THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

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

Présentée et soutenue le *14 Novembre 2025* **par :**

Mathieu CANCES

Reduced order models for the study and control of thermoacoustic instabilities in aeronautic gas turbines.

JURY

Nicolas Noiray	Prof. ETH Zürich	Rapporteur
Matthew Juniper	Prof. University of Cambridge	Rapporteur
Claire Bourquard	Prof. TU/e	Examinatrice
Thierry Schuller	Prof. IMFT	Examinateur
Franck Nicoud	Prof. Université de Montpellier	Directeur de thèse
Laurent Gicquel	Prof. CERFACS	Co-encadrant de thèse
Yoann Mery	Safran Aircraft Engines	Invité

École doctorale et spécialité :
MEGEP : Énergétique et transferts
Unité de Recherche :
CERFACS
Directeur(s) de Thèse :
Franck Nicoud et *Laurent Gicquel*
Rapporteurs :
Nicolas Noiray et *Matthew Juniper*

Keywords : Combustion instabilities, State-space, Reduced-order model, Modal expansion.

Abstract — Thermoacoustic instabilities, caused by the coupling between unsteady heat release and acoustic waves, are a major challenge for the safe and efficient operation of gas turbines. Since these self-sustained oscillations can lead to severe structural damage, developing fast and reliable simulation tools is crucial.

This thesis presents the extension of STORM (State-space Thermoacoustic Reduced Order Model), a reduced-order solver originally developed by C. Laurent, and designed to study thermoacoustic instabilities in realistic gas turbine configurations. STORM is based on a modal decomposition of the acoustic field and a state-space formulation that couples acoustics, flame response (via Flame Transfer Functions), and boundary conditions.

This work builds on a comprehensive understanding of existing modeling strategies, emphasizing the synergy between experimental and numerical approaches. In this thesis, the STORM methodology has been significantly strengthened: boundary conditions are handled more robustly, and a major numerical conditioning issue—responsible for spurious non-physical modes—has been resolved. Moreover, a key contribution is a new formulation for modeling jump conditions at interfaces, where the discontinuity is treated as a source term directly within the Helmholtz equation. This approach avoids domain splitting and the associated numerical instabilities, and has been successfully validated on academic 1D test cases.

The last chapter illustrates the application of STORM to several experimental and industrially relevant configurations.

- MIRADAS, an academic burner from IMFT, serves as a benchmark for validation, with experimental Flame Transfer Functions and outlet impedance data. STORM accurately reproduces the observed instability.

- Safran post-combustor: STORM is used for a parametric study by varying FTF parameters. Around 100 full thermoacoustic computations ($\sim$ 1000 modes on a 2-million-node mesh) were completed in under 2 hours. With conventional finite element solvers, a single mode typically takes $\sim$ 30 minutes, making this scale of analysis practically impossible. STORM thus enables the construction of complete multi-mode stability maps within feasible time.

- HYLON, an academic case with a two-velocity inlet flame, shows STORM's

flexibility. While frequencies match experiments well, growth rates are less accurate, likely due to missing injector impedance data or limitations in the double-input FTF model.

- MICCA, a 16-injector annular combustor (EM2C), was also modeled. Thanks to STORM's flexibility, injector impedances have been incorporated via transfer matrix models, an essential factor to accurately predict the thermoacoustic modes. STORM successfully reproduced the experimental parametric study involving two injector types arranged in various configurations, correctly capturing the evolution of thermoacoustic modes with injector placement.

In conclusion, this work delivers a complete and efficient tool for thermoacoustic stability analysis based on reduced-order modeling. Its main achievements include a robust and modular solver applicable to complex geometries, direct integration of experimental flame and impedance data, the ability to conduct fast parametric studies and design analyses, and validation on several academic and industrial configurations.

Future work will focus on applying STORM to increasingly complex systems to identify its current limitations and guide future developments. The aim is to progressively broaden the range of configurations the solver can handle.

Mots clés : Instabilités de combustion, State-Space, Modèle d'ordre réduit, Expansion modales.

Résumé — Les instabilités thermoacoustiques, causées par le couplage entre le dégagement de chaleur instationnaire et les ondes acoustiques, représentent un défi majeur pour le fonctionnement sûr et efficace des turbines à gaz. Ces oscillations autoentretenues pouvant entraîner des dommages structurels importants, le développement d'outils de simulation rapides et fiables est crucial.

Cette thèse présente l'extension de STORM (State-space Thermoacoustic Reduced Order Model), un solveur bas ordre initialement développé par C. Laurent, conçu pour étudier les instabilités thermoacoustiques dans des configurations réalistes de turbines à gaz. STORM repose sur une décomposition modale du champ acoustique et une formulation en state-space qui couple l'acoustique, la réponse de la flamme (via des Fonction de Transfert de Flamme) et les conditions aux limites.

Ce travail s'appuie sur une compréhension approfondie des stratégies de modélisation existantes, mettant en avant la synergie entre approches expérimentales et numériques. Dans cette thèse, la méthodologie STORM a été significativement renforcée : les conditions aux limites sont désormais traitées de manière plus robuste, et un problème majeur de conditionnement numérique — responsable de modes non physiques parasites — a été résolu. De plus, une contribution clé est une nouvelle formulation pour modéliser les conditions de saut aux interfaces, où la discontinuité est traitée comme un terme source directement dans l'équation de Helmholtz. Cette approche évite la subdivision du domaine et les instabilités numériques associées, et a été validée avec succès sur des cas académiques 1D.

Le dernier chapitre illustre l'application de STORM sur plusieurs configurations expérimentales et industrielles pertinentes.

- MIRADAS, un brûleur académique de l'IMFT, sert de référence pour la validation, avec des Fonction de Transfert de Flamme expérimentales et des données d'impédance en sortie. STORM reproduit avec précision l'instabilité observée.

- Chambre de post combustion Safran : STORM est utilisé pour une étude paramétrique en faisant varier les paramètres des FTFs. Environ 100 calculs thermoacoustiques complets ($\sim$ 1000 modes sur un maillage de 2 millions de nœuds) ont été réalisés en moins de 2 heures. Avec les solveurs classiques éléments finis, un mode unique prend typ-

iquement $\sim$ 30 minutes, rendant ce type d'analyse pratiquement impossible. STORM permet donc de construire des cartes complètes de stabilité multi-modes en un temps raisonnable.

- HYLON, un cas académique avec une flamme à deux entrées de vitesse, illustre la flexibilité de STORM. Si les fréquences correspondent bien aux expériences, les taux de croissance sont moins précis, probablement en raison de l'absence de données d'impédance injecteur ou des limites du modèle FTF à double entrée.
- MICCA, une chambre annulaire à 16 injecteurs (EM2C), a également été modélisée. Grâce à la flexibilité de STORM, les impédances des injecteurs ont été intégrées via des matrices de transfert, un facteur essentiel pour prédire précisément les modes thermoacoustiques. STORM a reproduit avec succès l'étude paramétrique expérimentale impliquant deux types d'injecteurs disposés selon diverses configurations, capturant correctement l'évolution des modes thermoacoustiques en fonction du positionnement des injecteurs.

En conclusion, ce travail fournit un outil complet et efficace pour l'analyse de stabilité thermoacoustique basé sur un modèle d'ordre réduit. Ses principales réalisations comprennent un solveur robuste et modulaire applicable à des géométries complexes, l'intégration directe de données expérimentales de flamme et d'impédance, la capacité à réaliser des études paramétriques rapides et des analyses de conception, ainsi que la validation sur plusieurs configurations académiques et industrielles.

Les travaux futurs viseront à appliquer STORM à des systèmes de plus en plus complexes afin d'identifier ses limites actuelles et orienter les développements à venir. L'objectif est d'élargir progressivement la gamme de configurations que le solveur peut traiter.

Contents

Nomenclature

Physical variables

$\boldsymbol{\phi}(\boldsymbol{x})$	Famiy of pure acoustic eigenmodes	$-$
$\boldsymbol{u'}$	Fluctuating velocity field	$m \cdot s^{-1}$
γ	Heat capacity ratio	$-$
$\omega'_T(\boldsymbol{x}, t)$	Heat release source	$W \cdot m^{-3}$
$\overline{Q}$	Mean flame power	W
$\overline{u}$	Mean velocity	$m \cdot s^{-1}$
$\phi_n(\boldsymbol{x})$	Pure acoustic eigenmode n	$-$
ρ	Density	$kg \cdot m^{-3}$
A^+, A^-	Forward/backward Riemann invariants of the fluctuating pressure	Pa
c	Sound speed	m/s
M_a	Dimensionless Mach number	$-$
p'	Fluctuating pressure field	Pa
$Q'(t)$	Fluctuating heat release rate	W
T	Temperature field	K
u_x, u_y, u_z	Velocity basis in cartesian coordinate system	$m \cdot s^{-1}$

Mathematical symbols

$\boldsymbol{\nabla}$	Gradient operator	m^{-1}

$\boldsymbol{A_i}$, $\boldsymbol{B_i}$, $\boldsymbol{C_i}$, $\boldsymbol{D_i}$	State-space matrices associated with subdomain i	–
$\boldsymbol{X_i}$, $\boldsymbol{U_i}$, $\boldsymbol{Y_i}$	State-space vectors associated with subdomain i	–
δ_{ij}	Kroenecker symbol	–
$\hat{f}$	Fourier transform of the function f	–
$\langle f \vert g \rangle$	Inner product functions f and g.	–
∇_n	Gradient operator projected onto the outward normal vector of a surface	m^{-1}
f	Analytical function f	–
f^T	Transposed of the function f	–
f_h	Numerical approximation of the function f	–

Geometrical symbols

Ω_i	Subdomain i	–

Relative to a mesh

$\underline{\underline{\boldsymbol{M_i}}}^{\boldsymbol{b}}$	Mass matrix associated with the boundary mesh $\mathcal{M}_i^b$	–
$\underline{\underline{\boldsymbol{M_i}}}$	Mass matrix associated with the mesh $\mathcal{M}_i$ (size $N_v \times N_v$)	–
$\underline{\boldsymbol{f}}$	Array of the values of f on each vertex of the mesh (size $N_v \times 1$)	–
$\underline{\boldsymbol{f}}^T$	Numerical transposed of the function f (size $1 \times N_v$)	–
$\mathcal{M}_i$	Mesh i	–
$\mathcal{M}_i^b$	A given boundary of the mesh $\mathcal{M}_i$	–
N_v	Number of vertices	–

Acronyms / Abbreviations

$(L)PDE$	(Linear) Partial Differential Equation	–
BCs	Boundary Conditions	–
DNS	Direct Numerical Simulation	–
FEM	Finite Element Method	–

FTF	Flame Transfer Function	–
HPC	High Performance Computing	–
IMFT	Institut de Mecanique des Fluides de Toulouse	–
LES	Large Eddy Simulation	–
LOM	Low Order Model	–
MEM	Modal Expansion Method	–
MIMO	Multiple Input Multiple Output	–
MISO	Multiple Input Single Output	–
MPI	Message Passing Interface	–
PBF	Pole Base Function	–
RANS	Reynolds Averaged Navier-Stokes	–
ROM	Reduced Order Model	–
RRQR	Rank Revealing QR	–
SAF	Sustainable Aviation Fuel	–
SLEPC	Scalable Library for Eigenvalue Problem Computations	–
SVD	Singular Value Decomposition	–

Chapter 1

General introduction

Contents

1.1 Context

Why should we seek to make aircraft engines more environmentally friendly when reducing the number of flights could directly lower emissions? This is primarily a political and societal question. From a scientific perspective, the issue can be reformulated as follows: how can aircraft engines be made more eco-friendly ?

The aerospace industry is striving to achieve the ambitious objective of net-zero carbon emissions by 2050, as promoted by international organizations such as the International Air Transport Association (IATA) . Achieving this target will require a combination of complementary levers rather than a single technological breakthrough. These include improvements in aircraft and engine efficiency, operational and air traffic management optimizations, as well as the development of alternative energy carriers and fuels. In the short to medium term, Sustainable Aviation Fuels (SAF) are widely regarded as a key contributor to emission reductions, as they can be integrated into existing propulsion systems with limited modifications. In parallel, hydrogen-based propulsion concepts are being investigated as a potential long-term solution, although they still face significant technological, infrastructural, and certification challenges. Institutions such as the European Union Aviation Safety Agency (EASA) and industrial actors like Airbus emphasize the need for sustained research efforts to assess the feasibility and maturity of these emerging technologies. This multi-faceted strategy is reflected in initiatives such as the Destination 2050 initiative , which outlines the European aviation sector's roadmap toward net-zero emissions.

Figure 1.1 shows that the two main development pathways to achieve the Clean Aviation goal according to Airbus are Sustainable Aviation Fuels and new aircraft technologies.

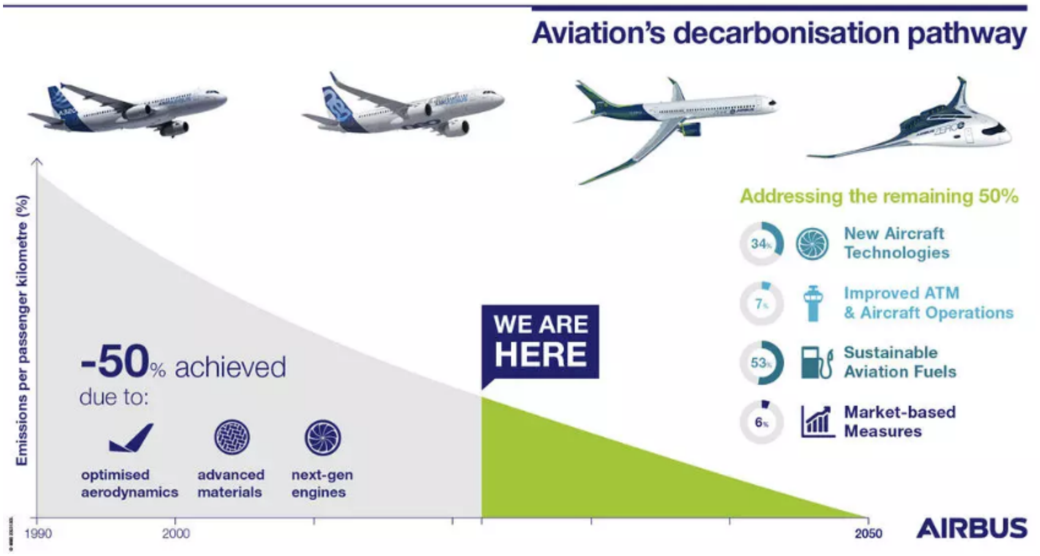


Figure 1.1: Aviation's decarbonisation pathway by Airbus.

Hydrogen is particularly appealing due to its clean combustion, primarily producing water vapor as a byproduct. However, when burned in air, it can also generate nitrogen oxides (NOx). Additionally, its unique physical and chemical properties, such as high flame speed and wide flammability limits, introduce significant challenges in terms of combustion stability and safety. On the other hand, SAF, derived from renewable sources like biomass or waste oils, can serve as a drop-in replacement for conventional jet fuel, allowing the use of existing engine technologies while reducing greenhouse gas emissions. The integration of these fuels into aviation represents a transformative step toward sustainability.

Despite the focus on alternative fuels, the continued study of traditional kerosene-fueled engines remains essential. Indeed kerosene will remain the dominant fuel in aviation for the foreseeable future, especially given the logistical and technological hurdles associated with the large-scale deployment of hydrogen and SAF. Ensuring that conventional engines are optimized for performance, emissions reduction, and reliability is therefore crucial to bridge the gap during this transitional period. Moreover, studying kerosene engines provides valuable insights into the fundamental combustion processes, many of which are also relevant to alternative fuels.

1.2 Combustion instabilities

Regardless of the fuel used —hydrogen, SAF, or kerosene— all combustion systems are susceptible to thermoacoustic instabilities. These instabilities, driven by the coupling between heat release and acoustic waves, pose significant challenges for the design and operation of propulsion systems. They can lead to self-sustained oscillations, structural damage (see Fig. 1.2), and decreased efficiency, making their study a critical priority for both current and future combustion technologies. Understanding and mitigating these instabilities is key to ensuring the reliability and safety of engines across all fuel types.

Figure 1.2: Burner nozzle without (left) and with (right) damage due to combustion instabilities. From [1].

Thermoacoustic instabilities in combustion systems arise from a complex feedback loop between unsteady heat release and acoustic pressure oscillations. When these two phenomena are in phase, the energy from the heat release amplifies the pressure waves, creating a positive feedback mechanism that can lead to self-sustaining oscillations. This coupling is governed by the Rayleigh criterion [2], which states that instabilities grow when the time and volume-integrated product of pressure and heat release rate fluctuations is positive.

Several physical mechanisms contribute to the onset of thermoacoustic instabilities:

- Flame Dynamics: The unsteady behavior of the flame is a primary driver of instabilities. Variations in the incoming air-fuel mixture (due to turbulence or flow instabilities) or fluctuations in the combustion process itself can lead to oscillations in the heat release rate. This unsteady heat release interacts with the acoustic field within the combustor [3–6]. Any disturbances in the mixing process, such as flow separation or vortex shedding, can lead to oscillations in heat release and, consequently, thermoacoustic instabilities.

- Acoustic Resonances: Combustion chambers and connected components, such as ducts and nozzles, exhibit natural acoustic modes. When the system is forced at or near one of these resonant frequencies, acoustic energy is retained within the chamber, allowing pressure fluctuations to accumulate over time [7–9]. However, for these fluctuations to actually drive thermoacoustic instabilities, the heat-release rate must be properly phased with the local pressure and temperature variations, as stated by the Rayleigh criterion [10].
- Boundary Interactions: The interactions between acoustic waves and boundaries, such as walls, nozzles, or flame holders, can either damp or amplify instabilities [11–14]. For example, reflecting waves at open or closed boundaries can reinforce oscillations, depending on the phase relationship of the pressure around the flame with the heat release.
- Nonlinearities and Saturation: As oscillation amplitudes grow, nonlinear effects in the combustion process or acoustic field often emerge, leading to complex behaviors such as limit cycles or chaotic dynamics [15–19]. Nonlinear acoustic effects are generally negligible in gas-turbine engines and become significant primarily in systems such as rocket engines.

These mechanisms are influenced by various factors, including combustor geometry, operating conditions, and the type of fuel used (e.g., kerosene, SAF, or hydrogen). The interplay between these factors is illustrated in

Fig. 1.3, and creates a highly coupled, multi-physics problem that is challenging to predict and control. A deep understanding of these mechanisms is desirable for designing stable combustion systems, whether for conventional fuels or emerging alternatives like hydrogen and SAF. By characterizing how instabilities arise and propagate, researchers can develop strategies to mitigate their impact and ensure safe, efficient, and reliable operation.

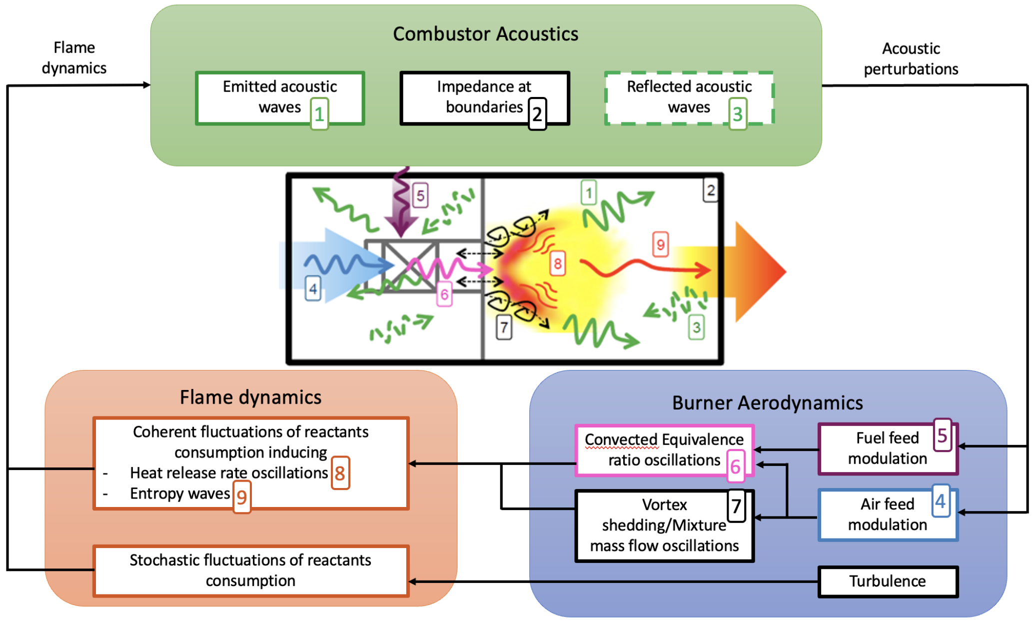


Figure 1.3: Thermoacoustic coupling in a combustion chamber. Adapted from ETH Zurich website (CAPS).

The study of thermoacoustic instabilities requires a multidisciplinary approach that combines theory, experimentation, and advanced computational techniques. Each method provides unique insights into the mechanisms driving instabilities and helps to develop strategies to mitigate their effects.

- **Laboratory Experiments**. Controlled experiments in laboratory settings are a cornerstone of research in thermoacoustic instabilities. These experiments often use simplified combustion setups that replicate key physical phenomena while allowing for precise measurement and control [20–22]. Techniques such as high-speed imaging, laser diagnostics (e.g., Particle Image Velocimetry or PIV) [23], and pressure transducers [24] are employed to study the flame dynamics and oscillations, the acoustic wave propagation or the interaction between flames and combustor walls. Laboratory studies are particularly useful for isolating specific mechanisms, testing hypotheses, and validating numerical models [25]. They provide a safe and cost-effective environment to explore extreme conditions that might be difficult or dangerous to test in real-world engines.
- **Numerical Simulations**. The advent of high-performance computing has revolutionized the study of combustion instabilities through numerical simulations. Techniques such as Computational Fluid Dynamics (CFD) allow researchers to model the complex interactions between fluid dynamics, heat release, and acoustics. The Large Eddy Simulations (LES) is widely used for capturing the detailed, turbulent flow structures that influence combustion dynamics [26, 27]. It provides high fidelity but is computationally expensive. Such simulations are computationally expensive, especially when fine resolution is required

across the entire domain. To mitigate these costs, Adaptive Mesh Refinement (AMR) techniques can be used. AMR dynamically refines the computational grid in regions where higher resolution is needed (e.g., near flame fronts or strong acoustic gradients), while coarsening it elsewhere. This allows for more efficient simulations by preserving accuracy [28, 29].

To complement high-fidelity simulations, reduced-order models (ROMs) offer faster predictions by simplifying the governing equations [30]. ROMs are particularly useful for exploring parameter spaces and designing active control strategies.

- **Theoretical Modeling**. Theoretical approaches play a foundational role in understanding thermoacoustic instabilities. Using simplified equations derived from fluid dynamics, acoustics, and combustion theory, researchers can identify the key parameters governing system behavior [30, 31]. Classical models, such as the Rayleigh criterion or linear stability analysis [32–34], help explain how fluctuations in heat release interact with acoustic waves. These models often lead to the development of analytical or semi-analytical tools [35–37] that provide physical insights and guide both experimental design and numerical simulation. Although idealized, theoretical models are essential for interpreting complex results and identifying dominant instability mechanisms.

- **Engine Testing by Manufacturers**. Large-scale testing in industrial facilities is essential for evaluating the performance and stability of full-scale engines under real operating conditions [38–40]. Companies like SAFRAN conduct extensive engine tests, simulating various operational scenarios to identify potential instabilities and ensure reliability. These tests often involve running engines at different power levels to identify critical resonance frequencies, measuring emissions, pressure oscillations, and structural responses or evaluating the impact of alternative fuels like SAF or hydrogen. While costly, these tests provide invaluable data that cannot be obtained from simplified setups or simulations. They ensure that research findings translate effectively into practical, operational systems.

