model approximation is strongly connected to complex functional analysis. Furthermore, when (very) large-scale or industrial model are used (potentially badly conditioned and obtained by complex processes), the numerical and computational considerations enter into the picture and probably become the most important barriers to cross. This last point is obviously linked with advances in linear algebra and implications in the construction of an efficient and robust algorithms, at the limit of the dynamical systems frame.

Consequently, last but not least, one important result presented in this manuscript, is the co-development, together with P. Vuillemin, of the MATLAB based **MOR** Toolbox which aims at implementing most of the methods gathered in this manuscript through an easy interface, tailored to both unfamiliar and expert users needs.

MOR Toolbox and its features (a MOR Digital Systems product)

The **MOR** Toolbox, now distributed by the **MOR Digital Systems** company (http://mordigitalsystems.fr/), embeds numerical tools for reduction and approximation of large-scale dynamical models and input-output data sets, as shown in Figure 2.1. In addition, the toolbox also provides tools for norm measurements and stability estimation. The following gives a glimpse of the content.

```
>> help mor
  MOR TOOLBOX
   Reduction and approximation of large-scale dynamical systems. See the
   documentation.

   Model approximation
    mor.lti         - main interface for LTI model approximation
    mor.guessOrder  - main interface for estimating the approximation
                      order of an LTI dynamical model

   Analysis
    mor.bode        - main interface for displaying Bode diagram of large
                      (sparse) LTI models, handle functions and input/output
                      data
    mor.bodeDamp    - main interface for displaying SISO bode diagram and
                      damping map of large (sparse) LTI models
    mor.norm        - main interface for computation of norms
    mor.sigma       - main interface for displaying Sigma diagram of large
                      (sparse) LTI models, handle functions and input/output
                      data
    mor.stability   - main interface for stability estimation of any
                      meromorphic model as a handle or (d)ss type

   Miscalleneous
    mor.demo        - load the demo file associated with the demonstrations in
                      the documentation
    mor.about       - display information about the toolbox
    mor.doc         - display the html documentation
```

Listing 6.1: **MOR** Toolbox MATLAB interface.

Moreover, by clicking on the documentation link, the window as the one given in Figure 6.7, is obtained. Then, by following the function link, leads to the function list as shown on Figure 6.8.

Without entering too much into details, the **`mor.lti`** function is the main interface for dynamical model and data-driven approximation. It gathers some of the algorithm presented in Chapters 2, 3 and 4. Then, **`mor.norm`** implements different model and data-driven norms presented in Chapter 1. Finally, **`mor.stability`** implements the stability estimation function exposed in Chapter 5. Some additional plotting functions tailored to the large-scale setting are also given. Additional functions such as the **`mor.guessOrder`**, a function suggesting an approximation order for a given dynamical model, are also embedded.

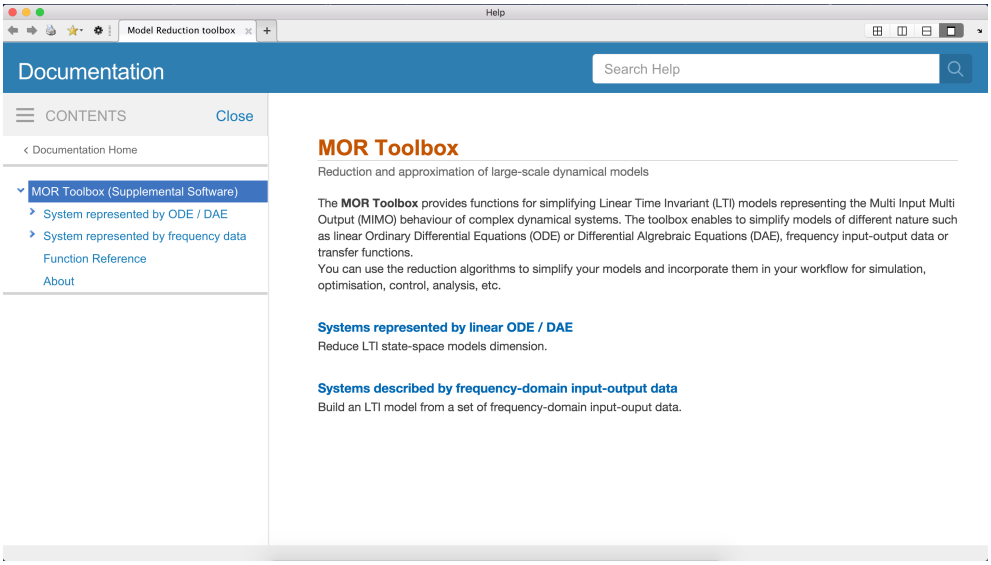


Figure 6.7: **MOR** Toolbox documentation.

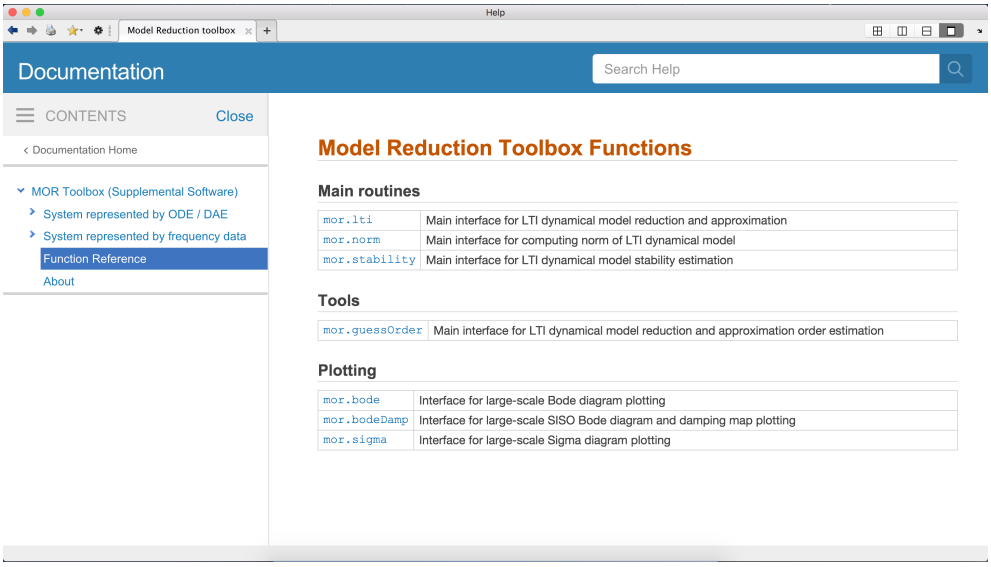


Figure 6.8: **MOR** Toolbox functions MATLAB documentation.

6.5 And now...

In this chapter we briefly remind the most important highlights and results described in the manuscript, without being too technical and avoiding mathematical details. In the next and final Chapter 7, we will give some hints for further developments in both short and long term.

Chapter 7

Future works and outlook

Don't be afraid of failure. This is the way to succeed.

(King) LeBron James

Contents

Following the conclusions presented in Chapter 6 and the overall manuscript philosophy and title[1], the perspectives can be considered along two major axis: first, *(i)* the dynamical model approximation and second, *(ii)* its application on real complex systems as well as on original mathematical problems. The former item, described in Section 7.1, concerns the extensions of the proposed methods and further developments within model approximation. The latter item, instead, focuses on the adaptation and application of these methods for the control and the analysis of large-scale models, and is detailed in Section 7.2. Moreover, as an other side effect of model approximation, discretisation of continuous-time dynamical systems is also an on-going project and is briefly pointed in Section 7.3. On top of these problems and objectives, the numerical considerations and technical (*e.g.* computational) limitations always need to be handled with careful attention. Indeed, numerical and linear algebra community development may be a game changer in the next years for model approximation.

7.1 Model approximation

Within the "classical" model approximation field, it is now clear that from a purely theoretical point of view, the **LTI ODE** approximation by an **LTI ODE** reduced model is well understood and mastered. Indeed, $\mathcal{H}_2$ optimality conditions have been derived using different approaches (tangential, gramian, complex optimisation, ...). The main remaining aspect concerns the treatment of even larger dimensional models, *e.g.* $n \gg 10^9$, which is more a problem for numerical researchers rather than control ones. Still, so many other problems may be investigated (some are already being studied by the community). Among them, the following may be mentioned:

- When considering the **LTI ODE** approximation by an **LTI ODE** reduced model, other criteria than the $\mathcal{H}_2$ one may be considered. As an example, in Chapter 3, the $\mathcal{H}_{2,\Omega}$ one is used, showing some

[1] "Large-scale dynamical model approximation and its applications".

interpolatory condition that would be interesting to develop in order to take advantage of these conditions to derive an **IRKA** like procedure for frequency limited model reduction. Similarly, even if the computational complexity nature the $\mathcal{H}_\infty$ and ν-gap metric may be binding, of the works on the approximation using these norms may also be investigated (Flagg et al., 2013; Sootla, 2014, see *e.g.*).