These main traditional approaches are increasingly supplemented by alternative techniques that maintain prediction capabilities while reducing computational costs. In particular, there is a growing interest in hybrid modeling strategies, which combine high-fidelity simulations with low-order models or empirical data. For example, coupling Large Eddy Simulations (LES) with acoustic solvers [41–44] has proven effective in predicting thermoacoustic instabilities in complex geometries by leveraging both detailed flow dynamics and efficient acoustic analysis.

In parallel, AI-based models are also being more widely explored to predict instabilities by learning from large experimental and simulation datasets. These approaches include Bayesian inference methods [45], classification algorithms [46], machine learning techniques [47], and neural networks [48].

The combination of experimental studies, industrial engine testing, and numerical simulations provides a robust framework for studying thermoacoustic instabilities. While high-fidelity simulations offer detailed insights, their computational cost remains a major limitation for industrial design applications. In response to this challenge, this thesis focuses on the development of a low-order numerical tool that maintains predictive accuracy while significantly reducing computational costs, enabling efficient parametric analyses and design iterations.

1.3 Thesis objectives

This thesis, funded by SAFRAN, aims to pursue the development of a software tool to calculate thermoacoustic instabilities in combustion engines. The focus of the project is on enhancing a low-order method based on the Galerkin method, specifically using a technique known as *frame modal expansion*. This approach was adapted for thermoacoustics by Laurent *et al.* in 2019 [49]. It particularly improves the definition of boundary conditions and jump conditions, which were previously poorly handled in the original Galerkin method. An overview of the different numerical methods that exist to compute thermoacoustic instabilities is provided in Chapter 2.

The *frame modal expansion* method is implemented in the code STORM (STate-space low-ORder Model). It is based on a network approach, where the combustion chamber is decomposed into different subdomains, each characterized by its own dynamics. To connect these subdomains, the state-space formalism is used [50], allowing the definition of both temporal evolution and input/output relations for each subdomain. Thanks to these choices, STORM offers the advantage of enabling easy parametric studies, such as varying boundary conditions or flame transfer functions, in a flexible manner. However, the main challenge with this method is that it generates non-physical modes in the output solution.

In Chapter 3, it is shown that these spurious modes can originate from different sources. The first source is related to the frame modal expansion method itself, which requires the inversion of an ill-conditioned Gram matrix to compute the final modes. This poor conditioning is one of the main causes of the generation of spurious modes. The second source lies in the modeling of the transfer functions used to represent various system components (such as impedance or flame response). To incorporate such frequency-dependent behavior into a low-order model, these transfer functions must be approximated using rational functions, which also introduce non-physical modes into the results. A detailed analysis of these types of spurious mode, along with a strategy to remove them from the output, is provided in Chapter 3.

Since the code is intended for application to geometrically complex configurations, it is essential to accurately model the system's boundary conditions. These boundaries, which are 2D surfaces that can be arbitrarily curved, introduce significant numerical challenges and are another source of non-physical modes in the output. Chapter 4 is dedicated to the study of these boundary conditions, although some aspects remain not fully understood. Moreover, an alternative approach is proposed in this chapter in which boundary conditions are treated as source terms in the Helmholtz equation. This new formulation may offer a way to overcome numerical issues related to the initial implementation of boundary conditions. It is tested and validated on 1D configurations, but has not yet been extended to 3D cases.

Following these steps, the method is applied to real configurations (post-combustion application, experimental test rigs), with the aim of demonstrating its potential for improving the prediction and control of thermoacoustic instabilities in more complex industrial settings (Chapter 5).

Ultimately, the STORM tool is intended to be used by engineers at SAFRAN, as well as by future PhD students working on thermoacoustic instabilities, either directly or indirectly. A remaining objective of this thesis was hence to implement a user-friendly Application Programming Interface (API), ensuring that STORM can be easily adopted by both researchers and engineers. This facilitates the integration of the tool into ongoing research and development efforts, and ensures that the outcomes and advancements achieved throughout this work contribute to the long-term progress in the field of combustion instability research. All of these aspects are detailed in the Annex D, which describes the API implementation and how STORM has been designed for accessibility and ease of use. Additional annexes also provide practical information regarding numerical treatments (Annex C) and state-space realizations (Annex A).

The work presented in Chapter 3 led to the publication of the following article, which was presented at the 2023 Symposium on Thermoacoustics in Combustion (SoTiC), held in Zürich:

M. Cances et al. "*Robustness and reliability of state-space, frame-based modeling for thermoacoustics*", JCP, 2025, vol. 520, p. 113472.

Chapter 2

Modeling combustion instabilities

Contents

2.1 Existing methods

The most complete mathematical description of combustion instabilities relies on the compressible, reactive, multi-species Navier–Stokes equations. Solving these equations typically requires advanced Computational Fluid Dynamics (CFD) approaches, such as Direct Numerical Simulation (DNS) [51, 52] or Large Eddy Simulation (LES) [53, 54], which offer high fidelity but are computationally expensive. To make the problem more tractable, the governing equations can be simplified by neglecting certain effects (e.g., viscosity, detailed chemistry), reducing them to acoustic wave equations and, ultimately, to the Helmholtz equation. Unlike high-fidelity methods such as DNS or LES, Helmholtz solvers do not resolve the Navier-Stokes equations. Instead, they focus exclusively on the acoustic field, making them a low-order modeling approach that is computationally efficient for thermoacoustic analysis.

Figure 2.1 summarizes different approaches, highlighting how they balance accuracy and computational cost depending on the complexity of the problem.

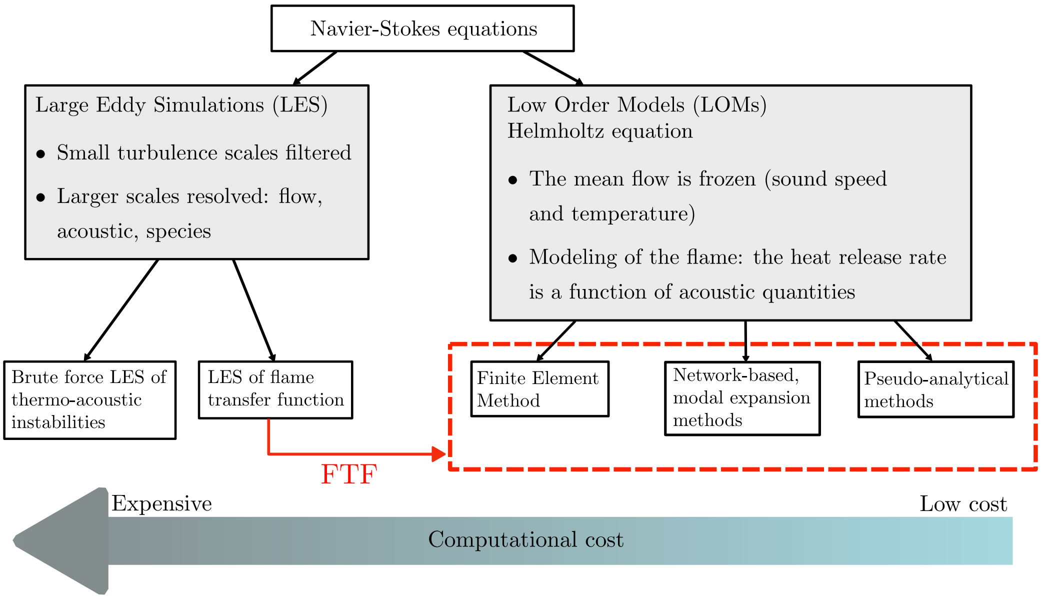


Figure 2.1: Numerical methods for thermoacoustic instabilities. On the left, the LES method. On the right, the class of Low-Order Models, which require a Flame Transfer Function (FTF) as an input. Adapted from Poinsot [30]

- **Direct Numerical Simulation (DNS)** is a high-fidelity computational method used to study turbulent flows and combustion phenomena [51, 52]. Unlike approaches that model turbulence (such as RANS or LES), DNS solves the Navier-Stokes equations without any turbulence model, resolving all relevant spatial and temporal scales —

from the largest flow structures to the smallest dissipative eddies (Kolmogorov scale). This makes DNS particularly powerful for capturing fine-scale physical details.

In combustion, DNS provides detailed insight into complex processes such as flame–turbulence interaction, heat transfer, and chemical kinetics [55]. It also inherently captures the coupling between unsteady heat release and pressure waves, making it a valuable tool for studying thermoacoustic instabilities [56]. Thanks to its high accuracy, DNS is often used to generate reference data for validating reduced-order models and improving the physical understanding of combustion dynamics.

However, DNS comes with a prohibitive computational cost. Because it must resolve all flow scales, the number of degrees of freedom for an industrially relevant case often reaches 10^9 to 10^{10}, far exceeding current computational capabilities. As a result, DNS remains restricted to academic studies or simplified geometries and is not practical for routine industrial design.

- **Large Eddy Simulation (LES)** is a numerical method that serves as an excellent alternative to Direct Numerical Simulation (DNS), particularly for studying turbulent combustion. Unlike DNS, which resolves all spatial and temporal scales of the flow, LES filters the flow field to resolve the larger, energy-containing turbulent eddies while modeling the smaller, subgrid-scale physics using a turbulence model. This makes it well-suited for high-Reynolds-number flows, such as those in gas turbines and aero-engines, where DNS is too costly. In unsteady combustion, LES effectively captures thermoacoustic interactions, offering valuable insights into the coupling between heat release and acoustic modes. As a result, it has become a key tool for studying combustion instabilities in both research and industry, with industrial cases typically involving $\sim 10^7$ to 10^8 degrees of freedom.

This approach significantly reduces the computational cost compared to DNS, making LES more practical for realistic engineering applications [26, 53, 54, 57, 58]. Based on massive parallelism, Wolf *et al.* [59] showed that full annular LES of two helicopter combustors can capture self-established thermo-acoustic modes.

- **Helmholtz solvers** are specialized numerical tools designed to solve the Helmholtz equation (see Eq. 2.3 in the next part), which governs the coupling phenomenon between acoustic waves and the heat release rate of a flame in a cavity.

  To use a Helmholtz solver, two critical inputs are required. First, the mean flow fields must be known, particularly the distribution of the speed of sound $c_0(\boldsymbol{x})$ and density $\rho_0(\boldsymbol{x})$. These mean fields are typically derived from higher-fidelity simulations, such as LES [26] or Reynolds-Averaged Navier-Stokes (RANS) computations [60, 61]. The accuracy of the acoustic predictions depends on the quality of these mean fields, as they define the basis through which the acoustic waves propagate.

  Second, a Flame Transfer Function (FTF) or a Flame Describing Function (FDF) is needed to model the acoustic response of each flame present in the geometry. The FTF modeling was first proposed by Crocco [62], and quantifies how the unsteady heat release rate from the flame responds to acoustic perturbations, capturing the coupling between the flame and the acoustic field. This concept of flame response has been widely used in thermoacoustic studies [63–70], as it is essential for accurately representing the interaction between combustion dynamics and acoustics. The Flame Describing Function (FDF) corresponds to a non-linear extension of the FTF that accounts for amplitude-dependent effects [66, 71, 72]. Instead of a single gain/phase per frequency, the FDF response varies depending on the amplitude of the perturbations. FDF is more accurate for large perturbations, where the flame response deviates from linear behavior.

A major disadvantage of Helmholtz solvers is that they do not account for certain important physical mechanisms of acoustic energy dissipation —such as perforated plates or dilution holes— which must be included as a separate model.

The main advantage of Helmholtz solvers is their efficiency. The mesh resolution required for a Helmholtz solver can be coarser, as it only needs to resolve the acoustic wavelengths, not the smaller flow or flame structures. As a rule of thumb, it typically requires 10 points per wavelength to capture thermoacoustic modes, which leads to a number of DoFs of the order of $\sim 10^5$ or 10^6 in industrial configurations.

Helmholtz solvers are particularly useful for analyzing thermoacoustic instabilities in practical systems, as they enable the rapid prediction of resonant modes and instability frequencies / growth rates. While they lack the detailed physical fidelity of high-order simulations, their reliance on mean fields and flame transfer functions makes them a practical tool for industrial design and optimization, especially for systems where multiple flames and complex geometries are involved.

2.2 Helmholtz equation

Coupling mechanisms between acoustic waves and flames control two important phenomena in the development of modern combustion systems: combustion noise and combustion instabilities. From a numerical point of view, the combustion noise is directly produced by the flame, but there is no feedback loop feeding the flame. In contrast, combustion instabilities result from the constructive coupling between the intrinsic acoustics of a cavity and the unsteady heat release rate of a flame, which generates sound [30].

Combustion instabilities are fully described by the well-known Navier-Stokes equations, but some simplifications are necessary to build a thermoacoustic Low Order Model (LOM). The main assumptions are listed below.

- **H1**: The fluid is a perfect gas.
- **H2**: Viscous forces are neglected, and no viscous stresses are retained in the volume. Consistently, only the normal velocity goes to zero at solid boundaries (i.e slip wall).
- **H3**: The flame is considered a black box with an acoustic input and a fluctuating rate of heat release output, where all the multi-species transport and chemical phenomena are hidden.
- **H4**: The linear acoustics assume that for any flow variable f, its oscillating part f' is very small compared to its mean value f_0. $f(\boldsymbol{x}, t) = f_0(\boldsymbol{x}) + f'(\boldsymbol{x}, t)$, except for the velocity fluctuations (see next point). The baseline temperature, sound speed and density are then static fields $T_0(\boldsymbol{x})$, $c_0(\boldsymbol{x})$ and $\rho_0(\boldsymbol{x})$.
- **H5**: The mean flow speed is assumed to be zero, and the acoustic velocity is small compared to the speed of sound: $u'(\boldsymbol{x}) \ll c_0(\boldsymbol{x})$. In

practice, this hypothesis remains true while the Mach number $M(\boldsymbol{x}) = u_0(\boldsymbol{x})/c_0(\boldsymbol{x})$ is lower than 0.3.

Under these approximations, and using some mathematics that are detailed in [73], one can obtain the spatio-temporal evolution of thermoacoustic fluctuations:

$$\boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} p'\right) - \frac{1}{\gamma p_0} \frac{\partial^2 p'}{\partial t^2} = -\frac{\gamma - 1}{\gamma p_0} \frac{\partial \omega'_T}{\partial t} \tag{2.1}$$

In this equation, p_0 is the homogeneous static pressure, ρ_0 and γ stand for the baseline density and heat capacity ratio, respectively. Moreover, $p' = p'(\boldsymbol{x}, t)$ is the field of pressure fluctuations, while ω'_T is the fluctuating rate of heat release (in $W.m^{-3}$). The right-hand-side term represents the acoustic forcing due to the work done by the gas during unsteady combustion, accounting for both energy added during gas expansion and energy absorbed during gas contraction. When dealing with linear acoustics, it is convenient to introduce $\hat{p}(\boldsymbol{x}, \omega)$, the Fourier transform of the acoustic field $p'(\boldsymbol{x}, t)$:

$$\begin{cases} \hat{p}(\boldsymbol{x}, \omega) = \displaystyle\int_{-\infty}^{+\infty} p'(\boldsymbol{x}, t) e^{-j\omega t} \mathrm{d}t \\ \\ p'(\boldsymbol{x}, t) = \displaystyle\frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{p}(\boldsymbol{x}, \omega) e^{j\omega t} \mathrm{d}\omega \end{cases} \tag{2.2}$$

With this convention, all the acoustic quantities are proportional to $e^{j\omega t}$ Applying this transformation to Eq. 2.1 leads to the frequency domain Helmholtz equation:

$$\boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} \hat{p}\right) + \frac{\omega^2}{\gamma p_0} \hat{p} = -j\omega \frac{\gamma - 1}{\gamma p_0} \hat{\omega_T} \tag{2.3}$$

Finally, boundary conditions (BCs) must be provided to close the problem. The three main BCs are:

- **Rigid wall** This condition assumes zero acoustic velocity normal to the surface $u'_n = 0$. It represents a perfectly reflecting boundary. It corresponds to the Neumann condition for the pressure, that is $\nabla_n p' = 0$.

- **Open atmosphere** This condition imposes zero acoustic pressure $p' = 0$, resulting in total reflection of the acoustic wave with a phase inversion. It is often used to approximate an idealized open end, although real open boundaries exhibit partial transmission and energy losses.

- **Impedance boundary** This more general condition relates pressure and velocity fluctuations through an acoustic impedance $Z(\omega)$. It imposes the dimensionless ratio between pressure and acoustic normal velocity: $Z(\omega) = \hat{p}/\rho_0 c_0 \hat{\boldsymbol{u}} \cdot \boldsymbol{n_s}$. It allows modeling of partially absorbing or reflective surfaces, depending on the impedance value. Some example of the use of impedance boundary conditions can be found in [74–78].

In addition to the standard boundary conditions listed above, more advanced models are sometimes required to accurately capture the complex acoustic behavior at interfaces:

- **Multiperforated Liners (MLPF)** In industrial combustion chambers, perforated plates are used to separate the combustion zone, where the hot gases reside, from the fresh gas flow that cools the chamber through the small holes in the plate. From a thermoacoustic perspective, these plates act as acoustic attenuators and are modeled by a jump term. For instance, the Howe's model [79] relates the acoustic pressure jump across the plate to the temporal variation of the normal velocity upstream of the plate. In the Helmholtz equation, this jump condition is represented by the relation: $\Delta p \propto \partial u'_n/\partial t$. The work of Gullaud *et al.* [80] investigates the impact of multiperforated plates on acoustic modes in combustion chambers within the Finite Element framework. A detailed modeling of these plates using the modal expansion approach is presented in Section 4.4.

- **Acoustic Transfer Matrices (ATM)** can be used to model acoustic elements in industrial combustion chambers, such as injectors. A transfer matrix relates the acoustic variables (pressure and velocity) at the inlet and outlet of a component, providing a compact representation of its acoustic response [81–83]. This approach is particularly useful for characterizing the acoustic behavior of injectors, which often exhibit complex interactions with combustion dynamics. Using measured or computed transfer matrices, engineers can efficiently incorporate these components into thermoacoustic stability analyses, enabling a more accurate prediction of instabilities.

When the flame is modeled using a flame transfer function (detailed in Section 3.7.2), the Helmholtz equation becomes a eigenvalue problem for the acoustic pressure. The complex eigenmodes $(\omega_m, \hat{p}_m)_{m\in\mathbb{N}}$ represent the thermoacoustic modes of the system, characterized by their frequency, growth rate, and mode shape.

2.3 Classes of Helmholtz solvers

Helmholtz solvers can themselves be classified into different categories, each associated with a distinct numerical resolution method.

2.3.1 Pseudo-analytical methods

The pseudo-analytical approach should be considered when dealing with simple academic cases. The system is treated as a network of 1D elements, in which the acoustic pressure is approximated by 1-dimensional planar waves $p'(x,t) = A^+e^{j(kx-\omega t)} + A^-e^{-j(kx+\omega t)}$. This approach allows the dispersion equation to be written analytically in the form $\boldsymbol{A}(\omega)X = 0$. The square matrix A depends on the geometrical dimensions of the system and the angular frequency ω. The eigenvector X contains the unknown amplitudes A^+ and A^-. The solutions ω satisfy the dispersion relation $\det(\boldsymbol{A}(\omega)) = 0$ and are usually computed using a numerical algorithm. Different academic cases use this approach to understand the fundamental mechanisms of thermoacoustic coupling. A canonical example of such a method is the ATACAMAC approach [37], where a one-dimensional model is developed to compute azimuthal thermoacoustic modes in an annular chamber fed by an annular plenum through N burners. In this approach, the acoustic waves in all the different components of the geometry (i.e chamber, plenum and burners) are approximated by 1-dimensional planar waves. This analytical approach allows to categorize the coupling between the chamber and the plenum, depending on the parameters of the flames that are set up in each burner. The pseudo-analytical method is the least computationally expensive approach, but it only allows for 1D calculations. At CERFACS, a pseudo-analytical tool called SoundTube allows to compute the thermoacoustic modes of N duct of surface S_n connected to each other, where a flame can be set in each duct.

2.3.2 Finite Element Method

For more complex 3D geometries, the coupled Helmholtz equation Eq. 2.3 is not reducible to an analytical set of equations. The Finite Element Method (FEM) must be employed to solve it on a discretized mesh of a complex geometry. A mesh is a division of the domain into smaller, non-overlapping subdomains called elements, connected by nodes. Figure 2.2 illustrates the discretization of $N_v \sim 2\times10^6$ of Degree of Freedoms (DoFs) in an aeronautical engine.

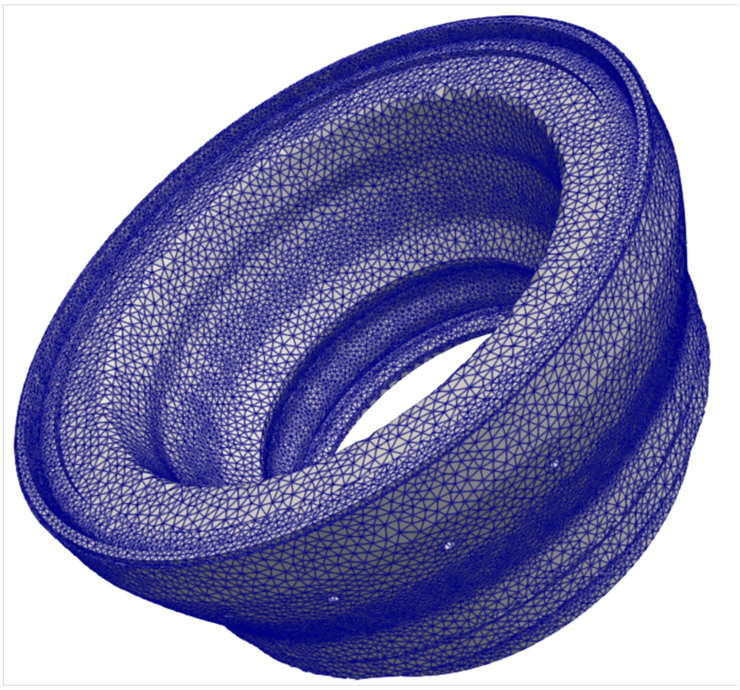


Figure 2.2: Example of a mesh discretization in the annular combustion chamber of the SILVERCREST engine from Safran Aircraft Engines.

Since the Helmholtz equation is typically solved as an eigenvalue problem to determine the natural frequencies and associated mode shapes, the pressure field is separated into a spatial component and a time-harmonic oscillation: $\hat{p}_m(\boldsymbol{x}, \omega_m) = p_m(\boldsymbol{x})e^{j\omega_m t}$. The integer subscript m labels each

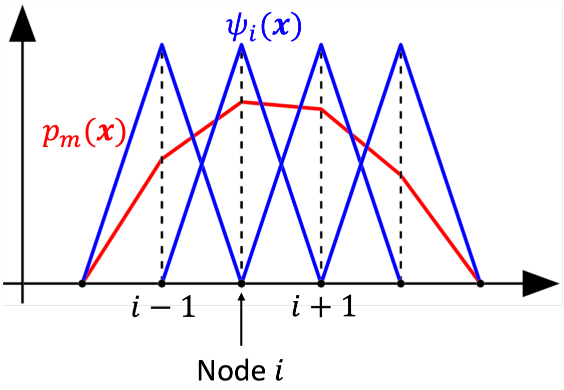


Figure 2.3: Basis functions corresponding to Lagrange elements, in the case of a 1-dimensional mesh.

eigenmode (frequency and shape pair). The unknown shape $p_m(\boldsymbol{x})$ is approximated using basis functions $\psi_i(\boldsymbol{x})$ (see Eq. 2.4), often simple polynomials. These functions are local, meaning they are non-zero only on specific elements. For example, Lagrange basis functions are piecewise linear and designed to be 1 at the node indexed by i, and 0 at all other nodes (see Fig. 2.3).

$$p_m(\boldsymbol{x}) = \sum_{i=1}^{N_v} a_i \psi_i(\boldsymbol{x}) \tag{2.4}$$

The problem is hence to find the N_v unknown amplitude coefficients a_i, called the *nodal values*, and the complex value of ω_m. The finite element formalism results in an eigenvalue system $\boldsymbol{A}(\omega_m)\boldsymbol{V_m} = -\omega_m^2 \boldsymbol{B}(\omega_m)\boldsymbol{V_m}$ where:

- $\boldsymbol{V_m}$ is the vector of unknowns a_i. The total number of unknowns a_i, also called degrees of freedom (DoFs), is proportional to the number of nodes in the mesh. When using Lagrange elements [84], the size of $\boldsymbol{V_m}$ is equal to N_v. Lagrange elements are the most commonly used as they are simply linear functions on the elements, as illustrated in Fig. 2.3.

- ω_m is the complex eigenfrequency in $rad.s^{-1}$ associated with the mode m, and for which the real part (resp. imaginary part) gives the angular frequency (resp. growth rate) of the mode m.

- $\boldsymbol{A}$ and $\boldsymbol{B}$ are the stiffness matrix and mass matrix respectively, which results from the contributions of all the elements of the mesh. Their structure depends on the physics and boundary conditions of the problem. In the case of non-reactive flows with simple boundary conditions (such as Dirichlet or Neumann), $\boldsymbol{A}$ and $\boldsymbol{B}$ are independent of the eigenfrequency ω. The resulting eigenvalue problem is therefore linear and can be solved directly. However, when the system involves more complex conditions —for instance, impedance boundaries, multiperforated liners, or an active flame— the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ become a function of the eigenfrequency ω. This dependence makes the eigenvalue problem nonlinear. To solve it, an iterative approach is used: (1) The user provides an initial guess ω_0 for the eigenfrequency. (2) Using this guess, $\boldsymbol{A}(\omega_0)$ and $\boldsymbol{B}(\omega_0)$ are assembled and the eigenvalue problem is solved. (3) The solution is used to update the estimate of ω. (4) Steps 2–3 are repeated until convergence toward a value ω_m. The algorithm ultimately returns the converged eigenfrequency ω_m corresponding to the thermoacoustic mode closest to the initial guess ω_0. Since the procedure converges toward a single mode per run, it must be repeated with different initial guesses to identify additional modes [74].

The mesh resolution must be chosen carefully to accurately capture the spatial variations of the acoustic pressure field. In particular, the mesh size is typically dictated by the smallest wavelength of the mode of interest. As a general rule of thumb, 10 elements per wavelength are recommended to ensure adequate accuracy. Moreover, for the resulting system to remain well-conditioned, it is important to control the variation in mesh element sizes. The ratio between the volume of the largest and smallest elements in the mesh should not exceed 10^8. Exceeding this threshold may lead to ill-conditioning of the system matrices, resulting in numerical instabilities or convergence issues during the eigenvalue computation. In industrial configurations, due to the geometrical complexity of components such as injectors, it is often necessary to refine the mesh locally to represent small features. This localized refinement leads to small elements, while the rest of the domain remains relatively coarser. As a consequence, the total number of mesh points can reach around $N_v \sim 10^6$.

Although considerably reduced compared to LES, the calculation time of the Finite Element Method applied to the Helmholtz equation is too large to be used in parametric studies, where different physical parameters are prone to be modified, and several modes have to be computed. Note also that there is no guaranty that all the thermoacoustic modes of interest will be found, as is often the case when dealing with non-linear eigenvalue problems. Intrinsic thermoacoustic modes (ITA) are usually missed when using this approach [85, 86]. Buschmann *et al.* [87] proposed an integral method based on Beyn's algorithm that guarantees the identification of all eigenmodes within a specified contour. However, this method is not explored in the present thesis, as it does not address the fundamental challenge in thermoacoustic mode computation: the nonlinear nature of the underlying eigenvalue problem.

2.3.3 Modal expansion methods

For linear partial differential equations (LPDEs) —such as the Helmholtz equation— another appropriate resolution approach is the Modal Expansion

Method (MEM). MEM are used to solve complex differential equations, particularly in dynamic systems [88], fluid mechanics [89], and acoustics [14]. The idea is to search the unknown pressure field $p'(\boldsymbol{x}, t)$ solution of the reactive Helmholtz equation as a linear combination of modes which are solutions of the non-reactive Helmholtz equation. The main steps are then:

- The choice of basis functions (also called modes). The solution to the problem is approximated as a linear combination of modes, which are pre-selected basis functions. For the Helmholtz equation, these modes are the pure acoustic modes of the domain, which are the solution of the non-reactive Helmholtz equation with trivial boundary conditions.
- The projection. The reactive Helmholtz equation is projected onto the space spanned by the chosen modes. This involves multiplying the residual of the equations by each basis function and integrating over the domain. This results in a reduced set of ordinary differential equations (ODEs) for the modal amplitudes.
- The reduced system of ODEs is assembled into a compact eigensystem that captures the dominant thermoacoustic dynamics. Because only a limited number of basis modes are retained, this reduced model can be solved efficiently to determine eigenfrequencies, growth rates, and mode shapes, enabling fast parametric studies and control-oriented analyses.

Contrary to FEM, the acoustic pressure field $p'(\boldsymbol{x}, t)$ is no longer decomposed into a mesh basis functions, but on a modal basis so that:

$$p'(\boldsymbol{x}, t) = \sum_{n=1}^{N_{modes}} p_i(t) f_i(\boldsymbol{x}) \tag{2.5}$$

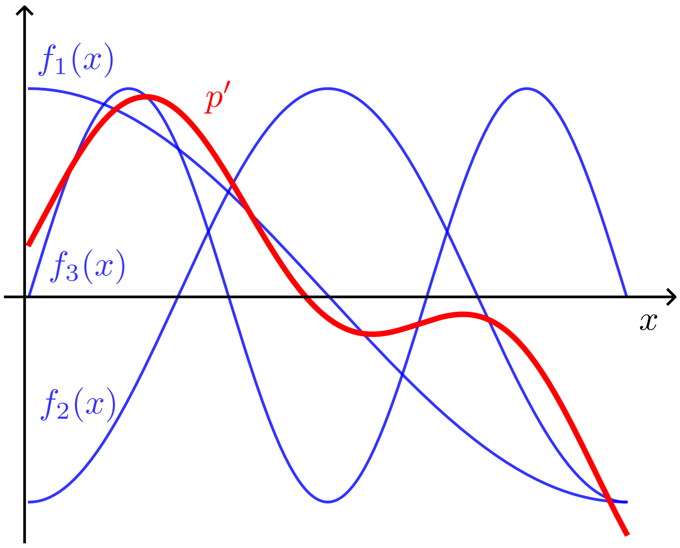


Figure 2.4: Illustration of the modal decomposition in a 1-dimensional duct.

The modal basis is composed by N_{modes} basis functions f_i, defined as the pure acoustic modes of the studied geometry. Fig. 2.4. illustrates this modal decomposition in a 1-dimensional case.

The modal expansion method focuses on a small number of dominant modes N_{modes}, significantly reducing the degrees of freedom in the system. Only a few ordinary differential equations (ODEs) are solved, rather than large systems of algebraic equations, as for the finite element method.

In the context of thermoacoustics, two types of modal expansion will be discussed, the Galerkin modal expansion [90] and the frame modal expansion [49]. The next chapter delves into the details of solving the Helmholtz equation using the modal expansion method and a network modeling. It explores the limitations of the Galerkin modal expansion, particularly in handling boundary conditions, and explains why the frame modal expansion is a more suitable approach. However, this alternative also comes with its own challenges, which will also be discussed in the following chapter.

Chapter 3

Frame modal expansion and state-space network modeling

Contents

The work presented in this chapter has been published in a peer-reviewed journal: M. Cances, L. Giraud, M. Bauerheim, L. Gicquel and F. Nicoud. «Robustness and reliability of state-space, frame-based modeling for thermoacoustics.» *Journal of Computational Physics*. Vol. 520, p. 113472, jan. 2025. doi: 10.1016/j.jcp.2024.113472. This paper was presented at the 2023 Symposium on Thermoacoustics in Combustion (SoTiC), held in Zürich.

3.1 Network representation of thermoacoustic systems

The present work makes use of the state-space formalism and a *divide and conquer* strategy. The idea is first to decompose the thermoacoustic system (e.g. a gas turbine) into a set of subdomains, individually described by their own dynamics. Input and output quantities are also defined for each subdomain, which allows them to be connected to each other. The state-space formalism adapted for connecting subdomains will be detailed later. To illustrate how a combustion chamber can be divided in this way, Fig. 3.1 shows an example of such a network decomposition.

In this sketch, several subdomains are present:

- Ω_1 is a 3D subdomain with hot gases.
- Ω_2 is a 3D subdomain with cold gases.
- H is a 3D subdomain associated with a heat source in the Ω_1 subdomain.
- Multiple 2D subdomains noted C, which represent different connection elements, such as a liner-type pressure jump condition between the two

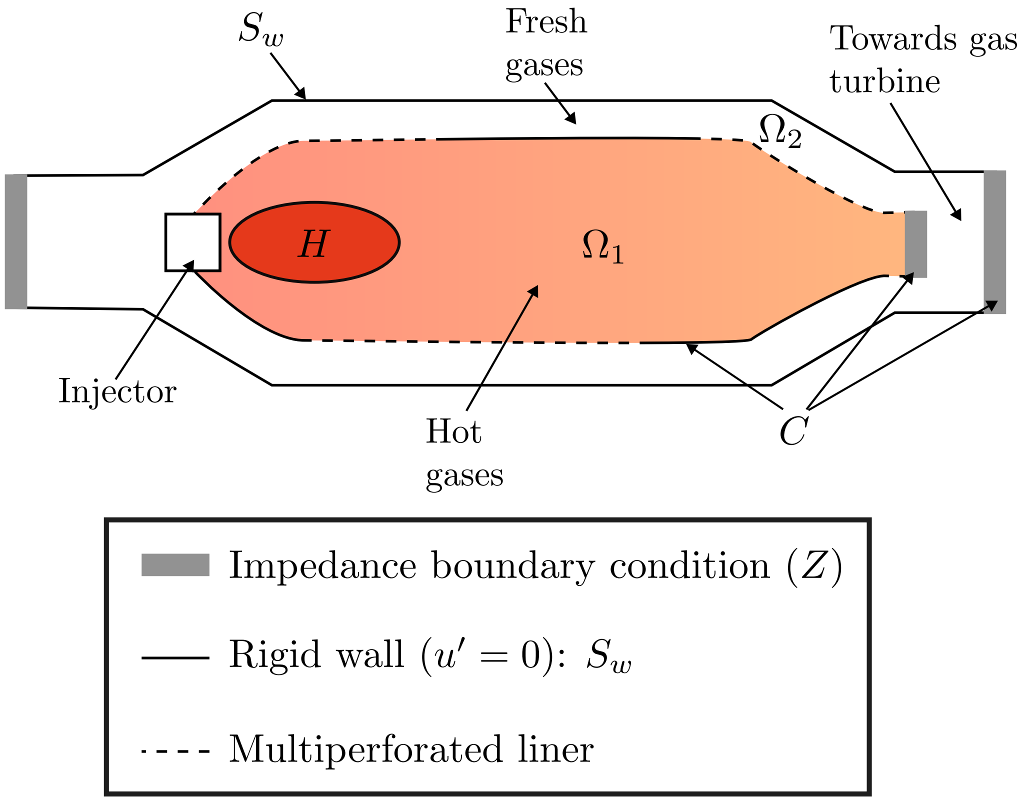


Figure 3.1: Schematic representation of a combustion chamber decomposition from state-space point of view.

geometrical subdomains Ω_1 and Ω_2, or impedance boundary conditions. They are all noted C because they all follow the same formalism specific to 2D surfaces.

All above subdomains can be sorted into 3 different classes.

- Geometrical subdomains, noted Ω, can be defined by their respective pure acoustic modes, obtained as solutions of the Helmholtz equation (Eq. 2.3) without combustion forcing (right-hand-side term set to 0) and associated with open ($\hat{p} = 0$) or closed ($\nabla_n \hat{p} = 0$) boundary conditions, or to a mixture of these two types of conditions. ∇_n stands for the gradient operator projected onto the normal direction of the boundary. Only objects of this class are modeled by the frame modal expansion detailed in Section 3.3.
- Connection subdomains, noted C, are defined by jump relations or boundary conditions. They can represent multiperforated liners, impedance condition or any type of condition applied to an interface or a frontier of a geometrical subdomain.
- Flame subdomains, noted H, and defined by a relation modeling the flame response to the acoustics. For instance, a flame transfer function links the unsteady heat release rate of the flame to the velocity fluctuations upstream of the flame.

3.2 State-space formalism

Once the network decomposition is chosen, the coupling between subdomains remains to be formulated. This operation is done by associating a continuous time-invariant state-space to each subdomain [91]. A state-space for any subdomain (i) is a system of two matrix equations, for which $\boldsymbol{A}_i$, $\boldsymbol{B}_i$ and $\boldsymbol{C}_i$ are time independent:

$$\begin{cases} \dot{\boldsymbol{X}}_i(t) &= \boldsymbol{A}_i\boldsymbol{X}_i(t) + \boldsymbol{B}_i\boldsymbol{U}_i(t) \\ \boldsymbol{Y}_i(t) &= \boldsymbol{C}_i\boldsymbol{X}_i(t) + \boldsymbol{D}_i\boldsymbol{U}_i(t) \end{cases} \tag{3.1}$$

The first line of Eq. 3.1 is the evolution equation. It describes the dynamics of the subdomain (i) under the forcing of other subdomains present in the network. This forcing is the product of the input matrix $\boldsymbol{B}_i$ and the input vector $\boldsymbol{U}_i(t)$. The dynamic variables of interest related to subdomain (i) are stored in the state vector $\boldsymbol{X}_i$. The second line of Eq. 3.1 serves to define an output vector $\boldsymbol{Y}_i$ for the subdomain (i), allowing to assemble it to other subdomains, by connecting their respective inputs/outputs $(U\ /\ Y)$. In the state-spaces presented here, all the feedthrough matrices $\boldsymbol{D}_i$ are zero, so they are omitted from the rest of the text. The Appendix A.1 details how an element of the input vector $\boldsymbol{U}_i$ can be incorporated into the state-vector $\boldsymbol{X}_i$ to simplify the output equation of the state-space representation by removing the feedthrough term $\boldsymbol{D}_i\boldsymbol{U}_i$. The inputs/outputs of each type of subdomains are summarized in Table 3.1, where the acoustic quantities u'_n, φ and p' are respectively the velocity normal to the boundary, the acoustic potential and the pressure measured at the edge of the connected subdomain.

All the individual state-spaces can then be assembled by applying recursively the Redheffer Star Product [92], which consists in relating the input and output vectors of subdomains that are connected. The result of this inter-connection process is a global state-space representing the complete thermoacoustic system, which reads:

$$\dot{\boldsymbol{X}}_g(t) = \boldsymbol{A_g}\boldsymbol{X}_g(t) \tag{3.2}$$

Once this global system is assembled, all physical information present in

Table 3.1: Input/output vectors associated with the three different subdomain classes. Q'_f is the fluctuating heat release rate representing the flame. φ is the acoustic potential defined as $p' = -\rho_0 \partial\varphi/\partial t$. The superscript T is the transpose operator.

Subdomains	Geometrical (Ω)	Flame (H)	Connection (C)
Input vectors $\boldsymbol{U}$	$\boldsymbol{U}_\Omega^T = [u'_n \;\; \varphi \;\; Q'_f]$	$\boldsymbol{U}_H^T = [u'_n \;\; p']$	$\boldsymbol{U}_C^T = [u'_n \;\; p']$
Output vectors $\boldsymbol{Y}$	$\boldsymbol{Y}_\Omega^T = [u'_n \;\; p']$	$\boldsymbol{Y}_H^T = [Q'_f]$	$\boldsymbol{Y}_C^T = [u'_n \;\; \varphi]$

matrices $\boldsymbol{A}_i$, $\boldsymbol{B}_i$ and $\boldsymbol{C}_i$ of each subdomain contribute to the global state matrix $\boldsymbol{A}_g$, while all the input/output vectors $\boldsymbol{U}_i$ and $\boldsymbol{Y}_i$ vanish. The global state vector $\boldsymbol{X}_g$ is merely the concatenation of each subdomain state vector $\boldsymbol{X}_i$. Note that Eq. 3.2 can be solved in time, or the eigen solutions of the dynamic matrix $\boldsymbol{A}_g$ can be computed to obtain the set of thermoacoustic modes of the global system. In the second case, the eigenvalues/eigenvectors of $\boldsymbol{A}_g$ give the pulsations/shapes of all the thermoacoustic modes of the global system. If ω_n is a complex eigenvalue of $\boldsymbol{A}_g$, $\Re(\omega_n)$ and $\Im(\omega_n)$, respectively, give the growth rate and the pulsation of the mode n. The associated eigenvector $\boldsymbol{V}_n$ contains all the information to compute the pressure and velocity fields corresponding to the mode n.

If we note $\boldsymbol{V_n^i} = [a_1, b_1, \cdots, a_{N_i}, b_{N_i}]^T$ the portion of the global eigenvector $\boldsymbol{V_n}$ associated with the geometrical subdomain Ω_i, then the pressure $p^{\Omega_i}(\boldsymbol{x})$ and velocity $\boldsymbol{u}^{\Omega_i}(\boldsymbol{x})$ fields in this subdomain are constructed such that:

$$p^{\Omega_i}(\boldsymbol{x}) = \sum_{n=1}^{N_i} b_n \phi_n^{\Omega_i}(\boldsymbol{x}) \tag{3.3}$$

$$\boldsymbol{u}^{\Omega_i}(\boldsymbol{x}) = -\frac{1}{\rho_0^i(\boldsymbol{x})} \sum_{n=1}^{N_i} a_n \boldsymbol{\nabla} \boldsymbol{\phi}_{\boldsymbol{n}}^{\Omega_i}(\boldsymbol{x}) \tag{3.4}$$

where $(\phi_n^{\Omega_i})_{n \leq N_i}$ is the modal frame associated with subdomain Ω_i and N_i is the size of this basis. These quantities are more detailed in the section 3.3. The density field in the subdomain Ω_i is noted $\rho_0^i(\boldsymbol{x})$.

Note that $\boldsymbol{A}_g$ is a square matrix, typically of size $\sim$ 1000 or much less, as it corresponds to the sum of the size of each connected subdomain ($\boldsymbol{X}_g$ is the concatenation of each individual $\boldsymbol{X}_i$).

3.3 From Galerkin to frame modal expansion

In order to solve Eq. 2.3, the *divide and conquer* strategy defined earlier is used. As illustrated in Fig. 3.1, a global system is decomposed into different subdomains, which individually represent a particular physics. This section is dedicated to any element belonging to the geometrical subdomain class (see Tab. 3.1), which is described by its pure acoustic modes. A classical method for describing an acoustic domain relies on the Galerkin modal expansion, introduced by Morse in 1968 [93]. In that case, the acoustic pressure p' on a subdomain Ω_i is decomposed on a known modal basis $(\phi_n^{\Omega_i})_{n \in \mathbf{N}}$. To simplify the notation, and since we are focusing on a single sub-domain, the index i referring to subdomain Ω_i is omitted, so that:

$$p'(\boldsymbol{x}, t) = \sum_{n}^{\infty} \dot{\Gamma}_n(t) \phi_n(\boldsymbol{x}) \tag{3.5}$$

$\dot{\Gamma}_n$ corresponds to the temporal derivative of the modal amplitude Γ_n, introduced here for convenience in the equations. $\phi_n(\boldsymbol{x})$ are solutions of the Helmholtz equation (Eq. 3.6), without sources and with the Neumann boundary condition, that is $\nabla_n \phi_n = 0$ (where ∇_n is the gradient operator, projected onto the outward normal vector of the surface). Since it is numerically not possible to build a set of infinite number of vectors, an arbitrary value N is usually prescribed to limit the size of the modal basis which becomes $(\phi_n)_{n\leq N}$ and for which,

$$\boldsymbol{\nabla}\cdot\left(\frac{1}{\rho_0}\boldsymbol{\nabla}\phi_n\right)+\frac{\omega_n}{\gamma p_0}\phi_n=0 \tag{3.6}$$

Using the relation $\boldsymbol{\nabla p'}=-\rho_0\partial\boldsymbol{u'}/\partial t$, the modal expansion for velocity fluctuations is also known through its projection onto the modal basis, or more precisely, the gradient of the modal basis:

$$\boldsymbol{u'}(\boldsymbol{x},t)=-\frac{1}{\rho_0(\boldsymbol{x})}\sum_{n=1}^{\infty}\Gamma_n(t)\boldsymbol{\nabla}\boldsymbol{\phi_n}(\boldsymbol{x}) \tag{3.7}$$

The problem is hence reduced to finding the N amplitudes $\Gamma_n(t)$ corresponding to each component $\phi_n(\boldsymbol{x})$ in the subdomain Ω_i. The inherent issue of such a Low-Order Model (LOM) using the Galerkin modal expansion is the proper definition of the boundary conditions of the problem. By construction, the acoustic field is always decomposed onto a basis satisfying a single type of boundary condition (homogeneous Neumann modeling a wall, or Dirichlet modeling an open atmosphere) at the frontier of each subdomain. To further illustrate this issue, let us consider the example of a 1D closed duct connected to an impedance block representing a complex boundary condition (Fig. 3.2) .

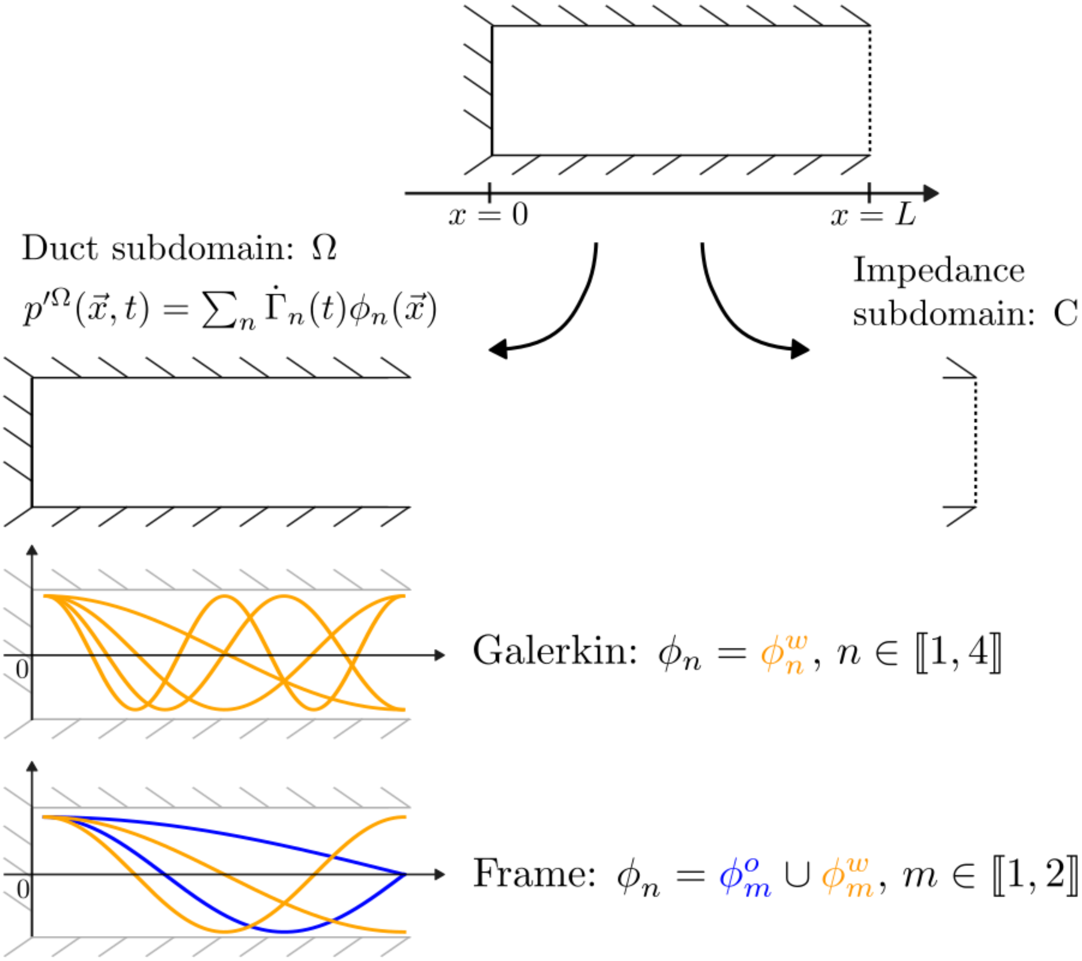


Figure 3.2: State-space decomposition of an academic configuration: A 1-dimensional duct connected to an impedance block defining the boundary condition $p' = \rho_0 c_0 Z u'_n$, where Z is the value of the impedance. The difference between the frame and the Galerkin modal decompositions is also illustrated. In the duct geometrical subdomain, the frame is built as the concatenation of two orthogonal basis ϕ^w and ϕ^o, while the Galerkin basis is only composed by one orthogonal basis ϕ^w.