- When considering the finite order **LTI ODE** large-scale model case, as done with the $\mathcal{H}_2$ optimal input-output delay structured model approximation in Chapter 4, an interesting problem would be to generalise the approximation by structured reduced order model embedding a more complex structures (*e.g.* including internal delays, second order, ...). Up to the author's knowledge, so far, structured approximation exists but without $\mathcal{H}_2$ optimality conditions. In addition to Pontes Duff et al. (2018), an attempt has been done in Pontes Duff et al. (2016a) to approximate models by a reduced order one embedding one single internal delay and of dimension one, leading to a set of interpolatory constraints series. However, this latter result is far to be satisfactory for a practitioner and leaves room for further theoretical developments on the best way to exploit them in practice.

- Additionally, as pointed in Chapter 4, connections between the tangential interpolatory conditions and the Lyapunov, Sylvester and Loewner matrices should be done in the presence of input-output delays in the reduced order model. Indeed, although it might not be a primal objective from a practical point of view, it is, to the author's feeling, a nice way to close this result and a relevant theoretical interesting exercise.

- As for the for the **LTI ODE** cases, future work may consider bilinear (see *e.g.* Lin et al., 2009) or quadratic models $\mathcal{H}_2$ approximation. Indeed, as an illustration, bilinear models may be used to treat parametric models. For this model class, even if quite complicated, an $\mathcal{H}_2$-norm may be computed and used for model approximation. Interestingly, bilinear model also naturally appear in some applicative problems coming from the aerodynamic, aeroelasticity and in fluid mechanics domains, justifying an increasing interest.

- As an extension, approximation of **p-LTI ODE** parametric models (see *e.g.* Eid et al., 2009; Benner et al., 2015; Benner and Grundel, 2015) are also of great interest. The Loewner framework seems allowing it in an almost straightforward manner (see *e.g.* Ionita and Antoulas, 2014), but still, L_2-L_2 or L_∞-L_2 optimality conditions are missing. And, in addition, a framework to link the parameter and its - potential - variation velocity is not clear yet (see *e.g.* nd G. Mercère, 2014). This last point is of great importance in many control-oriented problem, where the **LPV** community did already developed a bundle of methods for analysis, estimation, control, but quite a few for approximation (see Lovera et al., 1998; Previdi and Lovera, 2003; Toth, 2010)). This point is quite active since some years and may be investigated even more. Obviously, this is also a way to attack the nonlinear approximation in a fancy framework.

- From a more practical viewpoint, investigations of on-line model approximation methods may also be done Amsallem and Farhat (2011). Indeed, for complex and industrial applications, this may be relevant to limit the experimental phases and tot deploy model approximation mechanism directly-on line. Within the aeronautical field, such a mechanism may used for the so-called modal monitoring, useful for flutter detection, predictive maintenance, *etc.* Moreover, by linking the model approximation with model-based control strategies such as Model Predictive Control (or **MPC**), it might be interesting approximating on-line a model and updating the model constraint in the **MPC** process.

- Finally, and more directly connected to the analysis issues, as shown along the manuscript, shift points of **IRKA** like methods seem to embed informations which can be relevant for *e.g.* $\mathcal{H}_\infty$-norm estimation (shift usually glue to the pick values) or be used for optimal signals generation in view of complex systems simulations (shift may be connected to pulsation and phase signals where energy is more relevant)*etc.*

7.2 Model approximation for control

As one of my research activity, some links between model approximation and controller design are of great interest for (on-going and) future developments. Among them, model approximation may serve control...

7.2.1 ... for $\mathcal{H}_\infty$ robust control

As shown at multiple occurrences along the manuscript and in Chapter 6 through the industrial applications, model approximation and robust control are well connected. Indeed, in the robust framework the mismatch error modelling may be encapsulated in an uncertain block and included in the robust control synthesis problem, ensuring that the performances and stability are maintained on the original model, while synthesis done on the reduced one (see *e.g.* Vuillemin et al., 2017). This point raises multiple questions such as the mismatch error estimation (accuracy vs. numerical cost), the trade-off between a (too) simple model and the controller obtained by the optimiser (being non-convex)...

On top of these points, we saw on many applications that interpolation point mostly converge to high energy frequencies. As a direct consequence, an interesting point may to use these interpolation points to initialise an $\mathcal{H}_\infty$ norm estimation and eventually controller optimisation. The symbioses of model approximation with $\mathcal{H}_\infty$ controller synthesis may then open the road to large-scale, low complexity controller design in a numerically tractable way. To the experience of the author, it is no doubt that this strong connection may lead to fasten controller design developments phases.

7.2.2 ... for ILC control

The **ILC** for Iterative Learning Control is basically an open-loop control method allowing to tune a reference signal according to a criteria evaluated on the basis of the previous actions. The main purpose of **ILC** is to construct control signals that are tuned to the repeated process under consideration. To deploy this technique, on the basis of experiment, a model needs to be identified to adjust the signal following a descent method. In Kocan et al. (2018), the data-driven approximation methods were investigated to obtain such a model. The outlooks attached to this control family are twofolds: first the model approximation may be embedded in the process (*e.g.* in an on-line framework), second, when the optimal control signal is reached, a controller identification, or data-driven and meromorphic functions approximation may be performed to obtain a rational function and recover a standard closed-loop framework (see *e.g.* the preliminary result in Kocan et al., 2019).

7.2.3 ... for PDE control

Model approximation of infinite dimensional models is rendered possible thanks to the versatile interpolatory framework. As a consequence, as soon as a finite (low) order model has been obtained, one mays attack the infinite dynamical model control through the classical finite order angle. In addition, as exposed in Chapter 5, interpolation may be a reliable tool to estimate stability and norms (Pontes Duff et al., 2016b) and may then be embedded in an optimisation framework to directly synthesise a control law, based on the **PDE** models or any meromorphic $\mathcal{L}_2$ functions. Some preliminary results are under investigation using the benchmarks proposed by Curtain and Morris (2010), Caldeira et al. (2018) and Pilbauer et al. (2018). The latter considers the control of a drilling system which stability as a function of controller gains of a PI structure is illustrated in Figure 7.1. Similar work may be done by involving some tracking, norms, disturbance rejection performances *etc.* but this would require a dedicated attention.

7.2.4 ... for data driven-control

In Campi et al. (2002), authors did present a method for data-driven controller synthesis. This method basically recasts the control design problem as an identification one. One great advantage of such an approach is that it provides a controller tailored to the actual system or model. In its initial version, time-domain methods were used, but did consider a set of predefined poles for the controller. This approach has been also largely extended and improved in recent paper from *e.g.* Formentin et al. (2013, 2014). Recently,

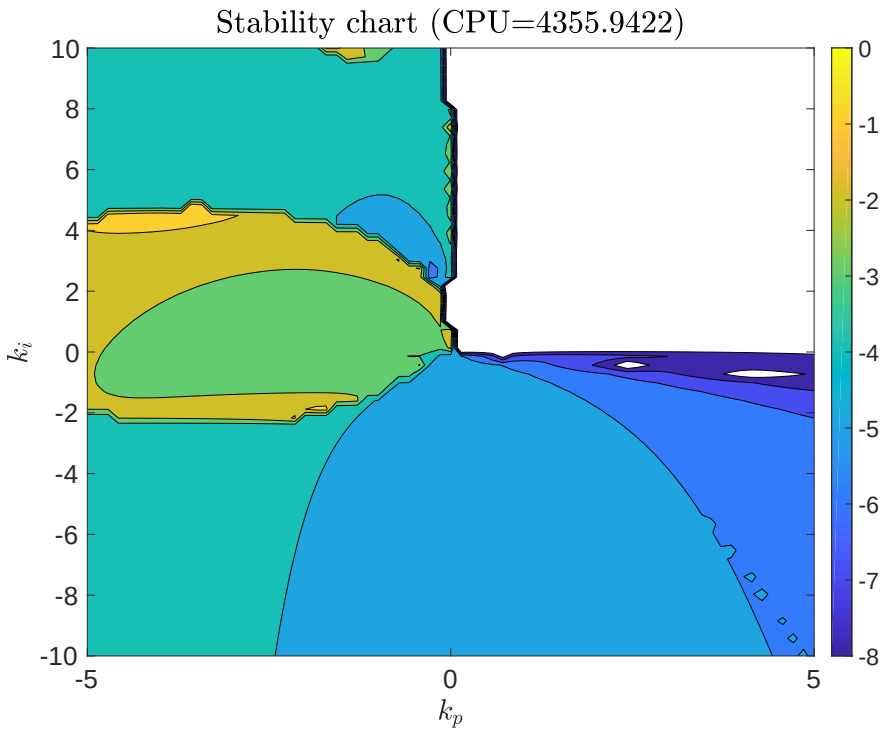


Figure 7.1: Drilling system stability area as a function of the PI controller parameters.