The network decomposition of this simple academic case is based on two elements:

- One geometrical subdomain Ω representing the 1D duct for which the pressure field is decomposed onto the corresponding basis.
- One connection subdomain C representing the impedance boundary condition $\hat{p} = \rho_0 c_0 Z \hat{u}_n$, where Z is the value of the impedance and $\hat{u}_n$ the fluctuating velocity normal to the surface.

If the Galerkin decomposition is used for the duct, and for instance the modal basis satisfies the Neumann boundary condition at the impedance interface, the gradient of the computed acoustic pressure is necessary 0 because the vectors $\phi_n(\boldsymbol{x})$ constituting the basis are all satisfying $\nabla_n \phi_n = 0$. So, regardless of the modal amplitudes Γ_n, the pressure gradient (and acoustic velocity) will be zero at the impedance location, which is non-physical. The same issue is also present if the basis satisfies Dirichlet boundary condition, which will impose a node for the acoustic pressure field at the impedance location. To remedy this inconsistency, the frame modal decomposition has been proposed and developed for thermoacoustics by Laurent *et al.* [49]. As the value of the pressure field is unknown at the interface, the Galerkin basis is over-completed with another basis, satisfying a different boundary condition at that interface. In the duct, the pressure field is hence decomposed onto a frame, so that:

$$\begin{cases} p'(\boldsymbol{x},t) &= \sum_n^N \dot{\Gamma}_n(t)\phi_n(\boldsymbol{x}) \\ \text{with } (\phi_n)_{n\leq N} &= (\phi_m^o)_{m\leq N/2} \cup (\phi_m^w)_{m\leq N/2} \end{cases} \tag{3.8}$$

Note that if N is not even, the frame can be defined as $(\phi_n)_{n\leq N} = (\phi_m^o)_{m\leq (N+1)/2} \cup (\phi_m^w)_{m\leq (N-1)/2}$ or vice-versa. Note that the set of vectors

$(\phi_n(\boldsymbol{x}))_{n\leq N}$ introduced in Eq. 3.8 is no more an orthogonal basis but a frame, defined as the concatenation of two orthogonal basis $(\phi_n^o(\boldsymbol{x}))_{n\leq N/2}$ and $(\phi_n^w(\boldsymbol{x}))_{n\leq N/2}$ which correspond to the solutions of the Helmholtz equation (Eq. 3.6) with two different types of BCs at the impedance interface: Dirichlet and Neumann (see Fig. 3.2).

Note also that for this simple case, an analytical solution of the system exists but it is not used, the goal is to propose a methodology for representing any geometrical subdomain under the state-space framework. Generally, for 1D, 2D or 3D cases, the frame representation is applied to each geometrical subdomain Ω which has an interface C (see Fig. 3.1) where the behaviour of pressure is non-trivial (neither Neumman nor Dirichlet). These interfaces can be either a jump relation between acoustic quantities as multiperforated liners, or impedances. Using the frame allows the pressure field to be well-defined at the interfaces/boundaries of each geometrical subdomain, contrary to an orthogonal basis for which a Gibbs phenomenon usually appears at these positions (Fig. 3 in [49]). Depending on the boundary condition used to define the Galerkin modal basis, the Gibbs phenomenon appears either on the pressure (Dirichlet BC) or on the velocity fluctuation (Neumann BC) fields.

3.4 Limits of the frame modal expansion

Although frame model expansion eases the treatment of interfaces and BCs, it also comes with its own limitations. To understand these limits, the test case presented in Fig. 3.2 is considered. The configuration consists in a 1-dimensional tube of length $L = 1$ m with zero velocity on one side ($x = 0$) and a finite-valued impedance $Z = 0.5$ at the other boundary ($x = L$). The value of the sound speed is set to $c_0 = 390.3\ m.s^{-1}$. The system is decomposed into two subdomains: the duct Ω and the impedance block C. A state-space (Eq. 3.1) is then associated with each subdomain.

3.4.1 State-space representation of an acoustic geometrical subdomain

The fact that pressure fluctuations decomposed using Eq. 3.8 are also solution of the thermoacoustic problem expressed by Eq. 2.3 must now be translated into the state-space language. Since the unknowns of the problem are the modal amplitude coefficients $\Gamma_n(t)$, these become the state variables of the state-space set of equations. The equation governing the time evolution of each Γ_n can be found by injecting the pressure frame decomposition (Eq. 3.8) into Eq. 2.3. After some algebra [49], the differential equation of the modal amplitudes in the duct hence reads,

$$\left\{\begin{array}{lcl} p'(\boldsymbol{x},t) & = & \sum_n \dot{\Gamma}_n(t)\phi_n(x) \\ \ddot{\Gamma}_n(t) & = & -\omega_n^2\Gamma_n(t) + \rho_0 c_0^2 \boldsymbol{\Lambda}^{-\mathbf{1}}\boldsymbol{\mathcal{I}}_{\phi}(t) + \boldsymbol{\Lambda}^{-\mathbf{1}}\boldsymbol{\mathcal{F}}_{\phi}(t) \end{array}\right. \tag{3.9}$$

The last term on right-hand side of the second line of Eq. 3.9 represents the external forcing contribution to the subdomain due to the impedance block C. This connection is managed by the term $\boldsymbol{\mathcal{I}}_{\phi}(t)$, which involves the projection of the frame $\boldsymbol{\phi}$ onto the connection interface C, the acoustic potential and the normal velocity at this interface. Its explicit expression, which is not useful for the present analysis, is provided in [94]. ρ_0 and c_0 are the mean density and sound speed at the location of the connection. $\boldsymbol{\Lambda}^{-\mathbf{1}}$ is the inverse of the Gram matrix, defined as the inner product between frame components: $\boldsymbol{\Lambda} = \boldsymbol{\phi}^T\boldsymbol{\phi}$ where $\boldsymbol{\phi}$ is the row vector $\boldsymbol{\phi} = [\phi_1, \cdots, \phi_N]$. The term $\boldsymbol{\mathcal{F}}_{\phi}(t)$ represents the volumic source in the case where a flame is present in the system. Its explicit expression is detailed in Section 3.7. Still, $\boldsymbol{\Lambda}^{-\mathbf{1}}$ can be difficult to compute numerically due to poor numerical conditioning.

This issue would arise as soon as a vector ϕ_n of the frame $\boldsymbol{\phi}$ is close to a linear combination of other vectors of the frame: $\phi_n \simeq \sum_{m \neq n} \alpha_m \phi_m$. This situation which is in fact more and more likely when the frame size increases as discussed in the next section.

3.4.2 Conditioning issues

In the case depicted in Fig. 3.2, the explicit expression of the modal frame $\boldsymbol{\phi}$ of the duct is:

$$\begin{aligned}
\boldsymbol{\phi} &= (\phi_n(\boldsymbol{x}))_{n \leq N} \\
\boldsymbol{\phi} &= \underbrace{\left(\cos \frac{m\pi x}{L}\right)_{m \leq N/2}}_{\phi_m^w} \bigcup \underbrace{\left(\cos \frac{(2m+1)\pi x}{2L}\right)_{m \leq N/2}}_{\phi_m^o}
\end{aligned} \tag{3.10}$$

This frame $\boldsymbol{\phi}$ is associated with the following eigen-frequencies:

$$\begin{aligned}
\boldsymbol{f} &= (f_n)_{n \leq N} \\
\boldsymbol{f} &= \underbrace{\left(\frac{mc_0}{2L}\right)_{m \leq N/2}}_{f_m^w} \bigcup \underbrace{\left(\frac{(2m+1)c_0}{4L}\right)_{m \leq N/2}}_{f_m^o}
\end{aligned} \tag{3.11}$$

The maximum value reached by $\boldsymbol{f}$ in Eq. 3.11 is $f_{max} = (N+1)c_0/4L$. It is the upper limit of the computable frequencies for the whole system and so that no mode with a frequency higher than f_{max} can be reconstructed by a linear combination of modes of lower frequencies. This reasoning can be extended to cases which involve several geometrical subdomains. Effectively, the global frequency upper limit is the minimum of all the f_{max} defined for each geometrical subdomains.

Once the frame is defined, the second line of Eq. 3.9 can be written as a state-space representation (Eq. 3.1), where the state vector associated with the duct, $\boldsymbol{X_d}$, is the concatenation of all the unknown modal amplitudes $\Gamma_n(t)$ and their temporal derivative $\dot{\Gamma}_n(t)$. The explicit expression of state-space matrices for a generic geometrical subdomain can be found in Appendix A of [49]. Another state-space is associated with the impedance block C. Once all the state-spaces are defined, the Redheffer star product allows us to assemble a global system of the form $\dot{\boldsymbol{X}}_{\boldsymbol{g}} = \boldsymbol{A_g X_g}$ by connecting respective inputs/outputs of each individual state-space. The schematic in Fig. 3.3 illustrates this specific step.

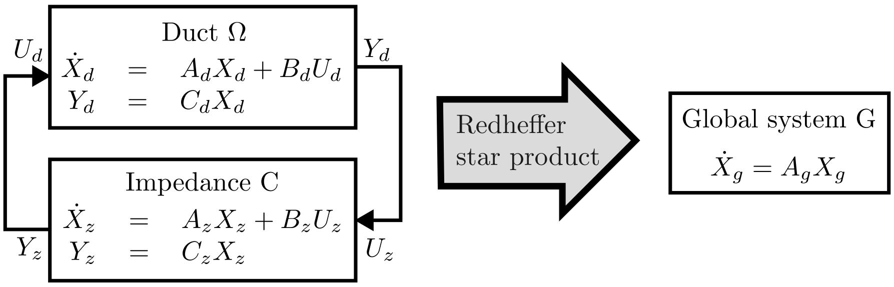


Figure 3.3: Block diagram of the state-space representation of the configuration depicted Fig. 3.2, in which 2 blocks are involved (i.e. 2 subdomains): the duct Ω and the impedance C.

Note that the final assembled equation $\dot{\boldsymbol{X}}_{\boldsymbol{g}} = \boldsymbol{A_g X_g}$ is also a State-space but without any input/output. For this case it reads:

$$\underbrace{\begin{bmatrix} \dot{\boldsymbol{X}}_{\boldsymbol{d}} \\ \dot{\boldsymbol{X}}_{\boldsymbol{z}} \end{bmatrix}}_{\dot{\boldsymbol{X}}_{\boldsymbol{g}}} = \underbrace{\begin{bmatrix} \boldsymbol{A_d} & \boldsymbol{B_d C_z} \\ \boldsymbol{B_z C_d} & \boldsymbol{A_z} \end{bmatrix}}_{\boldsymbol{A_g}} \underbrace{\begin{bmatrix} \boldsymbol{X_d} \\ \boldsymbol{X_z} \end{bmatrix}}_{\boldsymbol{X_g}} \tag{3.12}$$

where the index "d" stands for contributions and variables from the duct while the index "z" refer to variables or contributions associated with the impedance condition. The inverse of the Gram matrix $\boldsymbol{\Lambda^{-1}}$ appearing in the coupling terms of $\boldsymbol{A_g}$ (particularly in $\boldsymbol{B_d}$ and $\boldsymbol{C_d}$, see appendix A in [49]) is computed using the Moore-Penrose pseudo-inverse [95]. The global system (Eq. 3.12) is solved with LAPACK [96], for 3 different values of the frame size, namely $N = 12$, 20 and 26, increasing the range of accessible frequencies (i.e increasing f_{max}). Figure 3.4 shows the frequency and the growth rate of the modes between 0 and f_{max} returned by the frame expansion method for each frame size. They are compared to the analytical complex frequencies $s_n^A = f_n^A + i\sigma_n^A$, solutions of the dispersion relation associated with the configuration, given by:

$$iZ \tan\left(\frac{2\pi s_n^A}{c_0} L\right) = 1 \tag{3.13}$$

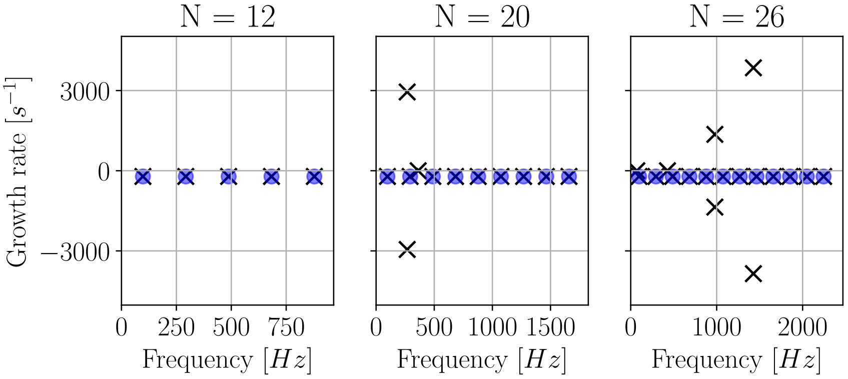


Figure 3.4: Modes of the configuration depicted in Fig. 3.2 displayed in the frequency plane. Crosses: Numerical results from the frame modal expansion method, with different values of the frame size: $N = 12$ (left), $N = 20$ (middle), $N = 26$ (right). Blue circles: analytical solutions (Eq. 3.13). Only modes with frequency between 0 and $f_{max} = (N + 1)c_0/4L$ are plotted.

The agreement is very good for $N = 12$ since all the five first analytical modes are well represented by the frame-based numerical procedure, both for the oscillation frequency and for the growth rate. The agreement is still good for $N = 20$ and $N = 26$ but some spurious modes appear (see the × symbols that do not match the reference circles for $N = 20$ and for $N = 26$). The generation of these non-physical components is due to numerical rounding errors in the inversion process of the Gram matrix $\mathbf{\Lambda}$. As the inverse of the Gram matrix is involved in the computation of modal amplitudes Γ_n (see Eq. 3.9), numerical errors appearing while estimating $\mathbf{\Lambda}^{-1}$ propagate, resulting in some spurious modes in the output of the method. To support this claim, Figure 3.5 displays the singular values $s(\mathbf{\Lambda})$ of the Gram matrix for each frame size $N = 12$, 20 and 26. The condition number of the matrix, $\kappa(\mathbf{\Lambda})$, allows quantifying the conditioning of the geometrical subdomain (the duct), and is defined as the ratio between the maximum and the minimum of singular values:

$$\kappa(\mathbf{\Lambda}) = \frac{\max s(\mathbf{\Lambda})}{\min s(\mathbf{\Lambda})} \tag{3.14}$$

The presence of spurious modes is expected if $1/\kappa(\mathbf{\Lambda})$ approaches the machine precision, about 10^{-16} for double precision coding. From Figure 3.5, this condition is met for $N = 20$ and 26 for which cases a plateau-like behaviour is present for the lowest singular values (around 10^{-16}), which is responsible for the poor numerical conditioning of the system. However this plateau is not reached for $N = 12$, in other words, $\mathbf{\Lambda^{-1}}$ is computed accurately for $N = 12$ only, which is consistent with the results in Fig. 3.4.

The frame size N must be chosen based on the highest frequency of interest in the system, denoted f_{up}. For a given N, the frame can only represent frequencies up to a limit f_{max}. To ensure that all relevant thermoacoustic modes are captured, we must select N such that: $f_{max}(N) > f_{up}$. In the example of Fig. 3.2, N is deliberately increased only to illustrate the numerical ill-conditioning that can arise for large frame sizes. In practical cases where high-frequency modes (such as screech) are of interest, a large N is unavoidable to reach the target frequency f_{up}, and ill-conditioning issues can occur.

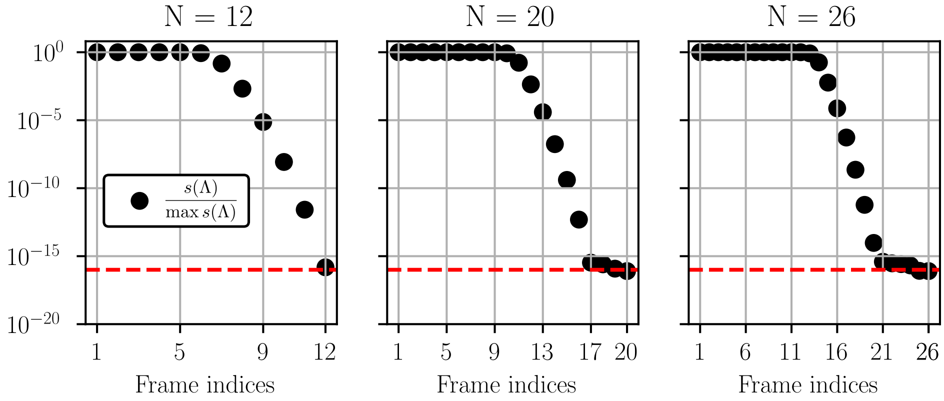


Figure 3.5: Singular values distribution of the Gram matrix normalized by the maximum of singular values, for $N = 12$ (left), $N = 20$ (middle) and $N = 26$ (right). The red horizontal dashed lines correspond to 10^{-16}, an indication of the machine precision.

3.5 Singular Value Decomposition of the frame

To overcome these ill-conditioning issues even for large values of N, the proposed methodology is first to transform the frame into an orthonormal basis using the Singular Value Decomposition (SVD) approach, and then truncate the frame by removing all vectors associated with the smallest singular values.

3.5.1 SVD of the frame

In each geometrical subdomain Ω_i (i.e only the duct in the case studied), the SVD decomposition is applied to the frame $\boldsymbol{\phi}$ as follows. For convenience, the subscript i is omitted for all provided expressions.

$$\boldsymbol{\phi} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{3.15}$$

where $\boldsymbol{U}$ is a matrix of columns vector $\boldsymbol{U} = [U_1(\boldsymbol{x}) \ \cdots \ U_N(\boldsymbol{x})]$ of the same shape as $\boldsymbol{\phi}$ and which contains a set of orthonormal column vectors spanning the same space as $\boldsymbol{\phi}$. $\boldsymbol{\Sigma}$ is the diagonal matrix of singular values, sorted in descending order. The unitary matrix $\boldsymbol{V}$ contains the right singular vectors of $\boldsymbol{\phi}$. Doing this SVD requires a metric adaptation detailed in Appendix C.1. Once this decomposition is done, the new state-space based on the $\boldsymbol{U}$ basis has to be derived. For this purpose, the unknown modal amplitudes are now denoted $\dot{\Upsilon}_n$ (see Eq. 3.16). Recall that the index i referring to any geometrical subdomain Ω_i is omitted so the pressure fluctuations is decomposed onto the $\boldsymbol{U}$ basis instead of the frame $\boldsymbol{\phi}$ such that,

$$p'(\boldsymbol{x},t) = \sum_{n=1}^{N} \dot{\Upsilon}_n(t) U_n(\boldsymbol{x}) \tag{3.16}$$

To proceed with this new basis, the corresponding Helmholtz equation for which solutions are the components of $\boldsymbol{U}$ needs to be obtained. This is achieved by remarking that the classical Helmholtz equation can be written in the following vector form:

$$\boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} \boldsymbol{\phi} \right) + \frac{1}{\gamma p_0} \boldsymbol{\phi} \boldsymbol{\Omega}^{\mathbf{2}} = \mathbf{0} \tag{3.17}$$

where $\boldsymbol{\Omega}$ is the diagonal of eigenfrequencies, $\boldsymbol{\Omega} = diag(\omega_1, \cdots, \omega_N)$. By substituting $\boldsymbol{\phi} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ and then right-multiplying the equation by $(\boldsymbol{\Sigma}\boldsymbol{V}^T)^{-1} = \boldsymbol{V}\boldsymbol{\Sigma}^{-1}$, the following modified Helmholtz equation is obtained:

$$\boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} \boldsymbol{U} \right) + \frac{1}{\gamma p_0} \boldsymbol{U} \left[\boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{\Omega}^{\mathbf{2}} \boldsymbol{V} \boldsymbol{\Sigma}^{-1} \right] = \mathbf{0} \tag{3.18}$$

For which a coupling between U_n components reads:

$$\boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} U_n(\boldsymbol{x}) \right) + \frac{1}{\gamma p_0} \sum_{k=1}^{N} [\boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{\Omega}^2 \boldsymbol{V} \boldsymbol{\Sigma}^{-1}]_{k,n} U_k(\boldsymbol{x}) = 0 \tag{3.19}$$

No boundary condition needs to be defined for this equation since it is never solved in practice; instead the $\boldsymbol{U}$ basis is computed thanks to the orthonormalization process $\boldsymbol{\phi} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$. This equation does not represent any physical phenomenon, but is needed to express the new state-space of geometrical subdomains. Note that there are similarities between this modified Helmholtz equation for U_n and the classical Helmholtz equation for ϕ_n. For example, the set of equivalent pulsations $[\boldsymbol{\Sigma V}^T\boldsymbol{\Omega}^2\boldsymbol{V\Sigma}^{-1}]_{k,n}$, with $k \in [\![1, N]\!]$ can be associated with the mode U_n just like ω_n^2 is associated with the mode ϕ_n. The new state-space of any geometrical subdomain Ω_i can hence be expressed from Eq. 3.19 and is written:

$$\begin{cases} p'(\boldsymbol{x}, t) & = & \sum_n^N \dot{\Upsilon}_n(t) U_n(x) \\ \ddot{\Upsilon}_n(t) & = & -\sum_k^N [\boldsymbol{\Sigma V}^T\boldsymbol{\Omega}^2\boldsymbol{V\Sigma}^{-1}]_{n,k}\Upsilon_k(t) \\ & + & \rho_0 c_0^2 \boldsymbol{\mathcal{I}_U}(t) \\ & + & \boldsymbol{\mathcal{F}_U}(t) \end{cases} \tag{3.20}$$

The state-space realization of this system is given in Appendix A.2. Some similarities can be highlighted between the modified state-space (Eq. 3.20) and the original one (Eq. 3.9). Terms $\boldsymbol{\mathcal{I}_U}(t)$ and $\boldsymbol{\mathcal{F}_U}(t)$ are the counterpart of the $\boldsymbol{\Lambda}^{-1}\boldsymbol{\mathcal{I}}_\phi$ and $\boldsymbol{\Lambda^{-1}\mathcal{F}}_\phi(t)$ of Eq. 3.9 and respectively represent the external forcing contributions from the other subdomains and the volumic source induced by the flame. Note that the $\boldsymbol{\Lambda}^{-1}$ dependency vanishes since $\boldsymbol{U}$ is an orthogonal basis, and the projection terms $\boldsymbol{\mathcal{I}_U}$ and $\boldsymbol{\mathcal{F}_U}$ are now depending on U_n vectors instead of ϕ_n. The numerical illposedness of the SVD-based state-space (Eq. 3.20) is related to the conditioning of $\boldsymbol{\Sigma}$ (see the term $\boldsymbol{\Sigma}^{-1}$ in Eq. 3.20) instead of $\boldsymbol{\Lambda}$ for the original state-space formulation (Eq. 3.9). Since $\boldsymbol{\Lambda} = \boldsymbol{V\Sigma}^2\boldsymbol{V}^T$, the singular values of $\boldsymbol{\Sigma}$ are the square root of those of $\boldsymbol{\Lambda}$, so $\kappa(\boldsymbol{\Sigma}) = \sqrt{\kappa(\boldsymbol{\Lambda})}$. Thanks to this specific relation, the state-space of Eq. 3.20 should be better conditioned than the one using the original frame modal expansion. Still, one can expect that numerical quality issues will remain for cases where $\kappa(\boldsymbol{\Sigma}) \geq 10^{16}$, or equivalently $\kappa(\boldsymbol{\Lambda}) \geq 10^{32}$. To illustrate this point, the configuration described in Fig. 3.2 is solved using the SVD-based state-space (Eq. 3.20) for the same values of frame size ($N = 12$, 20 and 26) as in Figs. 3.4 and 3.5. The results in terms of frequencies are presented in top row of Fig. 3.6.

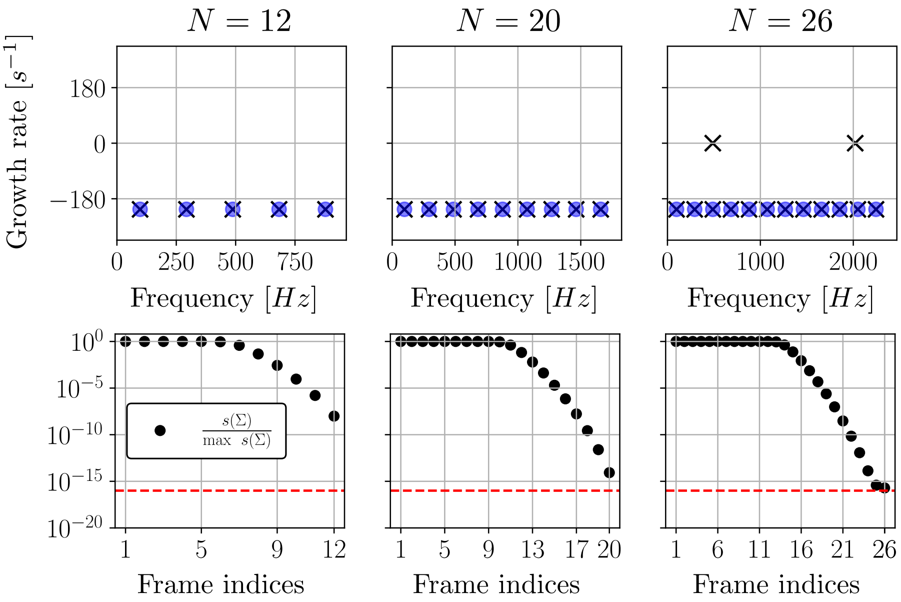


Figure 3.6: Top row: Modes of the configuration depicted in Fig. 3.2 displayed in the frequency plane. Crosses: Numerical results from the SVD-based modal expansion method, with different values of the frame size: $N = 12$ (left), $N = 20$ (middle), $N = 26$ (right). Blue circles: analytical solutions (Eq. 3.13). Only modes with frequency between 0 and $f_{max} = (N+1)c_0/4L$ are plotted. Bottom row: Singular values distribution of the Σ matrix normalized by the maximum of singular values, for $N = 12$ (left), $N = 20$ (middle) and $N = 26$ (right). The red horizontal dashed lines correspond to 10^{-16}, an indication of the machine precision.

The agreement between physical modes and the reference frequencies remains good for all the values of the frame size N. For $N = 12$, both the direct frame (Eq. 3.9) and the SVD-based approach (Eq. 3.20) give the correct results, without spurious modes, since the global system is well conditioned in both cases. For $N = 20$ the conditioning of the problem is $\kappa(\boldsymbol{\Sigma}) \simeq 1.2 \times 10^{14}$ using the SVD-based framework and $\kappa(\boldsymbol{\Lambda}) \simeq 1.4 \times 10^{28}$ for the original formulation (Eq. 3.9). Consistently, the three spurious modes observed previously disappear with the new framework (compare Fig. 3.4 and top row of Fig. 3.6 for $N = 20$). Note that, comparing Fig. 3.5 and the bottom row of Fig. 3.6, the theoretical relation between the sets of singular values $s(\boldsymbol{\Sigma}) = \sqrt{s(\boldsymbol{\Lambda})}$, is not met for the lowest singular values for $N = 20$ and $N = 26$. The computation of singular values itself is actually sensitive to the conditioning, and singular values close to/lower than $10^{-16} \times \max(s)$ cannot be computed accurately. The important point to understand is if a plateau-like behaviour is present for the lowest singular values, then all singular values located at or below this plateau are considered as numerical zeros, leading to an ill-conditioned system, and consequently, the apparition of spurious modes. As it is illustrated in Fig. 3.6 for $N = 26$, the two lowest singular values have reached a plateau around 6×10^{-16} and two spurious modes are still present even with the new framework. Removing spurious modes in such cases where even the SVD is ill-conditioned is obtained by truncating the $\boldsymbol{U}$ basis. This approach is considered in the following section.

3.6 Truncated SVD - Removing multicollinearity

Even if the $\boldsymbol{U}$ basis is better conditioned than the original frame $\boldsymbol{\phi}$, the SVD process is just a change of basis and we have no control on the conditioning. The previous section shows that even the $\boldsymbol{U}$ basis can be ill-conditioned (case $N = 26$), leading to the appearance of spurious modes in the results when $\kappa(\boldsymbol{\Sigma}) \sim 10^{16}$ or more. For further understanding, one needs to recall that in each geometrical subdomain, the frame $\boldsymbol{\phi}$ is built from two concatenated bases. Therefore, if a vector ϕ_n of the first basis can numerically be approximated by a linear combination of modes in the second basis ($\phi_n \simeq \sum_{m\neq n} \alpha_m \phi_m$) then this ϕ_n leads to a small singular value, which increases $\kappa(\boldsymbol{\Sigma})$. This phenomenon is commonly called multicollinearity [97] and can be removed by using a so-called truncated SVD expressed by,

$$\boldsymbol{\phi}_t = \boldsymbol{U}_t \boldsymbol{\Sigma}_t \boldsymbol{V}_t^T \tag{3.21}$$

The truncated SVD serves to approximate the space spanned by the initial $\boldsymbol{\phi}$ basis with another basis $\boldsymbol{\phi_t}$, built from a truncated set of vectors $\boldsymbol{U}_t$ as well as truncated matrices $\boldsymbol{\Sigma}_t$ and $\boldsymbol{V}_t^T$. The difference between the classical SVD (Eq. 3.15) and the truncated SVD (Eq. 3.21) is the size of matrices $\boldsymbol{U}$, $\boldsymbol{\Sigma}$ and $\boldsymbol{V}$. The matrix $\boldsymbol{\Sigma}$ in Eq. 3.15 is the diagonal matrix of singular values of $\boldsymbol{\phi}$, sorted in descending order such that $\boldsymbol{\Sigma} = diag(\sigma_1, \cdots, \sigma_N)$ with $\sigma_1 > \cdots > \sigma_N$, where N is the frame size. The matrix $\boldsymbol{\Sigma}_t$ in Eq. 3.21 is a reduction of $\boldsymbol{\Sigma}$, where singular values lower than $\varepsilon_t \times \max(s(\boldsymbol{\Sigma}))$ are removed, where ε_t is the truncation threshold. The truncated singular matrix is then $\boldsymbol{\Sigma}_t = diag(\sigma_1, \cdots, \sigma_{N_t})$, where $N_t < N$ is the number of remaining singular values in $\boldsymbol{\Sigma}_t$. $\boldsymbol{U}_t$ is a row vector of size N_t which contains the first N_t orthonormalized vectors, corresponding to the N_t largest singular values. $\boldsymbol{V}_t^T$ is of size $N_t \times N$ in order to have $\boldsymbol{\phi}_t$ of the same size as $\boldsymbol{\phi}$. Figure 3.7 illustrates the truncated SVD of $\boldsymbol{\phi}$.

The natural choice for the threshold of the truncated SVD is machine precision $\varepsilon_t = 10^{-16}$. However, as explained earlier, the lowest singular values being difficult to compute accurately, a preferred threshold is set to $\varepsilon_t = 10^{-15}$, just above the plateau of the two lowest singular values (see the left plot in Fig. 3.8). Since spurious modes are still present for the case $N = 26$, only this case is focused on the following. To study the robustness of the

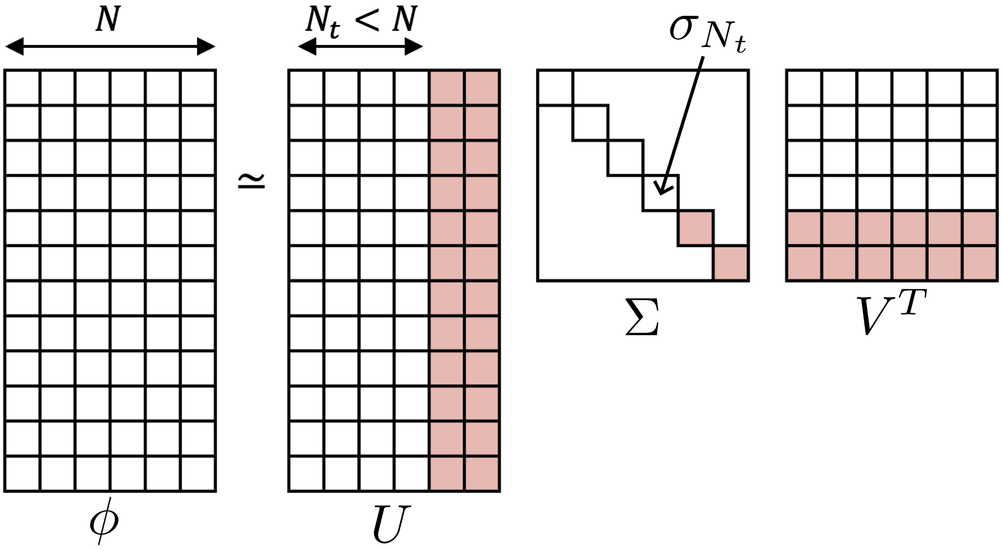


Figure 3.7: Schematic representation of the truncated SVD of the frame ϕ and associated matrix dimensions. Truncated blocks are highlighted in red.

method, six different values of the truncation threshold are tested from $\varepsilon_t = 10^{-15}$ to $\varepsilon_t = 10^{-1}$, which as a result automatically decreases the size of the reduced frame from $N = 26$ to $N_t = 24$ up to $N_t = 14$ in the last case. The results are presented on the right hand side frame of Fig. 3.8, where the two spurious modes have vanished from the numerical solutions because the two multicollinear modes have been removed from the frame.

The truncation threshold is seen not to impact the calculated growth rate compared to the analytical solutions until $\varepsilon_t = 10^{-3}$. While for $\varepsilon_t = 10^{-2}$ the results remain stable with a small shift occurring at higher frequency, for a truncation threshold fixed at $\varepsilon_t = 10^{-1}$, the method strongly deviates from the analytical solutions. Overall, results clearly show the robustness of the proposed methodology, indicated by the large extent of ε_t values that is user-defined, and which does not seem to impair the final quality of obtained results.

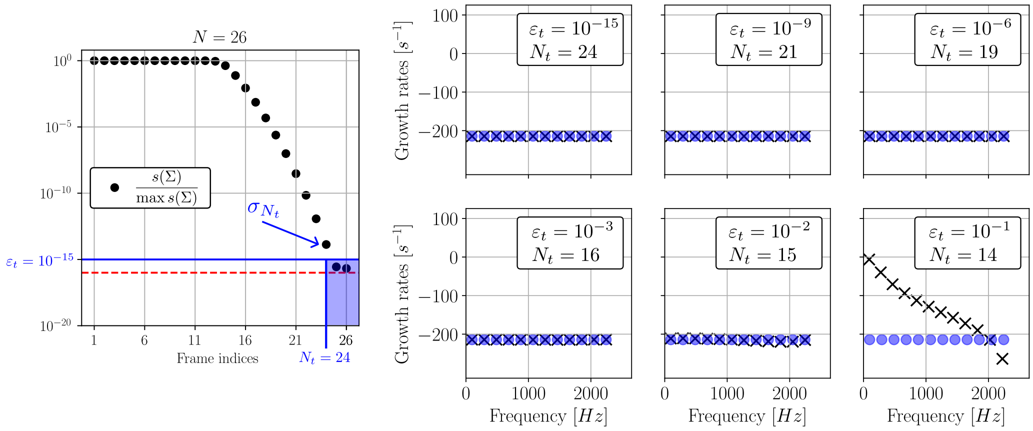


Figure 3.8: Left: Singular values distribution of the Σ matrix normalized by the maximum of singular values, for $N = 26$. The red horizontal dashed lines correspond to 10^{-16}, an indication of the machine precision. In this plot, the truncation threshold is set to $\varepsilon_t = 10^{-15}$ just above the plateau, leading to $N_t = 24$ remaining vectors in the U-basis. Right: Modes of the configuration depicted in Fig. 3.2 displayed in the frequency plane, for six different values of the truncation threshold ε_t. Crosses: Numerical results from the truncated-SVD-based modal expansion method, with the initial frame size $N = 26$. Blue circles: analytical solutions (Eq. 3.13). Only modes with frequency between 0 and $f_{max} = (N+1)c_0/4L$ are plotted.

Once a truncated SVD is performed, the distance between the spaces spanned by either $\boldsymbol{\phi}_t$ or $\boldsymbol{\phi}$ can be quantified thanks to the Eckart-Young Theorem (see [98, Theorem 2.4.8, p. 79]) which states:

$$\|\boldsymbol{\phi} - \boldsymbol{\phi}_t\|_2 = \sigma_{N_t+1} \tag{3.22}$$

where $\|\cdot\|_2$ denotes the 2-norm of a matrix that is:

$$\|\boldsymbol{\phi}\|_2 = \max_{\|x\|_2=1} \|\boldsymbol{\phi} x\|_2 \tag{3.23}$$

For instance, with $\varepsilon_t = 10^{-15}$, two modes have been removed from the frame so $\|\boldsymbol{\phi} - \boldsymbol{\phi}_t\|_2 = \sigma_{25} \simeq 2.8 \times 10^{-16}$. The truncation threshold hence allows the user to control the precision of the search space in which the pressure field is decomposed. This section has shown how the truncated SVD ensures that the geometrical subdomains are properly defined and well-conditioned. With this foundation in place, we next address the flame subdomains, which introduce the unsteady heat release driving thermoacoustic interactions.

3.7 Flame modeling

As explained in the introduction, Helmholtz solvers describe how acoustic waves propagate inside a given geometry. But unlike Large Eddy Simulations (LES), which explicitly resolve both acoustics and the flame dynamics in a time-dependent manner, Helmholtz solvers only solve the linear acoustics. The latter do not actually simulate the flame itself, but instead rely on a simplified model to represent how the flame responds to acoustic perturbations. This is done in the context of the frame approach by adding a new state-space dedicated to flame objects. To understand how flames are modeled in such a framework, a flame is added to the 1D duct case depicted in

Fig. 3.2, at the position $x_f = 0.23\ m$ (see Fig. 3.9). The truncation threshold of the SVD is set to $\varepsilon_t = 10^{-15}$ for this section.

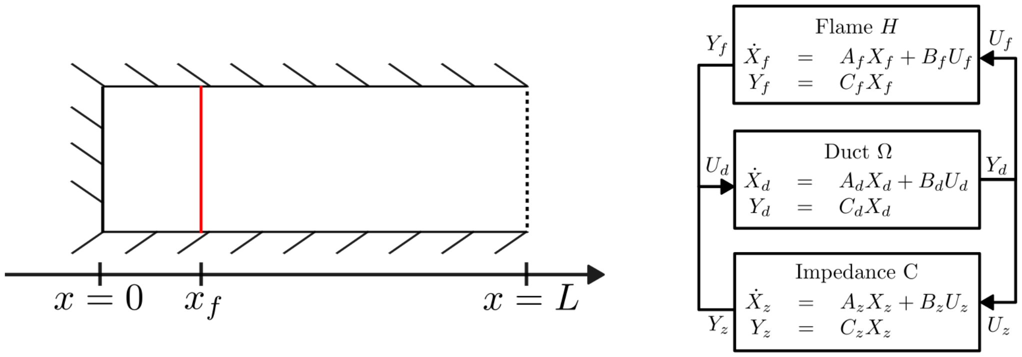


Figure 3.9: Left: studied academic configuration, composed of a 1D duct of length $L = 1\ m$, an impedance boundary condition on the right side and a flame positioned at $x_f = 0.23\ m$ of thickness $\delta_f = 0.02\ m$. Right: the corresponding block diagram.

The first step is to decompose the source term $\omega_T(\boldsymbol{x}, t)$ of Eq. 2.1 as a product of a support function that represents the mean flame shape and the fluctuating heat release rate that is,

$$\omega_T'(\boldsymbol{x}, t) = H(\boldsymbol{x})Q'(t) \tag{3.24}$$

where $H(\boldsymbol{x})$ is a support function (in m^{-3}) that allows to locate the flame in the system, and $Q'(t)$ is the fluctuating heat release that must be modeled. This is where the Flame Transfer Function (FTF) comes into play (see Section 3.7.2).

3.7.1 Flame shape and projection

A mean flame shape $H(\boldsymbol{x})$ must be provided to position the flame in the domain. As explained in [49], a Dirac function leads to Gibbs oscillations, so a Gaussian with a narrow width is preferred:

$$H(\boldsymbol{x}) = \frac{2}{\sqrt{\pi \delta_f}} e^{-\frac{(x-x_f)^2}{\delta_f^2}} \tag{3.25}$$

where x_f is the flame position and δ_f the flame thickness, set at $\delta_f = 0.02$ m in the example depicted in Fig. 3.9. Note that the function $H(\boldsymbol{x})$ (in m^{-3}) has to be normalized over the whole volume V such that: $\int_V H(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = 1$.

Following these notations, the expression of the term $\boldsymbol{\mathcal{F}_U}(t)$ in the state-space of the 1D duct (Eq. 3.20) is written:

$$\left\{ \begin{array}{rcl} p'(\boldsymbol{x},t) & = & \sum_n^N \dot{\Upsilon}_n(t) U_n(x) \\ \ddot{\Upsilon}_n(t) & = & -\sum_k^N [\boldsymbol{\Sigma V}^T \boldsymbol{\Omega^2 V \Sigma}^{-1}]_{n,k} \Upsilon_k(t) \\ & + & \rho_0 c_0^2 \boldsymbol{\mathcal{I}_U}(t) \\ & + & (\gamma-1)\langle H|U_n\rangle Q'(t) \end{array} \right. \tag{3.26}$$

The term $(\gamma-1)\langle H|U_n\rangle Q(t)$ traduces the influence of the flame onto each modal amplitude Υ_n of the unknown pressure field. γ is the heat capacity ratio, and is fixed at $\gamma = 1.4$. The term $\langle H|U_n\rangle$ is the projection of the normalized flame shape onto the frame mode U_n, defined as: $\langle H|U_n\rangle = \int_V H(\boldsymbol{x})U_n(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$. The flame shape is therfore approximated by its projection on the space spanned by the U-basis. The error between the real flame shape $H(\boldsymbol{x})$ and its projection can be evaluated using

$$\epsilon_{fs} = \frac{\left\| H(\boldsymbol{x}) - \sum_n^N \langle H|U_n\rangle U_n(\boldsymbol{x}) \right\|_2}{\|H(\boldsymbol{x})\|_2} \tag{3.27}$$

where $\|\cdot\|_2$ denotes the classical 2-norm of the vector. As highlighted by Fig. 3.10, the frame size $N = 26$ (used in the previous case without flame) is too low to yield a correct spatial representation of the flame shape through its projection onto the U-basis. The projection error is $\epsilon_{fs}(N = 26) = 2.7 \times 10^{-1}$.

The frame size is thus increased to $N = 100$ to have a better approximation of the flame shape, resulting in $\epsilon_{fs}(N = 100) = 5.8 \times 10^{-4}$

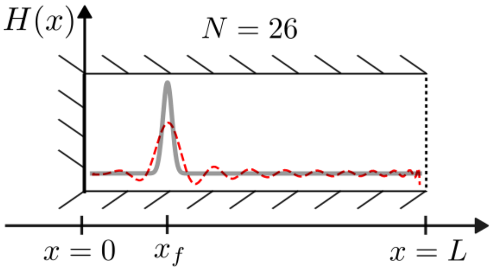


Figure 3.10: The flame added to the configuration depicted in Fig. 3.9. Gray plots: real flame shape $H(x)$ defined in Eq. 3.25. Red dashed plots: reconstruction of the flame shape through its projection onto the U-basis. Left: case $N = 26$. Right: case $N = 100$.

3.7.2 Flame Transfer Function

The FTF is a black box that characterizes how heat release fluctuations $Q'(t)$ respond to upstream acoustic perturbations [62, 63, 99]. Since heat release acts as an acoustic source, its interaction with pressure waves determines whether an instability will grow or decay. The $N - \tau$ FTF based on Crocco's ideas [62] is a common approach, which postulates that the instantaneous heat release of a flame, $Q'(t)$ (in watts), is directly proportional to a velocity fluctuation upstream of the flame, measured at a reference point $\boldsymbol{x_r}$, and at an earlier time $t - \tau$. This is written:

$$Q'(t) = \frac{\overline{Q}}{\overline{u}} N u'(\boldsymbol{x_r}, t - \tau) \tag{3.28}$$

where, $\overline{Q}$ and $\overline{u}$ are respectively the global flame power (in watts) and the velocity of the baseline flow (in $m.s^{-1}$). Note that the velocity fluctuation u' used in the flame transfer function must be evaluated at a specific reference point $\boldsymbol{x_r}$, typically located just upstream of the flame front $\boldsymbol{x_f}$. For the problem to be investigated, $\boldsymbol{x_r} = 0.21\ m$ and $\boldsymbol{x_f} = 0.23\ m$.

Taking the Fourier transform of Eq. 3.28 leads to the frequency-domain expression of the $N - \tau$ FTF:

$$\hat{Q}(\omega) = \frac{\overline{Q}}{\overline{u}} N e^{-j\omega\tau} \hat{u}(\boldsymbol{x_r}, \omega) \tag{3.29}$$

In certain cases, the flame response can also depend on the amplitude of the velocity oscillations in which case it is called a FDF (Flame Describing Function) [15, 100]. FTFs and FDFs are required for any Helmholtz solver and can be obtained analytically in simple cases [101, 102], numerically [103, 104] or experimentally [65]. Regardless of the method used to obtain the flame dynamics, once acquired, employing a Helmholtz solver allows for the computation of thermoacoustic instabilities of the system.

3.7.3 Flame low-order model - Fitting process

In addition to the conditioning issues related to the ill-conditioned Gram matrix (addressed in Sec. 3.5.1), the introduction of flame subdomains also leads to the emergence of non-physical modes that do not correspond to any physical thermoacoustic resonance. These modes originate from numerical artifacts associated with the flame modeling, particularly through the flame transfer function fitting process. Detecting and mitigating these modes is therefore essential. This part illustrates this phenomenon and presents a method to remove these non-physical modes.

The necessary condition to build a time-invariant state-space representation of the flame —or more generally any state-space— is that the ratio between the output and the input can be written at least as a ratio of polynomials. The first step is thus to express the transfer function of Eq. 3.29 as a ratio $\mathcal{N}(\omega)/\mathcal{D}(\omega)$. However, a flame transfer function is often available as a dataset of discrete points, defined by a gain N_k and a phase $\varphi_k = \omega_k \tau_k$ for each pulsation ω_k. The subscript k refers to the sampling of measurements.