Kergus et al. (2017) pushed the interpolatory framework, and especially the Loewner one, into the process, enabling designing **MIMO** controllers without a-priori pole selection. As a perspective, such an approach may be enriched to handle noise (see Kergus et al., 2018), structural constraints, *etc.*

7.3 Model approximation for models discretisation

Very recently, Vuillemin and Poussot-Vassal (2019) did exploit the interpolation framework in order to discretise a dynamical system or a controller. Indeed, the idea is based on the fact that discrete-time models uses the z-transform, where

$$z = e^{sh} \text{ then } z = \frac{1}{h}\log(z),$$

where h is the constant sampling time. Usually, when discretisation is performed, special Mobius transformation (with $\{a, b, c, d\} \in \mathbb{R}$) as follows is used:

$$s = \frac{az + b}{cz + d}.$$

The main benefit of using specific Mobius transform stands in the stability or instability preservation, and in the preservation of the rational degree of the underlying transfer function. One of the most famous one is the so-called Tustin transform where $a = -b = 2$ and $c = d = h$. It's quite largely used since it has the specificity of mapping the complex left hand plane with the unit circle centred in zero. However, such a transformation may lead to signal distortion, while an other one (*e.g.* with different $\{a, b, c, d\}$ coefficient set) may be better in term of input-output matching. As a direct extension, the same comment may be done by approximating the Laplace variable, instead of the Mobius transform, by a m-th order rational function as:

$$s = \frac{\sum_{i=1}^{m} b_i z^i}{\sum_{i=1}^{m} a_i z^i},$$

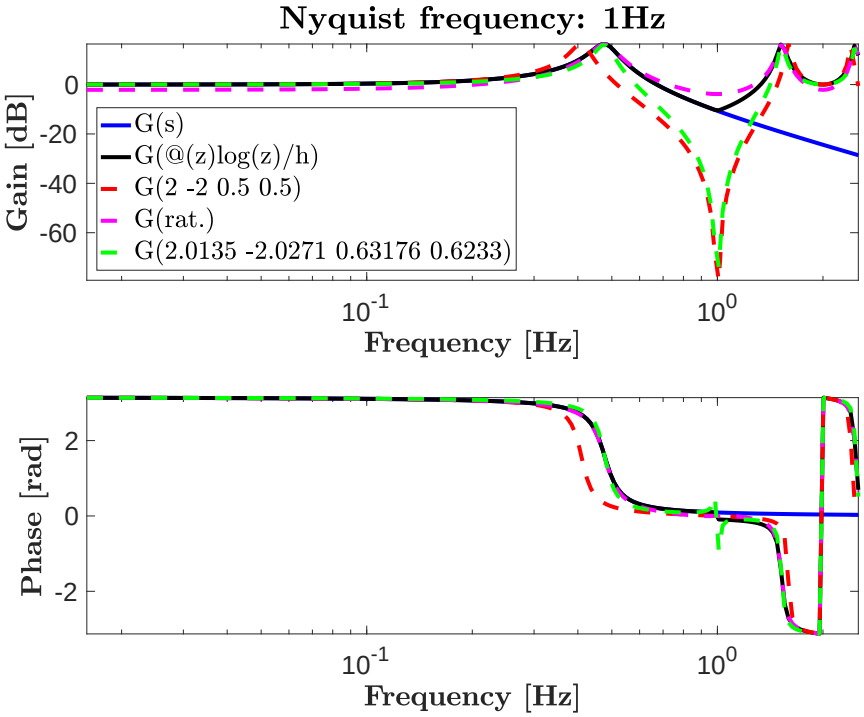


Figure 7.2: Bode responses of $\mathbf{H}(s)$ (solid blue) and its different discretisation. Exact (solid black), Tustin (dashed red), optimised Mobius form (dashed green) and rational form of order 12 (dashed pink).

where $a_i \in \mathbb{R}$ and $b_i \in \mathbb{R}$ are the tuning variable. In Vuillemin and Poussot-Vassal (2019), authors exploit the interpolatory versatility to construct such a (perfect) discretised model, with a potentially higher dimension. As an illustration of this idea, one may consider the following simple **LTI** continuous-time dynamical model:

$$\mathbf{H}(s) = \frac{Kw_0^2}{s + mw_0 s + w_0^2},$$

where $K = -1$, $w_0 = 3$ and $m = 0.15$. Let us now consider that one aims at sampling such a system, for simulation, control, *etc.* with a constant time period $h = 0.5$s. According to the discretisation method, one obtains the Bode responses shown on Figure 7.2 and the impulse response mismatch error reported on Figure 7.3.

By inspecting these figures, the following remarks may be done: first, the Tustin method yeld to $\mathbf{G}(a, b, c, d)$ with $a = -b = 2$ and $c = d = h$, provides a larger mismatch than the modified transformation obtained by a Mobius transform $\mathbf{G}(a, b, c, d)$ with $a = 2.0135$, $b = -2.0271$, $c = 0.63176$ and $d = 0.6233$. This means that the Tustin method is not necessary the most appropriate transformation to apply. Still one may consider pre-wrapping to improve the Tustin form. Second, the rational approximation $\mathbf{G}(\text{rat})$ (here with a rational function of dimension 12) leads to even better results, with a substantially lower impulse response error. This is of course obtained as the price of a more complex model of order 12 instead of 2.

Without entering into the details, to the author's feeling, the potential effects of such approach are very large (preliminary results are under process in Vuillemin and Poussot-Vassal (2019)):

- for controller, the discretisation may be lossless, at the price of an increased controller complexity,
- for fixed-time simulation (employing simple integration schemes), it may then be possible to reduce the sampling time, and then to increase the simulation time, while keeping the model accuracy.

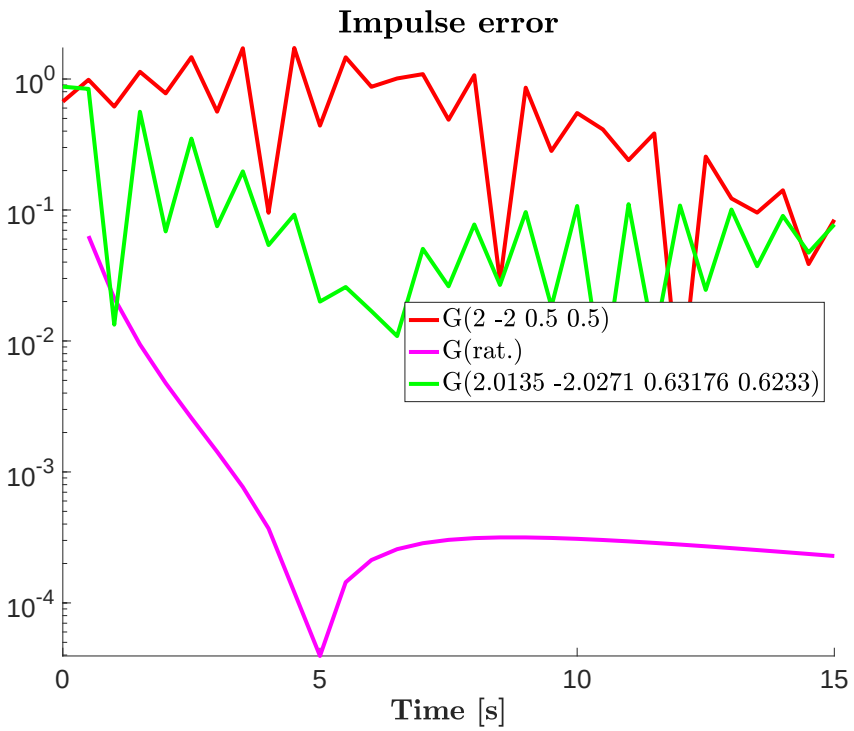


Figure 7.3: Impulse responses error between $\mathbf{H}(s)$ and its different discretisation. Exact (solid black), Tustin (dashed red), optimised Mobius form (dashed green) and rational form of order 12 (dashed pink).

7.4 Conclusions

In this last chapter, some of the relevant outlooks and possible perspectives to be studied in the forthcoming years have been briefly gathered. Obviously, these perspectives may be taken as a snapshot of the actual feeling of the author since breaking methods should be discovered and reconfigure the perspectives. Moreover, some of them may be theoretically trivial to deploy but rather complicated in a stable numerical way. Indeed, the model approximation research field composes an active and large community covering fields from control theory, functional analysis and numerical analysis. To the author's feeling and as pointed along this manuscript and along the sections of this final chapter, the data-driven methods linked with the interpolatory framework, clearly appear as good candidates for further interesting developments within the dynamical model approximation, control, analysis domains.

Bibliography

Abidi, O., Hached, M., and Jbilou, K. (2017). A global rational Arnoldi method for model reduction. *Journal of Computational and Applied Mathematics*, 325:175–187.

Alkhoury, Z., Petreczky, M., and Mercère, G. (2018). Identifiability of affine linear parameter-varying models. *Automatica*, 80:62–74.

Amsallem, D. and Farhat, C. (2011). An online method for interpolating linear parametric reduced-order models. *SIAM Journal of Scientific Computing*, 33(5):2169–2198.