In this context, the fitting step is essential for building the low-order model of the flame. To perform such a fit, the Vector Fitting (VF) algorithm developed by Triverio *et al.* [105–108] is chosen. To do so, the open source code provided by the authors in Matlab language has been translated in Python to be part of the STORM package. The reasons of the choice of this algorithm is discussed in more details in section 3.7.5. The Vector Fitting algorithm returns the function $\widetilde{\mathcal{H}}(j\omega)$ that approximates the given dataset, composed of $\bar{k}$ measurements:

$$\widetilde{\mathcal{H}}(j\omega_k) \simeq N_k e^{-j\omega_k \tau_k}, \quad \forall k \in [\![1, \bar{k}]\!] \tag{3.30}$$

The general form of $\widetilde{\mathcal{H}}$ is a sum of so-called Pole Base Functions (PBFs) of the form:

$$\widetilde{\mathcal{H}}(j\omega) = R_0 + \sum_{i=1}^{N_{PBF}} \frac{R_i}{j\omega - p_i} \tag{3.31}$$

where R_i and p_i are the residues and the poles of the fit, respectively and N_{PBF} is the order of the fit. The heat release rate that is finally read by the code follows the expression of the form

$$Q'(t) = \frac{\overline{Q}}{\overline{u}} \left(R_0 + \sum_{i=1}^{N_{PBF}} \frac{R_i}{j\omega - p_i} \right) u'(\boldsymbol{x_r}, t) \tag{3.32}$$

As detailed in Tab. 3.1, the state-space of the flame subdomain takes $u'(\boldsymbol{x_r}, t)$ as input and must return $Q'(t)$ as an output to be connected to the duct subdomain. The corresponding state-space is given in Appendix A.5. Under this formalism, a resolution can be performed for the case depicted in Fig. 3.9. All the different parameters used for this resolution are summarized in Tab. 3.2. The gain N and the delay τ of the FTF are *a priori* depending on the frequency ω but they are set constant for simplicity and to be compared to the pseudo-analytical solutions. The latter are obtained using the same methodology as provided in App. B.2.

Table 3.2: Physical and geometrical parameters of the configuration depicted in Fig. 3.9

Geometric	Flame and FTF
$L = 1\ m$	$x_r = 0.21\ m$
$\gamma = 1.4$	$x_f = 0.23\ m$
$\rho_0 = 1.177\ kg.m^{-3}$	$\delta_f = 0.02\ m$
$c_0 = 390.3\ m.s^{-1}$	$\overline{Q} = 40\ W$
$N = 100$ (Frame size)	$N_{PBF} = 20$
$Z = 0.5$	$\tau = 1\ ms$
	$\overline{u} = 1\ m.s^{-1}$
	$N = 1$ ($N - \tau$ FTF)

Results of the presented modal expansion method are compared to the pseudo analytical solutions, provided by the approximation of a 1D duct and an infinitely thin flame. Results in terms of frequencies and growth rates are reported in Fig. 3.11.

The frequency and growth rate of each physical modes appear to be correct, the little shift between black crosses and blue dots in Fig. 3.11 is due to the modeling of the flame, which has a zero thickness when using the pseudo-analytical method instead of $\delta_f = 0.02\ m$ in the presented method. Three non-physical modes however emerge in the numerically obtained solutions (see the cross points that do not match the blue circles in Fig. 3.11). They correspond to the complex poles of the fit p_k which have not converged

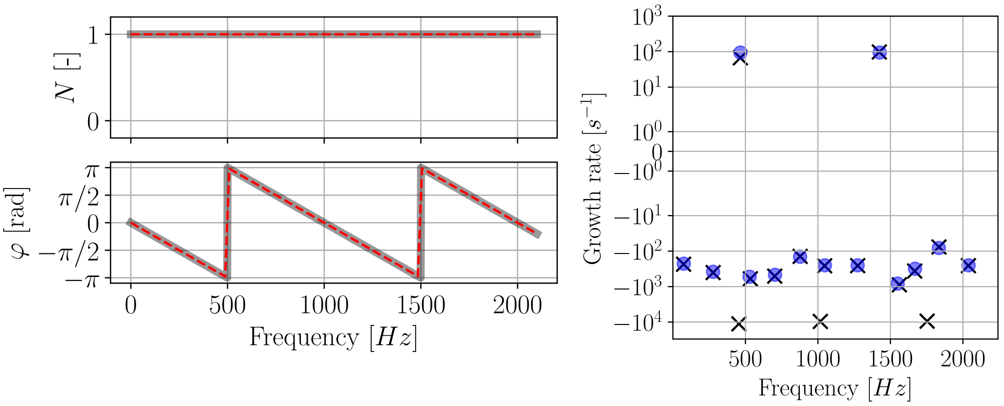


Figure 3.11: Left: Gain N and phase $\varphi = \omega\tau$ of the FTF. Gray plot: constant $N-\tau$ FTF with $N=1$ and $\tau = 0.001$ s. Red dashed plot: the fit reconstructed using Eq. 3.31. Right: direct results of the modal expansion method (black crosses), and pseudo analytical solutions (blue dots).

towards a physical mode.

3.7.4 Spurious eigenmodes due to the fit

The method to identify these spurious modes and remove them from the solution is based on the observation that physical modes should be independent of the order of the fit N_{PBF}. The strategy is to fit a given FTF with two different orders and compare both numerical results obtained using these two fits. The modes with the highest shift in the complex plane are then considered as non-physical. To do so, the shift $\epsilon_s(n)$ associated with each mode n is defined as follows. If $f_n^{PBF_1}$, $\sigma_n^{PBF_1}$ are the frequency and growth rate of the mode n in the family of modes obtained using N_{PBF_1} and $f_m^{PBF_2}$, $\sigma_m^{PBF_2}$ are the frequency and growth rate of the mode m in the family of modes obtained using N_{PBF_2} —it can be assumed that mode m is the closest mode in the frequency plane to mode n— then,

$$\epsilon_s(n) = \sqrt{\left(f_n^{PBF_1} - f_m^{PBF_2}\right)^2 + \left(\sigma_n^{PBF_1} - \sigma_m^{PBF_2}\right)^2} \tag{3.33}$$

For the problem considered, the second fit is obtained using $N_{PBF_2} = 25$ while $N_{PBF_1} = 20$. On the left part of Fig. 3.12, this shift is the cartesian distance between a black cross and the closest red star. The value of $\ln(\epsilon_s(n))$ is depicted on the right part of Fig. 3.12. The one-dimensional K-means algorithm [109] is applied to the dataset $\ln(\epsilon_s(n))$ to automatically identify two clusters: the first with the lowest center value is colored in green which corresponds to the physical modes (smaller shift), and the second one colored in orange which coincides with pole-induced solutions that are effectively non-physical.

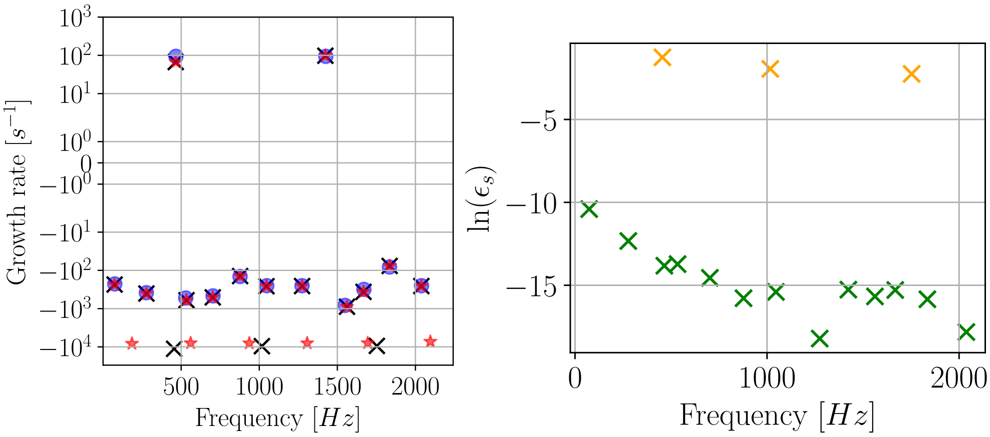


Figure 3.12: Left: direct results of the modal expansion method using either $N_{PBF_1} = 20$ (black crosses) or $N_{PBF_2} = 25$ (red stars), and pseudo analytical solutions (blue dots). Right: $\ln(\epsilon_s)$ computed for each modes (each black crosses on the left figure). The coloring is based on the K-means algorithm with two clusters.

In this academic configuration, the two clusters of physical and non-physical modes are well separated, allowing us to clearly identify the physical ones (green crosses one the right part of Fig. 3.12). The capability of the overall methodology to handle a more complex case is investigated in chapter 5.

Since these spurious modes originate from the poles of the fitted flame transfer functions (or impedance functions), it is essential to ensure that the fitting procedure itself is accurate and robust. The choice of the fitting algorithm plays a central role in ensuring numerical efficiency and stability. The next section presents the Vector Fitting algorithm and explains the reasons for its selection over alternative approaches.

3.7.5 Vector Fitting algorithm

Consider a dataset (ω_k, H_k) of size $\bar{k}$, which represents frequency-domain measurements, such as a flame transfer function or an impedance. The goal is to determine a rational function $\widetilde{\mathcal{H}}$ that approximates these measurements:

$$\widetilde{\mathcal{H}}(j\omega_k) \simeq H_k, \quad \forall k \in [\![1, \bar{k}]\!] \tag{3.34}$$

Enforcing a rational form for $\widetilde{\mathcal{H}}$ is essential, as it enables the construction of a state-space representation of the system under consideration (e.g., impedance or flame). The two main formulations of $\widetilde{\mathcal{H}}$ are the Padé approximation [110, 111] and the Pole-Residue approximation [112, 113].

- **Padé approximation**. The function $\widetilde{\mathcal{H}}$ is a ratio of two polynomials as follows:

  $$\widetilde{\mathcal{H}}(j\omega) = \frac{\sum_{n=0}^{\bar{n}} a_n (j\omega)^n}{\sum_{n=0}^{\bar{n}} b_n (j\omega)^n} \tag{3.35}$$

  where a_n and b_n are unknown coefficients, and $\bar{n}$ is the order of the fit.

- **Pole-Residue approximation**. The function $\widetilde{\mathcal{H}}$ is written as a pole-residue formulation:

$$\widetilde{\mathcal{H}}(j\omega) = R_0 + \sum_{n=1}^{\bar{n}} \frac{R_n}{j\omega - p_n} \tag{3.36}$$

where p_n are the poles, R_n are the residues, and R_0 is a constant offset that accounts for any nonzero behavior at high frequencies (e.g., a static gain or asymptotic value). Compared to polynomial-based approximations, pole-residue formulations provide a more compact and physically meaningful representation, particularly for systems exhibiting resonances or exponential decay. The fit order $\bar{n}$ also referred to as the number of Pole-Base Functions (PBFs) used in the approximation, is henceforth denoted as N_{PBF}.

Considering either the Padé or the pole-residue approximation, the problem is to determine the unknowns (e.g. a_n and b_n for Padé or R_n and p_n for pole-residue) by minimizing a given error. To do so, the L_2 norm between the sample H_k and the model $\widetilde{\mathcal{H}}(j\omega_k)$

$$e^2 = \frac{1}{\bar{k}} \sum_{k=1}^{\bar{k}} \left| H_k - \widetilde{\mathcal{H}}(j\omega_k) \right|^2 \tag{3.37}$$

is usually used. Note that the both the Padé and the pole-residue formulations are mathematically equivalent, as there is a relation between coefficients a_n, b_n and the pole-residue p_n, R_n. This relation is explicited by Douasbin [114] (Chapter 8).

Vector Fitting (VF) is a numerical algorithm that approximates a function H from frequency-domain data $(\omega_k, H_k)_{k \leq \bar{k}}$ in the form of a sum of rational basis functions. It has been originally proposed by Gustavsen and Semlyen in 1999 [115]. Its primary goal is to identify a set of poles and residues that accurately represent the function while ensuring stability and robustness. The combustion community has used this algorithm to approximate the response of flame and acoustic impedance [33, 116–122].

Unlike traditional polynomial regression or Padé approximation methods, Vector Fitting iteratively refines the pole locations using the Sanathanan-Koerner transformation [123], which helps mitigate numerical sensitivity issues and avoids the introduction of spurious poles. One of the key advantages of this method is its ability to provide a stable model, even when the initial data might lead to unstable poles using other approximation techniques. Additionally, it naturally enforces causality in the pole identification process. In dynamic system theory, causality dictates that a system's response to an excitation cannot precede the excitation itself. Mathematically, this implies that the poles must lie in the left half-plane

$$\Re(p_n) \leq 0, \; \forall n \tag{3.38}$$

to ensure an exponentially decaying impulse response [107, 124, 125]. This constraint guarantees a stable and physically realizable system.

Another fundamental aspect of Vector Fitting is the capability to handle real-valued transfer functions. In practice, if a transfer function originates from a physical system with real parameters, it must satisfy the Hermitian symmetry condition [126, 127]:

$$H(-j\omega) = H^*(j\omega) \tag{3.39}$$

This property imposes that poles and their associated residues must appear in complex conjugate pairs when the frequency-domain data is real. Consequently, during the pole decomposition process, if p_n is a pole, then p_n^* must also be a pole, with associated conjugate residues. This condition prevents the introduction of artificial imaginary components in the system's time-domain response. Under the reality condition, the general formulation of the Pole-Residue formulation Eq. 3.36 can be rewritten:

$$\widetilde{\mathcal{H}}(j\omega) = R_0 + \sum_{n=1}^{n_r} \frac{R_n}{j\omega - p_n} + \sum_{n=n_r+1}^{n_r+n_c} \left[\frac{R_n}{j\omega - p_n} + \frac{R_n^*}{j\omega - p_n^*} \right] \tag{3.40}$$

where n_r is the number of real poles and n_c the number of complex ones. The fit order (i.e number of pole-base functions) is then $N_{PBF} = n_r + 2n_c$.

Finally, a set of initial poles must be provided to the algorithm. This choice is discussed in [105, 128], where numerical tests show that a linear distribution of initial poles with a small and negative real part over the bandwidth spanned by the samples H_k leads to the best conditioning of the least squares problems to be solved. Furthermore, if N_{PBF} is even and the frequency sample $(\omega_k)_{k \leq \bar{k}}$ is sorted in ascending order, the initial poles can be set as [125]:

$$p_n^{(0)} = \begin{cases} (-\alpha + j) \left[\omega_1 + \dfrac{\omega_{\bar{k}} - \omega_1}{N_{PBF}/2 - 1}(n-1) \right] & \text{for } n \leq N_{PBF}/2 \\ \\ \left(p_{n-N_{PBF}/2}^{(0)} \right)^* & \text{for } n > N_{PBF}/2 \end{cases} \tag{3.41}$$

where α is a constant parameter that is typically set to 0.01. This initial setup generates $N_{PBF}/2$ pairs of complex conjugate poles, linearly distributed over the frequency sample range $[\omega_1, \omega_{\bar{k}}]$. The imaginary part is set to be significantly larger than the real part (relative to α which appears only for the real part) to improve the numerical conditioning of the algorithm. Initial poles can be also distributed logarithmically for optimal results if the frequency range of interest spans several decades, but this is rarely the case when dealing with thermoacoustics.

Note that although the algorithm ensures a precise fit over the given frequency points, it does not necessarily generalize well outside the sampled frequency range.

In summary, Vector Fitting is an effective approach for function approximation, ensuring causality and compliance with real-valued function properties. Its real-valued form, with the automatic pole initialization provided, is implemented in STORM.

3.8 Numerical frame

The first step in the modal expansion method is the computation of frame modes $(\phi_n(\boldsymbol{x}))_{n\leq N}$. For simple geometric domains, the frame can be constructed analytically. However, in more complex 3D cases, it must be computed using a Helmholtz solver, based on the FEniCS library in the present work.

FEniCS is an open-source library designed to solve PDEs using the finite element method (FEM) [129]. Widely used by scientists and engineers, it provides an efficient framework for solving partial differential equations (PDEs) while maintaining a high level of abstraction. It offers a scalable framework for computationally demanding problems with MPI [130] parallelization, adapted for HPC clusters. It also has Python interface named mpi4py [131, 132]. Using massive parallelism, it can handle large-scale simulations and complex problems with high efficiency. Additionally, its Python-based scripting interface simplifies the definition of equations in the code, while the C++ backend ensures speed and scalability.

The problem we are interested in relies on the homogeneous Helmholtz equation with no source term in the frequency space, or equivalently the eigenproblem expressed by:

$$\begin{cases} -\boldsymbol{\nabla}\cdot\left(\dfrac{1}{\rho_0}\boldsymbol{\nabla}\phi_n\right)=\dfrac{\omega_n^2}{\gamma p_0}\phi_n \text{ in } \Omega \\ \nabla_n\phi_n = 0 \text{ on } \Gamma_N \\ \phi_n = 0 \text{ on } \Gamma_D \end{cases} \tag{3.42}$$

where ∇_n is the gradient operator projected onto the outward-pointing normal vector to the surface where the BC is applied. Ω is the geometrical domain and its boundary $\partial\Omega = \Gamma_N \cup \Gamma_D$ is split into two parts where either a wall BC (Γ_N) or an open atmosphere BC (Γ_D) is applied. These conditions correspond to the homogeneous Neumann and Dirichlet boundary conditions, respectively.

3.8.1 Finite Element Method

To solve this equation using FEniCS, the weak form must be formulated. To do so, one starts by defining the Sobolev space $H^1(\Omega)$:

$$H^1(\Omega) = \{\phi_n \in L^2(\Omega) | \nabla_{x_i}\phi_n \in L^2(\Omega), 1 \leq i \leq 3\} \tag{3.43}$$

where ϕ_n is the unknown solution of Eq. 3.42 and function of ω_n, and L^2 is the space of square-integrable functions in the domain Ω. The inner product in this space is defined as:

$$\langle\phi_n|\phi_m\rangle = \int_\Omega \phi_n(\boldsymbol{x})\phi_m(\boldsymbol{x})\mathbf{d}\boldsymbol{x} \tag{3.44}$$

To approximate the solution numerically, test functions v_h from the finite-dimension function space $V_h \subset H^1(\Omega)$ are introduced so that:

$$V_h = \{v_h \in H^1(\Omega) | v_{h,K} \in \mathcal{P}_k(K)\ \forall K \in \mathcal{T}\} \tag{3.45}$$

where $\mathcal{P}_k(K)$ is the space of polynomials of degree $\leq k$ on each element K of the mesh (composed of triangles for 2D and tetraedra for 3D) and $\mathcal{T}$ stands for the set of elements of the mesh. For further developments, we only consider Lagrange elements of order 1, i.e. $k = 1$. Elements v_h of the space V_h are therefore of the form:

$$v_h(\boldsymbol{x}) = \sum_{i}^{N_v} a_i \psi_i(\boldsymbol{x}) \tag{3.46}$$

where a_i are unknown coefficients and ψ_i are the linear basis functions associated with the mesh, as illustrated in Fig. 2.3. N_v is the number of vertices of the mesh.

Within the finite element framework, Eq. 3.42 is multiplied by a test function $v_h \in V_h$ and integrated over the domain Ω to obtain:

$$-\int_\Omega \boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} \phi_n \right) v_h \mathbf{d}\boldsymbol{x} = \omega_n^2 \int_\Omega \frac{1}{\gamma p_0} \phi_n v_h \mathbf{d}\boldsymbol{x}, \ \forall v_h \in V_h \tag{3.47}$$

Applying integration by parts and the Green theorem to the first term leads to:

$$-\int_\Omega \boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} \phi_n \right) v_h \mathbf{d}\boldsymbol{x} = \int_\Omega \frac{1}{\rho_0} \boldsymbol{\nabla} \phi_n \cdot \boldsymbol{\nabla} v_h \mathbf{d}\boldsymbol{x} - \underbrace{\int_{\partial\Omega} \frac{1}{\rho_0} \nabla_n \phi_n v_h \mathbf{d}\boldsymbol{s}}_{\text{Boundary term}} \tag{3.48}$$

The boundary term can be further decomposed using the two kind of boundary conditions i.e Dirichlet and Neumann. The total boundary surface $\partial\Omega$ is the union of the surfaces where these two different boundary conditions are applied $\partial\Omega = \Gamma_D \cup \Gamma_N$:

$$\int_{\partial\Omega} \frac{1}{\rho_0} \nabla_n \phi_n v_h \mathbf{d}\boldsymbol{s} = \int_{\Gamma_D} \frac{1}{\rho_0} \nabla_n \phi_n v_h \mathbf{d}\boldsymbol{s} + \int_{\Gamma_N} \frac{1}{\rho_0} \nabla_n \phi_n v_h \mathbf{d}\boldsymbol{s} \tag{3.49}$$

The first term on the right hand side of Eq. 3.49 vanishes if the test function v_h is correctly chosen ($v_h = 0$ on Γ_D). The second term then always vanishes because $\nabla_n \phi_n = 0$ on Γ_N (Neumann boundary). The weak form of the homogeneous Helmholtz equation with no source term is hence written

$$\int_\Omega \frac{1}{\rho_0} \boldsymbol{\nabla} \phi_n \cdot \boldsymbol{\nabla} v_h \mathbf{d}\boldsymbol{x} = \omega_n^2 \int_\Omega \frac{1}{\gamma p_0} \phi_n v_h \mathbf{d}\boldsymbol{x} \tag{3.50}$$

Like $v_h(\boldsymbol{x})$, the unknown function $\phi_n(\boldsymbol{x})$ is approximated using Lagrange elements of order 1 so $\phi_{n,h}(\boldsymbol{x}) = \sum_i b_i \psi_i(\boldsymbol{x}) = \underline{\boldsymbol{\phi_n}} \boldsymbol{\psi}(\boldsymbol{x})^{\boldsymbol{T}}$. The vector $\underline{\boldsymbol{\phi_n}}$ contains the value of $\phi_{n,h}$ at each vertex of the mesh (i.e the amplitude coefficients b_i here). Note that in the Galerkin formulation [133] developed here, the function v_h is chosen to be equal to the basis function ψ_i for each degree of freedom i. Evaluating Eq. 3.50 at each degree of freedom leads to the generalized discrete eigenvalue problem

$$\underline{\underline{\boldsymbol{A}}}\ \underline{\boldsymbol{\phi_n}} = \omega_n^2\ \underline{\underline{\boldsymbol{B}}}\ \underline{\boldsymbol{\phi_n}} \tag{3.51}$$

where $A_{i,j} = \displaystyle\int_\Omega \frac{1}{\rho_0} \boldsymbol{\nabla} \psi_i \cdot \boldsymbol{\nabla} \psi_j \mathbf{d}\boldsymbol{x}$ are the coefficients of the stiffness matrix $\underline{\underline{\boldsymbol{A}}}$ and $B_{i,j} = \displaystyle\int_\Omega \frac{1}{\gamma p_0} \psi_i \psi_j \mathbf{d}\boldsymbol{x}$ are the coefficients of the mass matrix $\underline{\underline{\boldsymbol{B}}}$ weighted by γp_0 when the classical mass matrix $\underline{\underline{\boldsymbol{M}}}$ associated with a mesh reads,

$$M_{i,j} = \int_\Omega \psi_i \psi_j \mathbf{d}\boldsymbol{x} \tag{3.52}$$

which can be useful to approximate numerically the inner product between two functions as

$$\langle \phi_n | \phi_m \rangle = \int_\Omega \phi_n(\boldsymbol{x}) \phi_m(\boldsymbol{x}) \mathbf{d}\boldsymbol{x} \simeq \underline{\boldsymbol{\phi_n^T}}\ \underline{\underline{\boldsymbol{M}}}\ \underline{\boldsymbol{\phi_m}} \tag{3.53}$$

Note that $\underline{\underline{\boldsymbol{M}}}$ is *a priori* not proportional to $\underline{\underline{\boldsymbol{B}}}$ as γ may depend on $\boldsymbol{x}$.