Anic, B., Beattie, C., Gugercin, S., and Antoulas, A. (2013). Interpolatory weighted-$\mathcal{H}_2$ model reduction. *Automatica*, 49(5):1275–1280.

Antoulas, A., Lefteriu, S., and Ionita, A. (2016). *Model reduction and approximation theory and algorithms*, chapter A tutorial introduction to the Loewner framework for model reduction. SIAM, Philadelphia. P. Benner, A. Cohen, M. Ohlberger and K. Willcox Eds.

Antoulas, A. C. (2005). *Approximation of Large-Scale Dynamical Systems*. Advanced Design and Control, SIAM, Philadelphia.

Antoulas, A. C. (2009). An overview of model reduction methods and a new result. In *Proceedings of the 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, pages 5357–5361, Shanghai, China.

Antoulas, A. C., Beattie, C. A., and Gugercin, S. (2010). *Efficient Modeling and Control of Large-Scale Systems*, chapter 1 - Interpolatory model reduction of large-scale dynamical systems, pages 3–58. Robotics. Springer-Verlag.

Antoulas, A. C., Sorensen, D. C., and Gugercin, S. (2001). A survey of model reduction methods for large-scale systems. *Structured Matrices in Operator Theory, Numerical Analysis, Control, Signal and Image Processing, Contemporary Mathematics, AMS publications*, 280:193–219.

Apkarian, P. and Noll, D. (2006). Nonsmooth $\mathcal{H}_\infty$ Synthesis. *IEEE Transaction on Automatic Control*, 51(1):71–86.

Arzelier, D. (2014). *Théorie de Lyapunov, commande robuste et optimisation*. HDR. thesis, Toulouse University, Toulouse, France.

Bai, Z., Demmel, J., Dongarra, J., Ruhe, A., and Van der Vorst, H. (2010). *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM Advances in Design and Control.

Balas, G. J., Bokor, J., and Szabo, Z. (2003). Invariant Subspaces for LPV Systems and their Application. *IEEE Transaction on Automatic Control*, 48:2065–2069.

Baratchart, L. (1986). Existence and Generic Properties of L^2 Approximants for Linear Systems. *IMA Journal of Mathematical Control & Information*, 3:89–101.

Baratchart, L., Cardelli, M., and Olivi, M. (1991). Identification and rational $\mathcal{L}_2$ approximation: A gradient algorithm. *Automatica*, 27(2):413–417.

Barbagallo, A., Sipp, D., and Schmid, P. (2008). Closed-loop control of an open cavity flow using reduced-order models. *Journal of Fluid Mechanics*, 641:1–50.

Baur, U., Beattie, C., Benner, P., and Gugercin, S. (2009). Interpolatory projection methods for parametrized model reduction. Technical Report CSC/09-08, Fakultät fur Mathematik, TU Chemnitz, 09107 Chemnitz, FRG.

Baur, U., Beattie, C., Benner, P., and Gugercin, S. (2011). Interpolatory Projection Methods for Parameterized Model Reduction. *SIAM Journal on Scientific Computing*, 33(5):2489–2518.

Beattie, C. and Gugercin, S. (2009a). A Trust Region Method for Optimal $\mathcal{H}_2$ Model Reduction. In *Proceedings of the 48th IEEE Conference on Decision and Control*, pages 5370–5375, Shanghai, China.

Beattie, C. and Gugercin, S. (2009b). Interpolatory projection methods for structure-preserving model reduction. *System & Control Letters*, 58(3):225–232.

Beattie, C. and Gugercin, S. (2012). Realization-independent $\mathcal{H}_2$-approximation. In *Proceedings of the 51st IEEE Conference on Decision and Control*, pages 4953–4958.

Beattie, C. and Gugercin, S. (2016). *Model reduction and approximation theory and algorithms*, chapter Model Reduction by Rational Interpolation. SIAM, Philadelphia. P. Benner, A. Cohen, M. Ohlberger and K. Willcox Eds.

Benner, P. and Breiten, T. (2012). Interpolation-Based $\mathcal{H}_2$-Model Reduction of Bilinear Control Systems. *SIAM Journal on Matrix Analysis and Applications*, 33(3):859–885.

Benner, P. and Breiten, T. (2015). Two-sided projection methods for nonlinear model order reduction. *SIAM Journal on Scientific Computing*, 37(2):B239–B260.

Benner, P. and Damm, T. (2011). Lyapunov Equations, Energy Functionals, and Model Order Reduction of Bilinear and Stochastic Systems. *SIAM Journal on Control and Optimization*, 49(2):686–711.

Benner, P. and Grundel, S. (2015). Model Order Reduction for a Family of Linear Systems with Application in Parametric and Uncertain Systems. *Applied Mathematics Letters*, 39:1–6.

Benner, P., Grundel, S., Himpe, C., Huck, C., Streubel, T., and Tischendorf, C. (2018). *Differential algebraic equations forum*, chapter Gas network benchmark models. Springer, Berlin, Heidelberg.

Benner, P., Gugercin, S., and Willcox, K. (2015). A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531.

Benner, P., Kohler, M., and Saak, J. (2011). Sparse-Dense Sylvester Equations in $\mathcal{H}_2$-Model Order Reduction. Technical report, Max Plank Institute.

Benner, P., Mehrmann, V., and Sorensen, D. C. (2005). *Dimension Reduction of Large-Scale Systems*. Lecture Notes in Computational Science and Engineering, Vol. 45, Springer-Verlag, Berlin/Heidelberg.

Benner, P. and Mitchell, T. (2017). Faster and more Accurate Computation of the $\mathcal{H}_\infty$-Norm via Optimization. *arXiv Preprint arXiv:1707.02497*, pages 327–342.

Benner, P. and Stykel, T. (2016). *Surveys in Differential-Algebraic Equations IV. Differential-Algebraic Equations Forum*, chapter Model order reduction for differential-algebraic equations: a survey. Springer. A. Ilchmann and T. Reis Eds.

Benner, P. and Werner, S. (2017). MORLAB-3.0 – model order reduction laboratory. see also: http://www.mpi-magdeburg.mpg.de/projects/morlab.

Bentbib, A. and Jbilou, K. (2018). A computational global tangential Krylov subspace method for model reduction of large-scale dynamical systems. *Journal of Scientific Computing*, 75(3):1612–1632.

Biannic, J. (2010). *Theoretical Contributions to Aerospace Control Systems: Beyond Linear Control*. HDR. thesis, Onera, ISAE, Toulouse University, Toulouse, France.

Borggaard, J. and Gugercin, S. (2014). *Active Flow and Combustion Control*, volume 127 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, chapter Model Reduction for DAEs with an Application to Flow Control, pages 381–396. R. King editors, Springer-Verlag.

Boyd, S. and Balakrishnan, V. (1989). A regularity result fir the singular values of a transfer matrix and a quadratically convergent algorithm for computing the $\mathcal{L}_\infty$-norm. In *Proceedings of the 28th IEEE Conference on Decision and Control*, pages 954–955, Tampa, Florida, USA.

Boyd, S., Balakrishnan, V., and Kabamba, P. (1988). On computing the $\mathcal{H}_\infty$ norm of a transfer matrix. In *Proceedings of the American Control Conference*, pages 396–397, Atlanta, Georgia.

Brandwood, D. (1983). A complex gradient operator and its application in adaptive array theory. In *Proceedings of the of the Conference on Communications, Radar and Signal Processing*, pages 11–16.

Breiten, T. and Damm, T. (2010). Krylov subspace methods for model order reduction of bilinear control systems. *System & Control Letters*, 59(8):443–450.

Briat, C. (2015). *LPV & Time-Delay Systems - Analysis, Observation, Filtering & Control*, volume 3. Springer-Heidelberg, Germany.

Briat, C. and Verriest, E. (2009). A New Delay-SIR Model for Pulse Vaccination. *Biomedical Signal Processing and Control*, 4:272–277.

Bruinsma, N. and Steinbuch, M. (1990). A fast algorithm to compute the $\mathcal{H}_\infty$-norm of a transfer function matrix. *System & Control Letters*, 14(4):287–293.

Brunot, M. (2017). *Identification of rigid industrial robots - A system identification perspective*. Ph.D. thesis, Onera, ISAE, Toulouse University, Toulouse, France.

Bruzelius, F., Pettersson, S., and Breitholz, C. (2004). Linear parameter varying descriptions of nonlinear systems. In *IEEE American Control Conference*, pages 1374–1379, Boston, Massachusetts.

Burke, J. V., Henrion, D., Lewis, A. S., and Overton, M. L. (2006). HIFOO - A MATLAB Package for Fixed-order Controller Design and $\mathcal{H}_\infty$ Optimization. In *Proceedings of the IFAC Symposium on Robust Control Design*, Toulouse, France.

Caldeira, A., Prieur, C., Coutinho, D., and Leite, V. (2018). Regional stability and stabilization of a class of linear hyperbolic systems with nonlinear quadratic dynamic boundary conditions. *European Journal of Control*, 43:46–56.

Campi, M. C., Lecchini, A., and Savaresi, S. M. (2002). Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*, 28(8):1337–1346.

Chahlaoui, Y., Lemonnier, D., Vandendorpe, A., , and Dooren, P. V. (2006). Second-order balanced truncation. *Linear Algebra and its Applications*, 415(2):373–384.

Chilali, M. and Gahinet, P. (1996). $\mathcal{H}_\infty$ Design with Pole Placement Constraints: an LMI Approach. *IEEE Transaction on Automatic Control*, 41(3):358–367.

Curtain, R. and Morris, K. (2010). Transfer functions of distributed parameter systems: A tutorial. *Automatica*, 45(5):1101–1116.

Dalmas, V., Robert, G., Poussot-Vassal, C., Pontes Duff, I., and Seren, C. (2016). From infinite dimensional modelling to parametric reduced order approximation: Application to open-channel flow for hydroelectricity. In *Proceedings of the 15th European Control Conference*, pages 1982–1987, Aalborg, Denmark.

Dashkovskiy, S. and Mironchenko, A. (2018). Input-to-state stability of infinite-dimensional control systems. *Mathematics of Control, Signals, and Systems*, 25:1–35.

Demmel, J. and Nguyen, H. (2014). Parallel Reproducible Summation. *IEEE Transactions on Computers*, 64(7):2060–2070.

Demourant, F. and Poussot-Vassal, C. (2017). A new frequency-domain subspace algorithm with restricted poles location through LMI regions and its application to a wind tunnel test. *International Journal of Control*, 90(4):779–799.

Doyle, J. C., Glover, K., Khargonekar, P., and Francis, B. (1989). State Space Solution to Standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ Control Problems. *IEEE Transaction on Automatic Control*, 34(8):831–847.

Drmac, Z., Gugercin, S., and Beattie, C. (2015a). Quadrature-based vector fitting for discretized $\langle_2$ approximation. *SIAM Journal on Scientific Computing*, 37(2):625–652.

Drmac, Z., Gugercin, S., and Beattie, C. (2015b). Vector fitting for matrix-valued rational approximation. *arXiv:1503.00411*.

Eid, R., Lohmann, B., and Panzer, H. (2009). Interpolation-based Parametric Model Reduction using Krylov Subspaces. In *Presented at the Autumn School on Future Developments in Model Order Reduction*, Terschelling, The Netherlands.

Feldman, P. and Freund, R. (1995). Efficient linear circuit analysis by Padé approximation via a Lanczos method. *IEEE Transactions on Computer-Aided Design*, 14:639–649.

Flagg, G., Beattie, C., and Gugercin, S. (2013). Interpolatory $\mathcal{H}_\infty$ model reduction. *System & Control Letters*, 62(7):567–574.

Formentin, S., Karimi, A., and Savaresi, S. M. (2013). Optimal input design for direct data-driven tuning of model-reference controllers. *Automatica*, 49(6):1874–1882.

Formentin, S., van Heusden, K., and Karimi, A. (2014). A comparison of model-based and data-driven controller tuning. *International Journal of Adaptive Control and Signal Processing*, 28(10):882–897.

Francis, B. A. and Doyle, J. C. (1987). Linear Control Theory with an $\mathcal{H}_\infty$ Criterion. *SIAM Journal of Control and Optimization*, 25:815–844.

Fuhrmann, P. (1994). A duality theory for robust stabilization and model reduction. *Linear Algebra and its Applications*, 203-204:471–578.

Fulcheri, P. and Olivi, M. (1998). Matrix rational $\mathcal{H}_2$ approximation: A gradient algorithm based on Schur analysis. *SIAM Journal on Control and Optimization*, 36(6):2103–2127.

Gahinet, P. and Apkarian, P. (1992). Numerical computation of the $\mathcal{L}_\infty$ norm revisited. In *Proceedings of the 31st IEEE Conference on Decision and Control*, pages 2257–2258, Tucson, Arizona, USA.

Gahinet, P. and Apkarian, P. (1994). An linear matrix inequality approach to $\mathcal{H}_\infty$ control. *International Journal of Robust and Nonlinear Control*, 4(4):421–448.

Gahinet, P. and Apkarian, P. (2011). Structured $\mathcal{H}_\infty$ Synthesis in Matlab. In *Proceedings of the 18th IFAC World Congress*, Milan, Italy.

Gallivan, K. A., Vanderope, A., and Van Dooren, P. (2004a). Model reduction of MIMO systems via tangential interpolation. *SIAM Journal of Matrix Analysis and Application*, 26(2):328–349.

Gallivan, K. A., Vanderope, A., and Van Dooren, P. (2004b). Sylvester equations and projection-based model reduction. *Journal of Computational Applied Mathematics*, 162:213–229.

Garulli, A., Hansson, A., Pakazad, S., Masi, A., and Wallin, R. (2013). Robust finite-frequency analysis of uncertain systems with application to flight comfort analysis. *Control Engineering Practice*, 21(6):887 – 897.

Gawronski, W. (2004). *Advanced Structural Dynamics and Active Control of Structures*. Springer, Edition 1.

Ghosh, D., Bombois, X., Huillery, J., Scorletti, G., and Mercère, G. (2018). Optimal identification experiment design for LPV systems using the local approach. *Automatica*, 87:258–266.

Glover, K. (1984). All Optimal Hankel Norm Approximation of Linear Multivariable Systems, and Their $\mathcal{L}_\infty$ error Bounds. *International Journal Control*, 39(6):1145–1193.

Glover, K., Curtain, R., and Partington, J. (1988). Realisation and approximation of linear infinite-dimensional systems with error bounds. *SIAM Journal on Control and Optimization*, 26(4):863–898.

Gonchard, A. (2003). Rational approximation of analytic functions. *Sovremennye Problemy Matematiki*, 1:83–106.

Gozse, I., Luspay, T., Péni, T., Szabo, Z., and Vanek, B. (2016). Model order reduction of LPV systems based on parameter varying modal decomposition. In *Proceedings of the 55th IEEE Conference on Decision and Control*, pages 7459–7464, Las Vegas, Nevada, USA.

Grimme, E., Sorensen, D., and Dooren, P. V. (1996). Model reduction of state space systems via an implicitly restarted Lanczos method. *Numerical Algorithms*, 12(1):1–31.

Grimme, E. J. (1997). *Krylov Projection Methods for Model Reduction*. PhD thesis, University of Illinois.

Gu, K., Kharitonov, V., and Chen, J. (2003). *Stability of Time-Delay Systems*. Springer.

Gugercin, S. (2008). An iterative SVD-Krylov based method for model reduction of large-scale dynamical systems. *Linear Algebra and its Applications*, 428(8-9):1964–1986.

Gugercin, S. and Antoulas, A. (2004). A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77(8):748–766.

Gugercin, S. and Antoulas, A. C. (2006). Model reduction of large-scale systems by least squares. *Linear Algebra and its Applications*, 415(2-3):290–321.

Gugercin, S., Antoulas, A. C., and Beattie, C. A. (2008). $\mathcal{H}_2$ Model Reduction for Large Scale Linear Dynamical Systems. *SIAM Journal on Matrix Analysis and Applications*, 30(2):609–638.

Gugercin, S., Polyuga, R. V., Beattie, C., and van der Schaft, A. (2012). Structure-preserving tangential interpolation for model reduction of port-Hamiltonian systems. *Automatica*, 48(9):1963–1974.

Gugercin, S. and Willcox, K. (2008). Krylov projection framework for Fourier model reduction. *Automatica*, 44(1):209–215.

Guiver, C. and Opmeer, M. (2014). Model reduction by balanced truncation for systems with nuclear Hankel operators. *SIAM Journal on Control and Optimization*, 52(2):1366–1401.

G.W.Stewart (1972). On the sensitivity of the eigenvalue problem $Ax = \lambda Bx$. *SIAM Journal on Numerical Analysis*, 9:669–684.

Hached, M. and Jbilou, K. (2017). Numerical solutions to large-scale differential Lyapunov matrix equations. *Numerical Algorithms*, pages 1–17.

Halevi, Y. (1992). Frequency weighted model reduction via optimal projection. *IEEE Transactions on Automatic Control*, 37(10):1537–1542.

Halevi, Y. (1996). Reduced-order models with delay. *International Journal of Control*, 64(4):733–744.

Hammarling, S. (1982). Numerical solution of the stable, non-negative definite Lyapunov equation. *IMA Journal of Numerical Analysis*, 2:303–325.

Heinkenschloss, M., Reis, T., and Antoulas, A. (2011). Balanced truncation model reduction for systems with inhomogeneous initial conditions. *Automatica*, 47(3):559–564.

Heyouni, M. and Jbilou, K. (2009). An Extended Block Arnoldi Algorithm for Large-Scale Solutions of the Continuous-Time Algebraic Riccati Equation. *Electronic Transactions on Numerical Analysis*, 33:53–62.