FEniCS, SLEPc and UFL

The Unified Form Language (UFL) [134] is a domain-specific language in Python designed to define finite element variational forms. It is commonly used in libraries like FEniCS for solving partial differential equations (PDEs). UFL provides a high-level, symbolic syntax, enabling users to specify integrals, test functions, trial functions, and differential operators in a concise and readable way. Its symbolic expressions are compiled into efficient low-level code for numerical computation. We define the weak form of the Helmholtz equation (Eq. 3.50) as follows:

Listing 3.1: Defining weak formulation of the Helmholtz equation using FEnICS

```
from fenics import *
from ufl import dx, inner, grad
from slepc4py import SLEPc

# Assuming a given mesh,
# Lagrange elements of order 1
# are defined as follows
fs = FunctionSpace(mesh, "Lagrange", 1)
u_h = TrialFunction(fs)
v_h = TestFunction(fs)

# Define the UFL form of the stiffness
# matrix and weighted mass matrix
a = inner(1/rho0 * grad(u_h), grad(v_h)) * dx
b = inner(1/gamma/p0 * u_h, v_h) * dx
```

The weak form is then used by the FEniCS Form Compiler (FFC) [135], which generates the matrices of the generalized eigenvalue problem Eq. 3.51. This generates low-level C codes for local tensors to be globally assembled by DOLFIN, the C++ core of the FEniCS project. These tensors are then assembled as sparse matrices and formatted with PETSc (Portable, Extensible Toolkit for Scientific Computation) [136], the primary linear algebra backend.

Listing 3.2: Assembling PETSc sparse matrices with FEniCS

```
# Assuming a given mesh
a_petsc = PETScMatrix(mesh)
b_petsc = PETScMatrix(mesh)
assemble(a, tensor=a_petsc)
assemble(b, tensor=b_petsc)
```

FEniCS also uses SLEPc (Scalable Library for Eigenvalue Problem Computations) [137] and its Python bindings slepc4py [138]. This library is built on top of PETSc which is specialized in solving large-scale eigenvalue problems. SLEPc can handle several types of eigenproblem, particularly the Generalized Hermitian Eigen-Problem (GHEP) as defined by Eq.3.51.

Listing 3.3: Defining the eigensolver and some of its parameters using FEniCS

```
# Initialize the eigensolver with PETSc matrices
eps = SLEPc.EPS().create(mesh)
eps.setOperators(a_petsc.mat(), b_petsc.mat())
eps.setType(SLEPc.EPS.Type.KRYLOVSCHUR)
eps.setTolerances(tol, max_ite)
eps.setConvergenceTest(SLEPc.EPS.Conv.REL)

# Generalized Hermitian Problem
eps.setProblemType(eps.ProblemType.GHEP)
```

The SLEPc solver must however be configured according to different parameters summarized in Table 3.3 where default values are provided.

Among all possible parameters, provided ones are chosen for the following reasons.

- **Solver type**

  Krylov eigensolvers are among the most successful methods to approximate eigenvalues/vectors of large and sparse matrices. The reader is referred to [139, 140] for details. Those solvers use Krylov subspace

Table 3.3: Solver parameters and their default values once incorporated in the STORM package.

Solver parameter	Default value
Solver type	Krylov-Schur
Convergence Test	Relative to eigenvalues
Tolerance	10^{-6}
Maximum iterations	10^{5}
Spectral Transform	Shift and invert
Shift	10
Spectrum	Smallest positive

[141–143] to capture the essential information of the matrix $\boldsymbol{A}$ and allows us to approximate the solution of a system. To do so, the Krylov subspace $\mathrm{K}(\boldsymbol{A}, \boldsymbol{b})$ is generated by repeatedly applying the matrix $\boldsymbol{A}$ to an initial vector $\boldsymbol{b}$. The idea is to approximate the solution of a problem without having to handle the full matrix directly, which is particularly useful for large-scale problems and such that,

$$\mathrm{K}(\boldsymbol{A}, \boldsymbol{b}) = \mathrm{span}\left(\boldsymbol{b}, \boldsymbol{A}\boldsymbol{b}, \boldsymbol{A}^{\boldsymbol{2}}\boldsymbol{b}, \cdots, \boldsymbol{A}^{\boldsymbol{m-1}}\boldsymbol{b}\right) \tag{3.54}$$

where m is the size of the Krylov subspace, that grows at each iteration of the solver. Without restarting (standard Arnoldi/Lanczos), it increases indefinitely, making computations impractical for large problems, particularly due to memory issues. Restarting techniques, like Implicitly Restarted Arnoldi [144] or Krylov-Schur [145], limit the subspace size to a fixed value M. The solver hence expands the subspace up to M, then restarts by shrinking it while retaining key spectral information before growing again. This cycle prevents excessive memory usage while still refining eigenvalue approximations efficiently. By default in STORM, the Krylov-Schur solver is used but users are free to choose other options such as Implicitly Restarted Arnoldi, Jacobi-Davidson [146], or Lanczos [147]. They can even implement their own custom solver if needed.

Note that a general rule of thumb for the maximum size of Krylov subspace is $M = 2 * N_{modes}$, where N_{modes} is the number of eigenvalues requested by the user.

- **Convergence test**

  The solver stops iterating when the relative error in the computed eigenvalues is below a given tolerance. This ensures precision in the eigenvalue computation, particularly when their magnitudes vary widely. By default, the convergence test is set relative to the eigenvalue, meaning that the criterion compares the relative error of the computed eigenvalue with respect to its estimated value. Mathematically, if λ and $\boldsymbol{v}$ are a couple of eigenvalue/vector of $\boldsymbol{A}$ at a given iteration, the convergence condition is expressed as

  $$\frac{\|\boldsymbol{A}\boldsymbol{v} - \lambda \boldsymbol{v}\|}{|\lambda|} \leq tol \tag{3.55}$$

  Other types of convergence tests can also be used, like the absolute convergence test, or even a custom criterion defined by the user.

- **Tolerance**

  Specifies the stopping criterion for the convergence test. A lower tolerance provides higher accuracy but increases computational cost. The typical range of this parameter for our systems is about 10^{-2} to 10^{-8}.

- **Maximum iterations**

  The solver stops after the maximum of iterations is reached even if convergence hasn't been reached. Typically, the number of maximum acceptable iterations is set to $\sim 10^5$.

- **Spectral transform**

  One possibility for computing interior eigenvalues of a given matrix $\boldsymbol{A}$ is to apply a spectral transformation like a shift-and-invert. In this case, the eigensolver builds a Krylov subspace Eq. 3.54 associated to the shifted and inverted matrix, $(\boldsymbol{A} - \tau \boldsymbol{I})^{-1}$, so that the eigenvalues closest to the shifting parameter τ become dominant and well separated. This approach has been used successfully for a long time especially in the context of symmetric problems [148, 149]. Others spectral transformations like Cayley transforms [150], Jacobi-Davidson [151], and rational approximations [152] may be more suited for particular types of eigenvalue problems, such as non-Hermitian systems. Since the homogeneous Helmholtz equation with trivial boundary conditions Eq. 3.42 leads to a symmetric eigenproblem, the shift-invert preconditioner is chosen.

- **shift**

  The shift τ in the shift-invert preconditioner must be close to zero as we focus on the smallest positive eigenvalues. However, setting $\tau = 0$ leads to conditioning issues when 0 is also an eigenvalue of $\boldsymbol{A}$. This happens when Neumann boundary conditions are applied at all boundaries of a domain, which gives a constant mode with a zero frequency. To bypass this issue, the shift is set to a value close to zero: i.e by default $\tau = 10$. This value is close to zero in the sense that the first non-zero, positive eigenvalue λ_{min} is typically on the order of $\lambda_{min} = 4\pi^2 f_{min}^2 \sim 10^5$ and for which f_{min} is the minimum frequency of a pure acoustic mode: i.e $\sim 100\ Hz$ for mostly all problems of thermoacoustics.

- **Spectrum**

  The spectrum parameter refers to the part of the eigenvalue spectrum that the solver is targeting for computation. In the case of the Helmholtz equation, this spectrum corresponds to the smallest positive eigenvalues.

Note that the use of a preconditionner eases the resolutions of problems and Tab. 3.4 illustrates such gains in terms of computational time for different numbers of vertices of a mesh. For the three cases, the solver tolerance is set to 10^{-5} and the number of requested eigenvalues is 10.

Number of vertices	CPU Time $[s]$	
	no preconditioner	shift-invert
3×10^4	1.1×10^4	18
2×10^5	4.3×10^5	187
1.5×10^6	$> 1.6 \times 10^6$	1512

Table 3.4: CPU time for computing 10 modes on different meshes, when using the shift-invert preconditioner or not. Three meshes with different number of vertices have been tested. The simulations are performed on 36 processors.

3.8.2 Frame size and mesh resolution

To properly capture a wave-like function on a mesh, a general rule of thumb is to have at least 10 mesh points per wavelength for the mode in question. This ensures that the function is sufficiently resolved. In the case of a frame $\boldsymbol{\phi} = [\phi_1 \cdots \phi_N]$, the mesh that is needed to evaluate $\boldsymbol{\phi}$ in a discrete form should be refined such that there are at least 10 mesh points within the smallest wavelength, corresponding to the highest frequency f_{max} and associated to mode ϕ_N. Additionally, for a given subdomain Ω, the frame size N must be set large enough to properly decompose the pressure field $p' = \sum_{n=1}^{N} \dot{\Gamma}_n \phi_n$. Effectively, this indicates that the discretization quality actually depends on the value of N, and the mesh size, as well as node distributions, should change with N.

In this context, we introduce a notion of consistency, noted $\underline{\boldsymbol{\varepsilon_c}}$, between a given mesh and the minimum wavelength defined at each point of the mesh:

$$\underline{\boldsymbol{\varepsilon_c}} = \frac{\underline{\boldsymbol{c_0}}}{f_{max} \sqrt[n]{\underline{\boldsymbol{V_D}}}} \tag{3.56}$$

where the underline $\underline{\cdot}$ denotes an array that contains values at each point of the mesh. $\underline{\boldsymbol{c_0}}$ is the array of sound speed, $\underline{\boldsymbol{V_D}}$ is the dual volume of nodes and n is the geometrical dimension of the mesh. In 2D, the dual volume could be interpreted as the area of a polygon formed by the centroids of the neighboring cells, as shown in Fig. 3.13. In 3D, this would correspond to the volume of a polytope formed by the centroids of the adjacent cells. Note that $\sqrt[n]{\underline{\boldsymbol{V_D}}}$ is an estimate of the local resolution and thus $\underline{\boldsymbol{\varepsilon_c}}$ is a measure of the number of mesh points in the smallest local wavelength. As a general guideline, the mesh is considered sufficiently resolved if at least 95% of the mesh points satisfy the condition of having more than 10 points per wavelength.

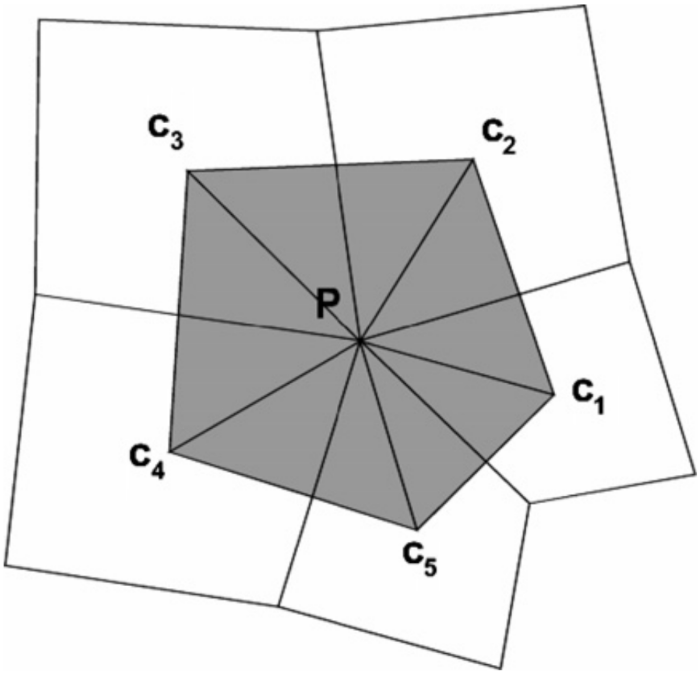


Figure 3.13: Two dimensional, unstructured, mesh with a dual volume associated with the node P. The centroid of neighbor elements are denoted by c_i. Taken from [153].

3.9 Conclusion

This chapter introduces the frame-based approach and details its implementation within the STORM solver. Beyond the general formulation, particular attention is given to addressing some intrinsic weaknesses of the method. It is shown that some modes in the initial frame are nearly linear combinations of others, which leads to difficulties when inverting the Gram matrix numerically. To overcome this problem, a truncated SVD method is introduced and validated on an academic configuration. This approach effectively "cleans" the initial frame, preventing numerical perturbations. It is now implemented in STORM for 1D, 2D, and 3D configurations.

Furthermore, the occurrence of non-physical modes associated with the fitting of transfer functions is analyzed, together with strategies to identify and remove them. These developments ensure that the frame expansion can be applied in practice while maintaining both robustness and physical relevance.

This chapter also introduces the FEniCS library, which is used to compute the initial frame by solving a linear eigenvalue problem (the homogeneous Helmholtz equation with trivial boundary conditions). Since the problem size can be relatively large ($\sim 10^6$ degrees of freedom), we rely on Krylov subspace methods for computing the eigenvalues and eigenvectors, combined with a shift-invert preconditioner to accelerate convergence.

By combining these elements, STORM now provides a robust and efficient framework for constructing and solving a well-conditioned thermoacoustic problem based on the frame approach.

The next chapter extends this methodology by incorporating additional types of boundary conditions into the modal expansion, enabling the treatment of increasingly complex thermoacoustic problems.

Chapter 4

Boundary and jump conditions

Contents

The purpose of this chapter is to present how various classical acoustic models such as impedance boundary conditions, pressure jump conditions and model components (e.g., injectors) can be reformulated within the state-space framework. While these models are well established in the thermoacoustic literature, their translation into a consistent matrix form is necessary for integration into the STORM solver. The standard boundary/jump conditions used in thermoacoustic modeling are the following:

- **Impedance boundary condition**

  Relates pressure and velocity fluctuations through an acoustic impedance. It is widely used to represent partially absorbing or reactive boundaries [74, 75, 78].

- **Multiperforated liners (MLPFs)**

  Commonly found in industrial combustion chambers, they act as acoustic dampers and can be modeled through pressure jump conditions across perforated plates [80, 154].

- **Acoustic Transfer Matrix (ATM)**

  Relates upstream and downstream acoustic variables across components like injectors. These are particularly valuable for integrating complex subsystems into global models [82, 83].

Accurate integration of these conditions is crucial in predicting the acoustic response of complex systems.

4.1 Impedance boundary condition

We begin with 1D constant impedance boundary conditions, which form the basis for many other models. This section recalls the definition of acoustic impedance and specifies the conventions adopted here, particularly regarding the orientation of the normal vectors. It also explains how impedance boundary condition can be modeled and incorporated within the state-space framework.

4.1.1 Convention and impedance definition

In this part, the acoustic quantities are supposed to be 1-dimensional planar waves, and the velocity is aligned to the x-axis: $\boldsymbol{u'_x} = u'_x \boldsymbol{e_x}$. Pressure fluctuations are noted p'. The value of the density and sound speed at the position $x = x_0$ are noted ρ_0 and c_0, respectively.

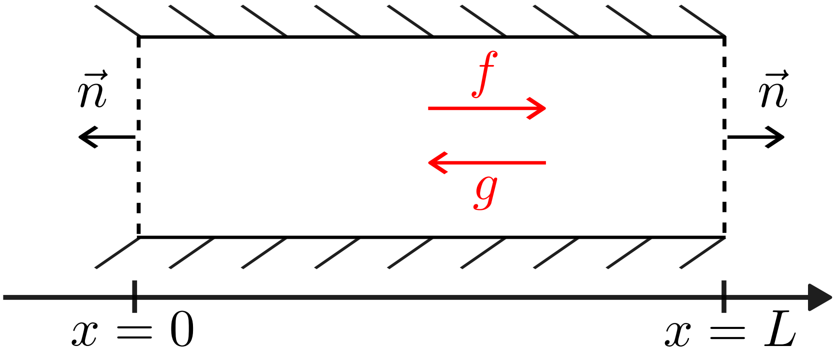


Figure 4.1: Scheme of a 1-dimensional duct of length L. f and g correspond to the forward/backward waves that propagate in the duct. In the presented method, the surface normals $\vec{n}$ are always pointing outward from the domain.

The impedance boundary condition is defined by the following relation:

$$\hat{p} = \rho_0 c_0 Z \hat{u}_n \tag{4.1}$$

where $\hat{u}_n = \hat{\boldsymbol{u}}_{\boldsymbol{x}} \cdot \boldsymbol{n}$ is the Fourier transform of the fluctuating velocity projected onto the outward pointing normal to the surface.

The presented low-order frame-based method involves acoustic perturbations of pressure and velocity, which directly describe the acoustic oscillations in the domain. Alternatively, some models use the Riemann invariants f and g, which represent traveling wave components moving in opposite directions. This formulation is particularly useful in problems involving characteristic methods or wave-based approaches [155, 156], as it naturally captures wave propagation and reflections at boundaries. Typically the reflection coefficient R is defined as the ratio between the reflected wave and the incoming one. In the scheme Fig. 4.1 the reflection coefficient is therefore defined differently at the inlet and the outlet of the domain:

$$\left\{ \begin{array}{rcll} R(x=0) & = & \dfrac{f}{g} & \text{at the inlet} \\ \\ R(x=L) & = & \dfrac{g}{f} & \text{at the outlet} \end{array} \right. \tag{4.2}$$

The pressure and velocity can be expressed by the f and g coefficients:

$$\left\{ \begin{array}{rcl} \dfrac{p'}{\rho_0 c_0} & = & f + g \\ \\ u'_x & = & f - g \end{array} \right. \tag{4.3}$$

Inversing this relation leads to the expression of the Riemann invariants:

$$\left\{ \begin{array}{rcl} f & = & \dfrac{1}{2}\left(\dfrac{p'}{\rho_0 c_0} + u'_x\right) \\ \\ g & = & \dfrac{1}{2}\left(\dfrac{p'}{\rho_0 c_0} - u'_x\right) \end{array} \right. \tag{4.4}$$

The point now is to express the relation between Z and R. Combining Eqs. 4.1 and 4.3 and using the outward pointing convention depicted in Fig. 4.1 leads to:

- **At the inlet:** $u'_n = -u'_x$

$$Z(x=0) = \frac{f+g}{g-f} = \frac{1+R(x=0)}{1-R(x=0)} \tag{4.5}$$

- **At the outlet:** $u'_n = u'_x$

$$Z(x=L) = \frac{f+g}{f-g} = \frac{1+R(x=L)}{1-R(x=L)} \tag{4.6}$$

The general formula involving the vector normal to the surface $\boldsymbol{n}$ and using the impedance definition of Eq. 4.1 is written:

$$Z = \frac{f+g}{f-g}\boldsymbol{e_x} \cdot \boldsymbol{n} \tag{4.7}$$

This equation shows that the direction of the normal to the surface $\boldsymbol{n}$ plays a role in the relation between Z and R. Three different conventions are possible, as detailed in the Tab. 4.1. For example, the convention for which $\boldsymbol{e_x}$ is always aligned with $\boldsymbol{u_n}$ is often utilized when dealing with 1-dimensional systems, as defined in Poinsot [73]. Note that, the reflection coefficient must always be defined as the ratio between the reflected wave over the incoming one.

Having established the definition and conventions for impedance boundary conditions, we now turn to their representation within the state-space framework. This translation is necessary to integrate the impedance conditions directly into the state-space formalism employed in STORM.

Table 4.1: Link between the impedance Z and the reflection coefficient R depending on the conventions used for the normal vector n to the surface.

Normal direction	Outward pointing	Inward pointing	$\boldsymbol{e_x}$
Inlet vectors	$u'_n = -u'_x$	$u'_n = u'_x$	$u'_n = u'_x$
Outlet vectors	$u'_n = u'_x$	$u'_n = -u'_x$	$u'_n = u'_x$
Inlet relation	$Z = \dfrac{1+R}{1-R}$	$Z = \dfrac{R+1}{R-1}$	$Z = \dfrac{R+1}{R-1}$
Outlet relation	$Z = \dfrac{1+R}{1-R}$	$Z = \dfrac{R+1}{R-1}$	$Z = \dfrac{1+R}{1-R}$

4.1.2 Constant impedance

As a first step, we focus on the case of a constant impedance, which offers a simple way to validate the numerical implementation. While it does not account for frequency-dependent effects, this assumption allows us to verify the consistency of the approach and to build the appropriate state-space.

The impedance boundary condition is typically defined in the complex frequency domain (see Eq. 4.1), as it originates from the Helmholtz formulation. In contrast, the state-space framework operates in the time domain and describes the physical dynamics of the system. Bridging these two representations requires a careful translation from the frequency-domain boundary condition to an equivalent time-domain formulation. A practical way to achieve this is to consider a small control volume around the boundary, which allows us to derive a dynamic relationship between pressure and velocity fluctuations, compatible with the state-space formulation. In this way, we denote by C the connection element, and we consider a small control volume of width L_t that encloses the boundary, as illustrated in Fig. 4.2.

To impose the boundary condition given in Eq. 4.1, we consider that the control volume receives the following two quantities: p^{Ω_1} and $\rho_0 c_0 Z u_n^{\Omega_1}$, which arise from the geometrical subdomain Ω_1. Under the assumption of zero Mach number, the Linearized Euler Equations (LEEs) are averaged over the control volume C:

$$\overline{\frac{\partial \boldsymbol{u}}{\partial t}} = -\frac{1}{\rho_0}\overline{\boldsymbol{\nabla p}} \tag{4.8}$$

The averaging operator $\bar{\cdot}$ is defined as the integral over the control volume and is written for any function f:

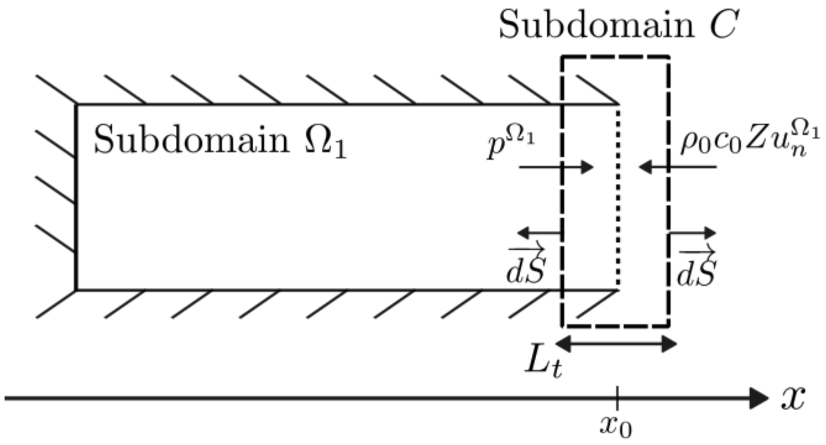


Figure 4.2: Schematic representation of a geometrical subdomain Ω_1 connected to a connection subdomain C. The control volume of length L_t that encloses the boundary is represented by the thick dashed lines. The thin dashed line denotes the boundary where the impedance must be imposed.

$$\overline{f} = \frac{1}{V_c} \iiint_C f \mathrm{d}V \tag{4.9}$$

Since the impedance is defined in the Fourier space, Eq. 4.8 is rewritten in that domain:

$$j\omega \overline{\hat{\boldsymbol{u}}} = -\frac{1}{\rho_0} \overline{\boldsymbol{\nabla} \hat{\boldsymbol{p}}} \tag{4.10}$$

Assuming that ρ_0 is uniform in the control volume, Eq. 4.10 becomes:

$$j\omega \overline{\hat{\boldsymbol{u}}} = -\frac{1}{\rho_0 V_c} \iiint_C \boldsymbol{\nabla} \hat{\boldsymbol{p}} \, \mathrm{d}V \tag{4.11}$$

Using the Green-Ostrogradski theorem leads to:

$$j\omega \overline{\hat{\boldsymbol{u}}} = -\frac{1}{\rho_0 V_c} \oiint_{\partial C} \hat{p} \, \mathbf{d}\boldsymbol{S} \tag{4.12}$$

where $\mathbf{d}\boldsymbol{S}$ is the outward pointing surface vector. The closed integral term represents the fluxes entering or leaving the control volume (see Fig. 4.2), which originate from the subdomain Ω_1. This equation is then projected onto the x-axis (normal to the boundary), and is written:

$$j\omega\overline{\hat{u}_x} = -\frac{1}{\rho_0 L_t}\left(-\hat{p}(x = x_0 - L_t/2) - \hat{p}(x = x_0 + L_t/2)\right) \tag{4.13}$$

which leads to:

$$j\omega\overline{\hat{u}_x} = \frac{1}{\rho_0 L_t}\hat{p}^{\Omega_1} - \frac{c_0 Z}{L_t}\hat{u}_n^{\Omega_1} \tag{4.14}$$

Note that replacing one of the two boundary pressure values with the impedance condition yields the same global results. Here, we choose to apply the impedance condition at the right boundary. This invariance arises because, at low frequencies (i.e., $\omega << c_0/L_t$), equation 4.14 simplifies to the quasi-static expression $\hat{p}^{\Omega_1} - \rho_0 c_0 Z \hat{u}_n^{\Omega_1} = 0$ which is exactly the impedance boundary condition to impose. Therefore, by choosing a sufficiently small value for L_t, for instance $L_t = 10^{-4}$, the correct acoustic boundary condition is enforced for the thermoacoustic frequencies of interest (typically until ~ 10 kHz).