Higham, N. (2002). *Accuracy and stability of numerical algorithms (2nd edition)*. SIAM.

Himpe, C. (2016). emgr - The Empirical Gramian Framework. *arXiv:1611.00675*.

Himpe, C. (2017). *Combined State and Parameter Reduction for Nonlinear Systems with an Application in Neuroscience*. PhD thesis, Westfalische Wilhelms Universitat Munster.

Himpe, C., Leibner, T., Rave, S., and Saak, J. (2013). Fast Low-Rank Empirical Cross Gramians. In *Proceedings of the 17th Applied Mathematics and Mechanics*, pages 841–842, Hangzjou, China.

Himpe, C. and Ohlberger, M. (2013). A unified software framework for empirical gramians. *Journal of Mathematics*.

Hochstenbach, M. and Sleijpen, G. (2008). Harmonic and refined extraction methods for the polynomial eigenvalue problem. *Numerical Linear Algebra with Applications*, 15:35–54.

Hochstenbachi, M. (2001). A Jacobi-Davidson type SVD method. *SIAM Journal on Scientific Computing*, 23:606–628.

Hoffman, K. (1962). *Banach spaces of analytic functions*. Prentice Hall.

Hyland, D. C. and Bernstein, D. S. (1995). The optimal projection equations for model reduction and the relationships among the methods of Wilson, Skelton and Moore. *IEEE Transaction on Automatic Control*, 30(12):1201–1211.

Imran, M. and Ghafoor, A. (2015). Model reduction of descriptor systems using frequency limited Gramians. *Journal of the Franklin Institute*, 352:33–51.

Ionita, A. and Antoulas, A. (2014). Data-Driven Parametrized Model Reduction in the Loewner Framework. *SIAM Journal on Scientific Computing*, 36(3):A984–A1007.

Ionita, C. (2013). *Lagrange rational interpolation and its applications to approximation of large-scale dynamical systems*. PhD thesis, University of Houston, Texas, USA.

Ionutiu, R., Rommes, J., and Antoulas, A. C. (2008). Passivity-Preserving Model Reduction Using Dominant Spectral-Zero Interpolation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 27(12):2250–2263.

Izmailov, R. (1996). Analysis and optimization of feedback control algorithms for data transfers in high-speed networks. *SIAM Journal on Control Optimization*, 34:1767–1780.

Jacob, B., Nabiullin, R., Partington, J. R., and Schwenninger, F. (2016). On input-to-state-stability and Integral input-to-state-stability for parabolic boundary control systems. In *Proceedings of the 55th IEEE Conference on Decision and Control*, pages 2265–2269, Las Vegas, Nevada, USA.

Janot, A. (2017). *On the identification of continuous-time inverse dynamic model of electromechanical systems operating in closed loop with an instrumental variable approach: application to industrial robots*. HDR. thesis, Onera, ISAE, Toulouse University, Toulouse, France.

Jarlebring, E., Meerbergen, K., and Michiels, W. (2010). A Krylov Method for the Delay Eigenvalue Problem. *SIAM Journal on Scientific Computing*, 32(6):3278–3300.

Jongsma, H., Mlinari, P., Grundel, S., Benner, P., and Trentelman, H. (2018). Model Reduction of Linear Multi-Agent Systems by Clustering and with $\mathcal{H}_2$- and $\mathcal{H}_\infty$ Error Bounds. *Mathematics of Control, Signals, and Systems*, 30.

Kahan, W. (1987). Branch cuts for complex elementary functions, or much ado about nothing sign bit. In *Proceedings of the Joint IMA/SIAM Conference on the State of the Art in Numerical Analysis*, pages 165–211.

Kergus, P., Formentin, S., Poussot-Vassal, C., and Demourant, F. (2018). Data-driven control design in the Loewner framework: dealing with stability and noise. In *Proceedings of the European Control Conference*, Limassol, Cyprus.

Kergus, P., Poussot-Vassal, C., Demourant, F., and Formentin, S. (2017). Frequency-domain data-driven control design in the Loewner framework. In *Proceedings of the 20th IFAC World Congress*, pages 2095–2100, Toulouse, France.

Kocan, O., Manecy, A., and Poussot-Vassal, C. (2018). A Practical Method to Speed-up the Experimental Procedure of Iterative Learning Controllers. In *Proceedings of the IEEE International Conference on Intelligent Robots*, Madrid, Spain.

Kocan, O., Poussot-Vassal, C., and Manecy, A. (2019). Toward automatic tuning of linear controllers using iterative learning control. In *Proceedings of the European Control Conference*, Naples, Italy.

Kohler, M. (2014). On the closest stable descriptor system in the respective spaces $\mathcal{RH}_2$ and $\mathcal{RH}_\infty$. *Linear Algebra and its Applications*, 443:34–49.

Kurschner, P. (2010). Two-sided Eigenvalue Algorithms for Modal Approximation. Master's thesis, Chemnitz University of Technology, Faculty of Mathematics at Chemnitz University of Technology.

Lefteriu, S. and Antoulas, A. (2013). On the convergence of the vector-fitting algorithm. *IEEE transaction on Microwave Theory and Techniques*, 61(4):1435–1443.

Lehoucq, R. B. and Sorensen, D. C. (1996). Deflation Techniques for an Implicitly Restarted Arnoldi Iteration. *SIAM Journal on Matrix Analysis & Applications*, 17(4):789–821.

Leibfritz, F. (2003). COMPl$_e$ib, COnstraint Matrix-optimization Problem LIbrary - a collection of test examples for nonlinear semidefinite programs, control system design and related problems. Technical report, Universitat Trier.

Lin, Y., Bao, L., and Wei, Y. (2009). Order reduction of bilinear MIMO dynamical systems using new block Krylov subspaces. *Computer and Mathematics with Applications*, 58:1093–1102.

Ljung, L. (2013). Convexity issues in system identification. In *Proceedings of the 10th IEEE International Conference on Control and Automation*, pages 1–9, Hangzjou, China.

Lohmann, B. and Eid, R. (2009a). Efficient Order Reduction of Parametric and Nonlinear Models by Superposition of Locally Reduced Models. In *Methoden und Anwendungen der Regelungstechnik. Erlangen-Manchener Workshops*, Aachen, Germany.

Lohmann, B. and Eid, R. (2009b). Parametric Model Reduction by Krylov Subspace Methods. In *Workshop GAMM-FA Dynamik und Regelungstheorie*, München, Germany.

Lovera, M., Verhaegen, M., and Chou, C. (1998). State space identification of MIMO linear parameter varying models. In *Proceedings of the International Symposium of the Mathematical Theory of Networks and Systems*, Padova, Italy.

Magruder, C., Beattie, C. A., and Gugercin, S. (2010). Rational Krylov methods for optimal $\mathcal{L}_2$ model reduction. In *Proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, pages 6797–6802, Atlanta, Georgia, USA.

Marmorat, J.-P. and Olivi, M. (2002). The RARL2 software: Realizations and rational approximation in $\mathcal{L}_2$-norm. Technical report, Project APICS, INRIA, Sophia Antipolis, France.

Marmorat, J.-P., Olivi, M., Hanzon, B., and Peeters, R. (2002). Matrix rational $\mathcal{H}_2$ approximation: a state-space approach using Schur parameters. In *Proceedings of the IEEE Conference on Decision and Control*, pages 4244–4249.

Martins, N., Lima, L., and Pinto, H. (1996). Computing dominant poles of power system transfer functions. *IEEE Transactions on Power Systems*, 11(1):162–170.

Mayo, A. J. and Antoulas, A. C. (2007). A framework for the solution of the generalized realization problem. *Linear Algebra and its Applications*, 425(2):634–662.

Melchior, S. A., Van Dooren, P., and Gallivan, K. A. (2012). Finite Horizon Approximation of Linear Time-Varying Systems. In *Proceedings of the 16th IFAC Symposium on System Identification*, Brussel, Belgium.

Meyer, C., Broux, G., Prodigue, J., Cantinaud, O., and Poussot-Vassal, C. (2017). Demonstration of innovative vibration control on a Falcon Business Jet. In *Proceedings of the International Forum on Aeroelasticity and Structural Dynamics*, Como, Italy.

Meyer, C., Prodigue, J., Broux, G., Cantinaud, O., and Poussot-Vassal, C. (2016). Ground test for vibration control demonstrator. In *Proceedings of the 13th International Conference on Motion and Vibration Control*, pages 1–12, Southampton, United Kingdom.

Meyer, D. and Srinivasan, S. (1996). Balancing and model reduction for second-order form linear systems. *IEEE Transactions on Automatic Control*, 41(11):1632–1644.

Michiels, W. and Niculescu, S. (2007). *Stability and Stabilization of Time-Delay Systems*. SIAM.

Michiels, W. and Niculescu, S. (2014). *Stability, Control, and Computation for Time-Delay Systems*. SIAM Advances in Design and Control.

Moore, B. (1981). Principal Component Analysis in Linear Systems: Controllability, Observability and Model Reduction. *IEEE Transaction on Automatic Control*, 26:17–31.

Moulin, B. and Karpel, M. (2007). Gust Loads Alleviation Using Special Control Surfaces. *Journal of Aircraft*, 44(1):17–25.

nd G. Mercère, D. V. (2014). $\mathcal{H}_\infty$-based LPV model identification from local experiments with a gap metric-based operating point. In *Proceedings of the European Control Conference*, pages 388–393, Strasbourg, France.

Nguyen, H. and Revol, N. (2011). Solving and Certifying the Solution of a Linear System. *Reliable Computing*, 15(2):120–131.

Niculescu, S. (2002). On delay robustness analysis of a simple control algorithm in high-speed networks. *Automatica*, 38(5):885–889.

Olivi, M., Seyfert, F., and Marmorat, J.-P. (2013). Identification of microwave filters by analytic and rational $\langle_2$ approximation. *Automatica*, 49(2):317–325.

Ossmann, D. and Poussot-Vassal, C. (2019). Design and assessment of a two degree of freedom gust load alleviation system. In *to appear in Euro GNC*, Milan, Italy.

P. Gahinet, P. A. (2013). Automated tuning of gain-scheduled control systems. In *Proceedings of the Conference on Decision and Control*, pages 2740–2745, Florence, Italy.

Panzer, H., Hubele, J., Eid, R., and Lohmann, B. (2009). Generating a Parametric Finite Element Model of a 3D Cantilever Timoshenko Beam Using Matlab (Vol. TRAC-4, Nov. 2009). Technical report, Technical Reports on Automatic Control.

Partington, J. (1997). *Interpolation, identification and sampling*. Oxford University Press.

Partington, J. (2004). *Linear operators and linear systems: an analytical approach to control theory*, volume 60. Cambridge University Press.

Peherstorfer, B. and Willcox, K. (2018). Dynamic data-driven reduced-order models. *Computer Methods in Applied Mechanics and Engineering*, 291:21–41.

Petersson, D. (2013). *A nonlinear optimization approach to $\mathcal{H}_2$-optimal modeling and control*. PhD thesis, Linkoping University.

Petersson, D. and Lofberg, J. (2014). Model reduction using a frequency-limited $\mathcal{H}_2$-cost. *System & Control Letters*, 67(1):32 – 39.

Pilbauer, D., Bresh-Pietri, D., Meglio, F. D., Prieur, C., and Vyhlidal, T. (2018). Input shaping for infinite dimensional systems with application on oil well drilling. In *Proceedings of the European Control Conference*, pages 1183–1188, Limassol, Cyprus.

Pontes, I. (2017). *Large-scale and infinite dimensional dynamical model approximation*. Ph.D. thesis, Onera, ISAE, Toulouse University, Toulouse, France.

Pontes Duff, I., Gugercin, S., Beattie, C., Poussot-Vassal, C., and Seren, C. (2016a). $\mathcal{H}_2$-optimality conditions for time-delay reduced systems of order one. In *Proceedings of the 13th IFAC Workshop on Time Delay Systems*, pages 7–12, Istanbul, Turkey.

Pontes Duff, I., Poussot-Vassal, C., and Seren, C. (2015a). Realization independent single time-delay dynamical model interpolation and $\mathcal{H}_2$-optimal approximation. In *Proceedings of the 54th IEEE Conference on Decision and Control*, pages 4662–4667, Osaka, Japan.

Pontes Duff, I., Poussot-Vassal, C., and Seren, C. (2018). Optimal $\mathcal{H}_2$ model approximation based on multiple input/output delays systems. *Systems & Control Letters*, 117:60–67.

Pontes Duff, I., Vuillemin, P., Poussot-Vassal, C., Briat, C., and Seren, C. (2015b). Approximation of stability regions for large-scale time-delay systems using model reduction techniques. In *Proceedings of the 14th European Control Conference*, pages 356–361, Linz, Austria.

Pontes Duff, I., Vuillemin, P., Poussot-Vassal, C., Briat, C., and Seren, C. (2016b). *Control-oriented modelling and identification: theory and practice*, chapter Chapter - Model reduction for norm approximation: An application to large-scale time-delay systems, pages 37–55. Advances in Dynamics and Delays. Springer.

Pontes Duff, I., Vuillemin, P., Poussot-Vassal, C., Seren, C., and Briat, C. (2015c). Stability and Performance Analysis of a Large-Scale Anti-Vibration Control Subject to Delays Using Model Reduction Techniques. In *Proceedings of the EuroGNC*, Toulouse, France.

Poussot-Vassal, C. (2011). An Iterative SVD-Tangential Interpolation Method for Medium-Scale MIMO Systems Approximation with Application on Flexible Aircraft. In *Proceedings of the 50th IEEE Conference on Decision and Control - European Control Conference*, pages 7117–7122, Orlando, Florida, USA.

Poussot-Vassal, C., Loquen, T., Vuillemin, P., Cantinaud, O., and Lacoste, J.-P. (2013). Business Jet Large-Scale Model Approximation and Vibration Control. In *Proceedings of the 11th IFAC ALCOSP*, pages 199–204, Caen, France.

Poussot-Vassal, C., Quero, D., and Vuillemin, P. (2018). Data-driven approximation of a high fidelity gust-oriented flexible aircraft dynamical model. In *Proceedings of the IFAC Mathematical Modelling*, Vienna, Austria.

Poussot-Vassal, C. and Roos, C. (2012). Generation of a Reduced-Order LPV/LFT Model from a Set of Large-Scale MIMO LTI Flexible Aircraft Models. *Control Engineering Practice*, 20(9):919–930.

Poussot-Vassal, C., Roos, C., Loquen, T., Vuillemin, P., Cantinaud, O., and Lacoste, J.-P. (2015). *Control-oriented modelling and identification: theory and practice*, chapter 11 - Control-oriented Aeroelastic BizJet Low-order LFT modeling, pages 241–268. The Institution of Engineering and Technology (M. Lovera eds.).

Poussot-Vassal, C. and Sipp, D. (2015). Parametric reduced order dynamical model construction of a fluid flow control problem. In *Proceedings of the 1st IFAC Workshop on Linear Parameter Varying Systems*, pages 133–138, Grenoble, France.

Poussot-Vassal, C. and Vuillemin, P. (2019). Input-output stability estimation of $\mathcal{L}_2(\imath\mathbb{R})$ functions. *submitted*.

Previdi, F. and Lovera, M. (2003). Identification of a class of nonlinear, parametrically varying models'. *International Journal of Adaptive Control and Signal Processing*, 17(1):33–50.

Prokhorov, V. (1994). Rational approximation of analytic functions. *Russian Academia of Science Sb. Math*, 78(1):139–164.

Quero, D. (2017). *An Aeroelastic Reduced Order Model for Dynamic Response Prediction to Gust Encounters*. PhD thesis, Technical University of Berlin.

Reis, T. and Selig, T. (2014). Balancing transformations for infinite-dimensional systems with nuclear Hankel operator. *Integral Equations and Operator Theory*, 79(1):67–105.

Remmert, R. (1991). *Theory of complex functions*, volume 122. Springer.

Richard, J. P. (2003). Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39(10):1667–1694.

Rommes, J. (2008). Arnoldi and Jacobi-Davidson methods for generalized eigenvalue problems $Ax = \lambda Bx$ with singular B. *Mathematics of Computation*, 77:995–1015.

Rommes, J. and Martin, N. (2006). Efficient computation of transfer function dominant poles using subspace acceleration. *IEEE Transactions on Power Systems*, 21(3):1218–1226.

Rommes, J. and Martins, N. (2008). Computing transfer function dominant poles of large-scale second-order dynamical systems. *SIAM Journal on Scientific Computing*, 30(4):2137–2157.

Rommes, J., Martins, N., and Freitas, F. (2010). Computing Rightmost Eigenvalues for Small-Signal Stability Assessment of Large-Scale Power Systems. *IEEE Transactions on Power Systems*, 25(2):929–938.

Rommes, J. and Sleijpen, G. (2008). Convergence of the dominant pole algorithm and Rayleigh quotient iteration. *SIAM Journal on Matrix Analysis and Applications*, 30(1):346–363.

Ruhe, A. (1984). Rational Krylov algorithms for nonsymmetric eigenvalue problems II: matrix pairs. *Linear Algebra and its Applications*, 197:357–385.

Saad, Y. (2000). *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia.

Saad, Y. and Schultz, M. (1986). GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal of Scientific and Statistical Computing*, 7(3):856–869.

Safonov, M. G. and Chiang, R. Y. (1989). A Schur Method for Balanced Model Reduction. *IEEE Transaction on Automatic Control*, 32(7):729–733.

Sandberg, H. (2006). A case study in model reduction of linear time-varying systems. *Automatica*, 42(3):467–472.

Sandberg, H. and Murray, R. (2008). Model reduction of interconnected linear systems. *Optimal Control Applications and Methods*, 30(3):225–245.

Scarciotti, G. and Astolfi, A. (2014). Model reduction by moment matching for linear time-delay systems. In *Proceedings of the 19th IFAC World Congress*, pages 9462–9467, Cape Town, South Africa.

Scherer, C., Gahinet, P., and Chilali, M. (1997). Multiobjective Output-Feedback Control via LMI Optimization. *IEEE Transaction on Automatic Control*, 42(7):896–911.

Scherpen, J. (1993). Balancing for nonlinear systems. *System & Control Letters*, 21(2):143–153.

Schilders, W., der Vorst, H. A. V., and J.Rommes (2008). *Model order reduction: theory, research aspects and applications*. Springer.

Schreiber, K. (2008). *Nonlinear Eigenvalue Problems: Newton-type Methods and Nonlinear Rayleigh Functionals*. PhD thesis, TU Berlin.

Schulze, P. and Unger, B. (2016). Data-driven interpolation of dynamical systems with delay. *System & Control Letters*, 97:125–131.

Schulze, P., Unger, B., Beattie, C., and Gugercin, S. (2018). Data-driven Structured Realization. *Linear Algebra and Its Applications*, 537:250–286.

Seuret, A. (2017). *Contributions to the stability analysis and control of Networked Systems*. HDR. thesis, Université Toulouse 3 Paul Sabatier.

Seuret, A. and Gouaisbaut, F. (2015). Hierarchy of LMI conditions for the stability analysis of time-delay systems. *System & Control Letters*, 81:1–7.

Shaabani-Ardali, L., Sipp, D., and Lesshafft, L. (2017). Time-delayed feedback technique for suppressing instabilities in time-periodic flow. *Physical Revue Fluids*, 2.

Shank, S., Simoncini, V., and Szyld, D. (2016). Efficient low-rank solutions of Generalized Lyapunov equations. *Numerische Mathematik*, 134(3):327–342.

Siahaan, H. B. (2008). A balancing approach to model reduction of polynomial nonlinear systems. *Proceedings of the 17th IFAC World Congress*, 41(2):3269 – 3273.

Simoncini, V. (2007). A new iterative method for solving large-scale Lyapunov matrix equations. *SIAM Journal of Scientific Computation*, 29(3):1268–1288.

Simoncini, V. (2010). Extended Krylov Subspace for Parameter Dependent Systems. *Applied Numerical Mathematics*, 60(5):550–560.

Sipahi, R., Niculescu, S., Abdallah, C., Michiels, W., and Gu, K. (2011). Stability and stabilization of systems with time delay. *IEEE Control Systems Magazine*, 2:38–65.

Sleijpen, G. and Van der Vorst, H. (1996). A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 17(2):401–425.

Sonneveld, P. and Van Gijzen, M. (2008). IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations. *SIAM Journal of Scientific Computing*, 31(2):1035–1062.

Sootla, A. (2014). Nu-gap model reduction in the frequency domain. *IEEE Transactions on Automatic Control*, 59(1):228–233.

Sootla, A., Rantzer, A., and Kotsalis, G. (2009). Multivariable optimization-based model reduction. *IEEE Transactions on Automatic Control*, 54(10):2477–2480.

Stewart, G. (2000). The decompositional approach to matrix computation. *Computing in science engineering*, 2(1):50–59.

Theis, J., Takarics, B., Pfifer, H., Balas, G., and Werner, H. (2015). Modal matching for lpv model reduction of aeroservoelastic vehicles. In *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, Kissimmee, Florida.

Therapos, C. (1989). Balancing transformations for unstable non minimal linear systems. *IEEE Transaction on Automatic Control*, 34(4):455–457.

Tomljanovic, Z., Beattie, C., and Gugercin, S. (2018). Damping optimization of parameter dependent mechanical systems by rational interpolation. *Advances in Computational Mathematics*.

Toth, R. (2010). *Modeling and identification of linear parameter-varying systems*. Lecture Notes in Control and Information Sciences, Vol. 403, Springer, Heidelberg.

Toth, R., Felici, F., Heuberger, P. S. C., and der Hof, P. M. J. V. (2008). Crutial aspects of zero-order hold LPV state-space system discretization. In *Proceedings of the 17th IFAC World Congress (WC)*, Seoul, South Korea.

Van der Vorst, H. (2010). *Computational Methods for Large Eigenvalue Problems, vol. VIII*. Ciarlet, J.L. Lions.

Van Dooren, P., Gallivan, K. A., and Absil, P. A. (2008). $\mathcal{H}_2$-optimal model reduction of MIMO systems. *Applied Mathematics Letters*, 21(12):53–62.

Van Dooren, P., Gallivan, K. A., and Absil, P. A. (2010). $\mathcal{H}_2$-optimal model reduction with higher order poles. *SIAM Journal on Matrix Analysis and Applications*, 31(5):2738–2753.

Varga, A. (1999). Selection of software for controller reduction. SLICOT working note 1999-18. *WGS*.

Villemagne, C. D. and Skelton, R. E. (1987). Model Reduction using a Projection Formulation. *International Journal Control*, 40:2141–2169.

Vuillemin, P. (2014). *Frequency-limited approximation of large-scale LTI dynamical models*. Ph.D. thesis, Onera, ISAE, Toulouse University, Toulouse, France.

Vuillemin, P., Demourant, F., Biannic, J.-M., and Poussot-Vassal, C. (2017). Stability analysis of a set of uncertain large-scale dynamical models with saturations: application to an aircraft system. *IEEE transactions in Control Systems Technology*, 25(2):661–668.

Vuillemin, P. and Poussot-Vassal, C. (2019). Discretisation of continuous-time linear dynamical model with the Loewner interpolation framework. *submitted*.

Vuillemin, P., Poussot-Vassal, C., and Alazard, D. (2013a). A Spectral Expression for the Frequency-Limited $\mathcal{H}_2$-norm. *arXiv:1211.1858*.

Vuillemin, P., Poussot-Vassal, C., and Alazard, D. (2013b). $\mathcal{H}_2$ optimal and frequency limited approximation methods for large-scale LTI dynamical systems. In *Proceedings of the 6th IFAC Symposium on Systems Structure and Control*, pages 719–724, Grenoble, France.

Vuillemin, P., Poussot-Vassal, C., and Alazard, D. (2014a). Poles Residues Descent Algorithm for Optimal Frequency-Limited $\mathcal{H}_2$ Model Approximation. In *Proceedings of the 13th European Control Conference*, pages 1080–1085, Strasbourg, France.

Vuillemin, P., Poussot-Vassal, C., and Alazard, D. (2014b). Spectral expression for the Frequency-Limited $\mathcal{H}_2$-norm of LTI Dynamical Systems with High Order Poles. In *Proceedings of the 13th European Control Conference*, pages 55–60, Strasbourg, France.

Vuillemin, P., Poussot-Vassal, C., and Alazard, D. (2014c). Two upper bounds on the $\mathcal{H}_\infty$-norm of LTI dynamical systems. In *Proceedings of the 19th IFAC World Congress*, pages 5562–5567, Cape Town, South Africa.

Vuillemin, P., Poussot-Vassal, C., and Alazard, D. (2019). Frequency-limited model approximation through limited $\mathcal{L}_2$-norm minimisation. *submitted*.

Wang, Z. and Chen, P. (2017). Accurate rational function approximation for time-domain gust analysis. 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA SciTech Forum, Grapevine, Texas, USA.

Willcox, K. and Megretski, A. (2005). Fourier series for accurate, stable, reduced-order models in large-scale linear applications. *SIAM Journal on Scientific Computing*, 26(3):944–962.

Willcox, K., Peraire, J., and White, J. (2002). An Arnoldi approach for generation of reduced-order models for turbomachinery. *Journal of Computers & Fluids*, 31(3):369–389.

Wilson, D. A. (1974). Model reduction for multivariable systems. *International Journal of Control*, 20(1):57–64.

Xu, Y. and Zeng, T. (2010). Optimal $\mathcal{H}_2$ Model Reduction for Large Scale MIMO Systems via Tangential Interpolation. *International Journal of Numerical Analysis*, 8(1):174–188.

Yin, Q. and Lu, L. (2006). An Implicitly Restarted Block Arnoldi Method in a Vector-Wise Fashion. *Numerical Mathematics*, 15(3):268–277.

Yousouff, A. and Skelton, R. E. (1985). Covariance equivalent realization with application to model reduction of large-scale systems. *Control and Dynamic Systems*, 22:273–348.

Yousouff, A., Wagie, D., and Skelton, R. E. (1985). Linear system approximation via covariance equivalent realizations. *Journal of mathematical analysis and applications*, 106(1):91–115.

Zhou, K. and Doyle, J. C. (1997). *Essentials Of Robust Control*. Prentice Hall.

Zhou, K., Salomon, G., and Wu, E. (1999). Balanced realization and model reduction for unstable systems. *International Journal of Robust and Nonlinear Control*, 9(3):183–198.