Recasting this equation in the time-domain allows us to write the state-space realization of the constant impedance subdomain, where the dynamic equation of the form $\dot{\boldsymbol{X}}(\boldsymbol{t}) = \boldsymbol{A}\boldsymbol{X}(\boldsymbol{t}) + \boldsymbol{B}\boldsymbol{U}(\boldsymbol{t})$ is written:

$$\frac{d}{dt}\left[\ \overline{u_x}\ \right] = \left[\ 0\ \right]\left[\ \overline{u_x}\ \right] + \left[\begin{array}{cc} -\dfrac{c_0 Z}{L_t} & \dfrac{1}{\rho_0 L_t} \end{array}\right]\left[\begin{array}{c} u_n^{\Omega_1} \\ p^{\Omega_1} \end{array}\right] \tag{4.15}$$

and the output equation of the form $\boldsymbol{Y}(\boldsymbol{t}) = \boldsymbol{C}\boldsymbol{X}(\boldsymbol{t})$ is written:

$$\left[\ \overline{u_x}\ \right] = \left[\ 1\ \right]\left[\ \overline{u_x}\ \right] \tag{4.16}$$

This state-space is the one used for the case studied in Section 3.4. The state variable $\overline{u_x}(t)$ is the averaged velocity in the normal direction of the boundary, which is defined only in the control volume. The input variables $p^{\Omega_1}(t)$ and $u_n^{\Omega_1}(t)$ are the pressure and normal velocity fluctuations evaluated through their modal expansion at the boundary, and correspond to the output variables of the state-space of the geometrical subdomain Ω_1.

4.1.3 Frequency dependent impedance

In general cases, impedance depends on frequency because the physical phenomena that affect the absorption, reflection, and transmission of acoustic waves vary with frequency. This dependence arises from several factors such as the dissipation and visco-thermal losses at the walls. Viscous and thermal conduction effects influence how acoustic energy is dissipated [157, 158], and these mechanisms are inherently frequency dependent. In addition, the properties of the materials used for acoustic damping contribute to this dependence. Such materials often exhibit dispersive behavior [159, 160], meaning that their acoustic response—and thus their impedance—changes with frequency.

As a result, in realistic thermoacoustic models, the impedance is often represented as a complex function of frequency $Z = Z(\omega)$ to better capture the dynamics of thermoacoustic instabilities. However, Equation 4.15, which provides the state-space formulation for a constant impedance, is no longer valid when Z is frequency-dependent. This is because a frequency-dependent function can only be expressed in state-space form if it can be written as a rational function of ω, i.e., as a ratio of polynomials. To enable this, the impedance function $Z(\omega)$ is approximated using the Pole-Residue formulation —similar to the approach used for modeling flame transfer functions— using the Vector Fitting algorithm. This method constructs a rational approximation of $Z(\omega)$ suitable for conversion into a state-space representation.

The detailed state-space formulation of the corresponding subdomain is given in Appendix A.6.

4.2 3D-3D spectral connections

The second type of boundary condition addressed in this chapter differs from impedance conditions, as it does not apply at the outer boundary of a domain but rather at the interface between two geometrical subdomains. This type of condition, more precisely, a jump condition, is motivated by configurations found in real combustion chambers, where multiperforated plates separate the hot (combustion) and cold zones. These plates induce a jump in the acoustic pressure across the interface. In the STORM framework, such conditions are modeled using spectral connections, which connect two subdomains via a shared surface and allow to impose general acoustic jump relations between upstream and downstream quantities.

4.2.1 Modeling

Two 3D subdomains can be connected to each other through a connection interface S_c that is a 2-dimensional manifold. It is demonstrated in [94] that a spatial discretization of the surface gives some oscillations in the 3D shape of the global output modes. To fix this issue, a spectral discretization is preferred as it provides a more robust method, less sensitive to numerical approximations. To do so, we define the 2D acoustic potential over the surface $\varphi(\boldsymbol{x_s}, t)$ and the normal velocity to the surface $u_n(\boldsymbol{x_s}, t)$. These quantities are decomposed onto a 2D spectral basis $(\mathcal{K}_k)_{k\in\mathbb{N}}$ that satisfies the curvilinear Helmholtz equation on the surface manifold, which is written:

$$\boldsymbol{\nabla_c}\left(\frac{1}{\rho_0}\boldsymbol{\nabla_c}\mathcal{K}_k\right) + \frac{\omega_k^2}{\gamma p_0}\mathcal{K}_k = 0 \tag{4.17}$$

where $\boldsymbol{\nabla_c}$ is the curvilinear gradient operator over the surface manifold. The 2D acoustic potential and the normal velocity are decomposed onto the basis formed by the functions $\mathcal{K}_k$, truncated to a finite number of elements N_K:

$$\left\{\begin{array}{rcl} \varphi(\boldsymbol{x_s},t) & = & \displaystyle\sum_{k=1}^{N_K}\nu_k(t)\mathcal{K}_k(\boldsymbol{x_s}) \\ u_n(\boldsymbol{x_s},t) & = & \displaystyle\sum_{k=1}^{N_K}\mu_k(t)\mathcal{K}_k(\boldsymbol{x_s}) \end{array}\right. \tag{4.18}$$

As for the geometrical subdomains, the goal here is to derive a differential equation for each modal amplitudes $\mu_k(t)$ and $\nu_k(t)$ in order to construct the corresponding state-space formulation. This derivation has already been carried out in [94]. Only the temporal evolution of modal amplitudes are recalled:

$$\left\{\begin{array}{rcl} \ddot{\nu}_k + \omega_k^2\nu_k & = & -\dfrac{\overline{c_0}^2}{\lambda_k L_t}\left(s_i\langle u_n^i|\mathcal{K}_k\rangle + s_j\langle u_n^j|\mathcal{K}_k\rangle\right) \\ \dot{\mu}_k & = & \dfrac{1}{\lambda_k L_t\overline{\rho_0}}\left(s_i\langle p^i|\mathcal{K}_k\rangle + s_j\langle p^j|\mathcal{K}_k\rangle\right) \end{array}\right. \tag{4.19}$$

The operator $\langle\cdot\rangle$ denotes the 2-dimensional inner product on the interface S_c, for which the family of spectral modes $(\mathcal{K}_k)_{k\in\mathbb{N}}$ is an orthogonal basis:

$$\langle\mathcal{K}_i|\mathcal{K}_j\rangle = \iint_{S_c}\mathcal{K}_i(\boldsymbol{x_s})\mathcal{K}_j(\boldsymbol{x_s})\mathbf{d}\boldsymbol{x_s} = 0,\ \forall i\neq j \tag{4.20}$$

The acoustic variables p^i, p^j, u_n^i and u_n^j are the pressure and normal velocity arising from the two connected subdomains Ω_i and Ω_j. They are evaluated at the connection interface using their modal expansion, which allows Eq. 4.19 to be rewritten in terms of modal quantities:

$$\left\{\begin{array}{rcl} \ddot{\nu}_k + \omega_k^2 \nu_k & = & -\dfrac{\overline{c_0}^2}{\lambda_k L_t}\left(s_i \displaystyle\sum_{n=1}^{N_i} - \left\langle \dfrac{\nabla_n U_n^i}{\rho_0^i} \middle| \mathcal{K}_k \right\rangle \Upsilon_n^i + s_j \sum_{n=1}^{N_j} - \left\langle \dfrac{\nabla_n U_n^j}{\rho_0^j} \middle| \mathcal{K}_k \right\rangle \Upsilon_n^j \right) \\ \dot{\mu}_k & = & \dfrac{1}{\lambda_k L_t \overline{\rho_0}} \left(s_i \displaystyle\sum_{n=1}^{N_i} \langle U_n^i | \mathcal{K}_k \rangle \dot{\Upsilon}_n^i + s_j \sum_{n=1}^{N_j} \langle U_n^j | \mathcal{K}_k \rangle \dot{\Upsilon}_n^j \right) \end{array}\right. \tag{4.21}$$

In these equations, the superscripts i and j refer to the two connected 3D subdomains Ω_i and Ω_j. Their orientation with respect to the surface are defined by s_i and s_j that satisfy $s_i = -s_j = \pm 1$. This value is automatically determined when the network is constructed: for each spectral connection, the user specifies the two connected geometrical domains. The first domain listed by the user is assigned $s_i = +1$, while the second is assigned $s_j = -1$. λ_k is the norm of the k^{th} spectral mode, $\lambda_k = \langle \mathcal{K}_k | \mathcal{K}_k \rangle$, $\overline{c_0}$ and $\overline{\rho_0}$ are the mean sound speed and density at the interface. As in the case of impedance modeling, the length L_t here represents the thickness of a small control volume that surrounds the surface defining the interface between the two geometrical subdomain subdomains. The state-space representation of Eq. 4.21 is given in Appendix A.3.

4.2.2 Selection Algorithm

As explained in [94], not all components $\mathcal{K}_k$ are required to accurately project the 3D modes U_n. Using a too large basis $(\mathcal{K}_k)_k$ may introduce non-physical oscillations in the output. The selection algorithm described in [94] enables the identification of a reduced subset of surface modes that are sufficient to represent all projections of the 3D vectors U_n on the interface. Once the solutions to the curvilinear Helmholtz eigenproblem in Eq.4.17 are obtained —either analytically or with a FEM solver— a finite number N_K of surface modes must be retained to construct the state-space representation of the interface. This selection ensures an accurate representation of the acoustic pressure and velocity at the boundary of the 3D subdomain. As shown in Eq. 4.19, the interface pressure $p(\boldsymbol{x_s})$ and velocity $\boldsymbol{u(x_s)}$ are evaluated by projecting the restriction of the 3D modal basis $U_n(\boldsymbol{x_s})$ onto the selected surface modes $\mathcal{K}_k(\boldsymbol{x_s})$.

$$\left\{\begin{array}{rcl} U_n(\boldsymbol{x_s}) & = & \displaystyle\sum_{k=1}^{N_K} \langle U_n|\mathcal{K}_k\rangle \mathcal{K}_k(\boldsymbol{x_s}) \\ \\ \boldsymbol{\nabla U_n}(\boldsymbol{x_s}) & = & \displaystyle\sum_{k=1}^{N_K} \langle \boldsymbol{\nabla U_n}|\mathcal{K}_k\rangle \mathcal{K}_k(\boldsymbol{x_s}) \end{array}\right. \tag{4.22}$$

where $\langle \cdot|\cdot \rangle$ is the 2D inner product over the connection interface (see Eq. 4.20). The selection algorithm is then built to construct automatically and robustly the optimal surface basis $(\mathcal{K}_k)_k$. It is based on the Singular Value Decomposition of the matrix defined as the inner products between the 3D modal basis and the 2D surface modes $(\langle U_n|\mathcal{K}_k\rangle)_{n,k}$. It is detailed in Appendix C. in [94].

4.2.3 Practical implementation

Equation 4.21 is easy to write on paper, but computing the inner products between the restriction of the U_n modes at the interface and the spectral modes $\mathcal{K}_k$ require some adaptations. To build this corresponding connection, two 3D meshes $\mathcal{M}_i$ and $\mathcal{M}_j$ are necessary, which defines the two 3D subdomains that must be connected. The connection interface is therefore defined by two boundary patches $\mathcal{M}_i^b$ and $\mathcal{M}_j^b$ from the two 3D meshes. Theoretically, these two surface meshes $\mathcal{M}_i^b$ and $\mathcal{M}_j^b$ represent the same manifold, but numerically they are two distinct surfaces. Consequently, two distinct basis of spectral modes are computed on each interface mesh $\mathcal{M}_i^b$ and $\mathcal{M}_i^b$, respectively denoted $(\mathcal{K}_k^i)_{k\geq 1}$ and $(\mathcal{K}_k^j)_{k\geq 1}$. The amplitude coefficients of the acoustic potential and velocity are also associated with a mesh, so we denote by μ_k^i, ν_k^i and μ_k^j, ν_k^j these coefficients, respectively for the meshes $\mathcal{M}_i^b$ and $\mathcal{M}_j^b$. Theoretically, μ_k^i must equal μ_k^j (same for ν_k) so the numerical treatments detailed in this section will impose this equality.

To understand how the Eq. 4.21 is computed, we take the example of μ_k but the same treatments are also applied to the equation on ν_k. The temporal evolution of μ_k is rewritten to highlight the meshes on which the inner products are numerically computed.

$$\dot{\mu}_k^i = \dot{\mu}_k^j = \frac{1}{\lambda_k L_t \overline{\rho_0}} \left[s_i \sum_n^{N_i} \dot{\Upsilon}_n^i \underbrace{\langle U_n^i | \mathcal{K}_k^i \rangle}_{\text{Same mesh}} + s_j \sum_n^{N_j} \dot{\Upsilon}_n^j \underbrace{\langle U_n^j | \mathcal{K}_k^j \rangle}_{\text{Same mesh}} \right] \tag{4.23}$$

This equation shows that the inner products are well defined on each separate mesh and are computed with the classical first order finite element approximation:

$$\langle U_n^i | \mathcal{K}_k^i \rangle = \int_{S_c} U_n^i(\boldsymbol{x}) \mathcal{K}_k^i(\boldsymbol{x}) \mathbf{d}\boldsymbol{x} = (\underline{\boldsymbol{U_n^i}})^{\boldsymbol{T}} \underline{\boldsymbol{M_i^b}} \underline{\boldsymbol{\mathcal{K}_k^i}} \tag{4.24}$$

where $\underline{\boldsymbol{M_i^b}}$ is the mass matrix associated with the 2-dimensional mesh $\mathcal{M}_i^b$ representing the interface manifold. $\underline{\boldsymbol{U_n^i}}$ and $\underline{\boldsymbol{\mathcal{K}_k^i}}$ are the numerical arrays containing the values of the functions U_n^i and $\mathcal{K}_k^i$, respectively. In some cases where the connection interface is a non-degenerate surface, the two surface basis $(\mathcal{K}_k^i)_k$ and $(\mathcal{K}_k^j)_k$ are exactly the same (same frequencies and shapes) and no further numerical treatments are necessary. However, it is not rare that the surface interface is degenerate as it is the case for the case depicted in Fig. 4.3, the spectral modes $\mathcal{K}_k^i$ and $\mathcal{K}_k^j$ can be rotated by a given angle that can depend on the mode index k. However, only a single spectral basis is supposed to describe the interface. So, let us assume that we choose the basis $\mathcal{K}^i_k$ to describe the interface. The equation on μ^i_k is then written:

$$\dot{\mu}^i_k = \frac{1}{\lambda_k L_t \overline{\rho_0}} \left[s_i \sum_n^{N_i} \dot{\Upsilon}^i_n \underbrace{\langle U^i_n | \mathcal{K}^i_k \rangle}_{\text{Same mesh}} + s_j \sum_n^{N_j} \dot{\Upsilon}^j_n \underbrace{\langle U^j_n | \mathcal{K}^i_k \rangle}_{\text{Different mesh}} \right] \tag{4.25}$$

In this case, the first inner product is well defined on the same mesh, but the second one is defined on the two different meshes $\mathcal{M}^b_i$ and $\mathcal{M}^b_j$. The goal here is to rewrite this equation to highlight the $\mathcal{K}^j_k$ modes, so that the second inner product is correctly computed on a single mesh: $\mathcal{M}^b_j$. The two sets of spectral modes $(\mathcal{K}^i_k)_k$ and $(\mathcal{K}^j_k)_k$ are linked by the following relation:

$$\mathcal{K}^i_k = \sum_l^{N_K} \langle \mathcal{K}^j_l | \mathcal{K}^i_k \rangle \mathcal{K}^j_l \tag{4.26}$$

Combining Eqs. 4.26 and 4.25 leads to:

$$\dot{\mu}^i_k = \frac{1}{L_t \lambda_k \overline{\rho_0}} \left[s_i \sum_n^{N_i} \dot{\Upsilon}^i_n \underbrace{\langle U^i_n | \mathcal{K}^i_k \rangle}_{\text{Same mesh}} + \sum_l^{N_K} \left(\underbrace{\langle \mathcal{K}^j_l | \mathcal{K}^i_k \rangle}_{\text{Different mesh}} \; s_j \sum_n^{N_j} \dot{\Upsilon}^j_n \underbrace{\langle U^j_n | \mathcal{K}^j_l \rangle}_{\text{Same mesh}} \right) \right] \tag{4.27}$$

This equation demonstrates that even though the two families $(\mathcal{K}^i_k)_{k \leq N_K}$ and $(\mathcal{K}^j_k)_{k \leq N_K}$ originate from different meshes, the scalar products with their respective 3D modes can still be computed on the same meshes. However, to establish a connection between the two domains, it is necessary to compute by extrapolation the scalar products between the two families of 2D modes: $\langle \mathcal{K}^j_l | \mathcal{K}^i_k \rangle$. These inner products are gathered in the matching matrix, defined as $\boldsymbol{M_m}[k, l] = \langle \mathcal{K}^j_l | \mathcal{K}^i_k \rangle$. There are some points to note here:

- In the case of a non-degenerate connection, $\mathcal{K}^j_k = \mathcal{K}^i_k$, so $\langle \mathcal{K}^j_l | \mathcal{K}^i_k \rangle = \delta_{l,k}$ and Eq. 4.27 leads to the previous equation (4.23).

- Actually, even for non-degenerate connection, $\mathcal{K}_k^i$ might be equal to $-\mathcal{K}_k^j$ (reversed), so $\langle\mathcal{K}_l^j|\mathcal{K}_k^i\rangle = -\delta_{lk}$. This new formula takes into account this change of sign.
- For all degenerate surface connections, $\langle\mathcal{K}_l^j|\mathcal{K}_k^i\rangle \in [-1, 1]$. The matching matrix therefore corresponds to a rotation matrix and satisfies the orthogonality relation: $\boldsymbol{M_m^{-1}} = \boldsymbol{M_m^T}$.
- Errors that appear from numerical extrapolation are reduced because only the surface modes are extrapolated, and in practical cases the number of spectral modes is lower than the number of 3D modes in the two connected geometrical subdomains: $N_K < N_i$ or N_j.
- The two basis $(\mathcal{K}_k^i)_{k\leq N_K}$ and $(\mathcal{K}_k^j)_{k\leq N_K}$ are required to span (approximately) the same subspace. In addition, the vectors of the two basis must be ordered consistently in terms of frequency, since no processing is applied to the pulsation values. This is equivalent to assuming that $\forall k \in [\![0, N_K]\!]$, $\omega_k^i = \omega_k^j$.

The state-space realization of the 3D3D spectral connection element is given in Appendix A.3.

4.3 3D3D connections - Validation

To study the robustness of the 3D3D spectral connection modeling, this formalism is tested on different simple cases that cover a wide range of industrial configurations. To do so, the geometry depicted in Fig. 4.3 is used for all the cases where different mean fields are set. The mean fields considered in the course of this study are as follows:

- Uniform over the whole geometry.

- Uniform with a temperature jump at the interface.
- Non-uniform without temperature jump.
- Non-uniform with a temperature jump.

Case 4 typically mimics an industrial post-combustion configuration (see the real case in Sec. 5.4). The first three cases are designed to verify, step by step, that each physical effect, such as non-uniformity and temperature discontinuity, is correctly captured by the model.

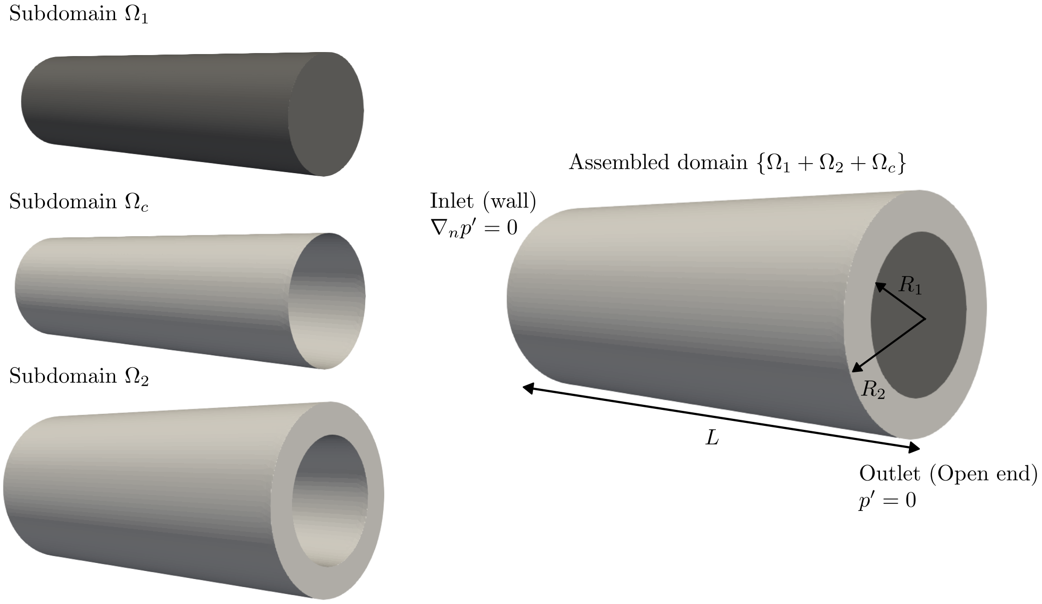


Figure 4.3: 3-dimensional configuration used to validate the surface modal expansion method. Two geometrical subdomains Ω_1 and Ω_2 are connected through a connection subdomain Ω_c.

Still in the objective of approaching industrial configurations, the geometric dimensions of the test cases are chosen as follows: $R_1 = 22.90$ mm, $R_2 = 34.16$ mm and $L = 156$ mm. The boundary conditions are set to an open atmosphere ($p' = 0$) at the outlet, i.e right side of the ducts, and a wall ($\nabla_n p' = 0$) at all the other boundaries.

4.3.1 Uniform fields

Here we consider the case depicted in Fig. 4.3. For the first test case we assume that the temperature is uniform and constant over the whole geometry. The value of the mean pressure, mean sound speed and mean density are set to $p_0 = 101300\ Pa$, $c_0 = 347.12\ m.s^{-1}$ and $\rho_0 = 1.177\ kg.m^{-3}$. The reference solutions are obtained by calculating the acoustic modes with FEniCS, on a single domain that represents the same geometry as the assembled system.

For the network-based approach, two frames are computed (one for each subdomain), and the SVD methodology detailed in Sec. 3.5.1 is applied in both the subdomains Ω_1 and Ω_2. The formalism of spectral connection with the matching matrix correction detailed in Sec. 4.2.3 is used to model the connection interface. To compute the 2-dimensional basis $(\mathcal{K}_k)_{k \leq N_K}$, the first step is to extract the two manifolds from the two 3D subdomains. The two boundary meshes $\mathcal{M}_1^b$ and $\mathcal{M}_2^b$ are shown in Fig. 4.4.

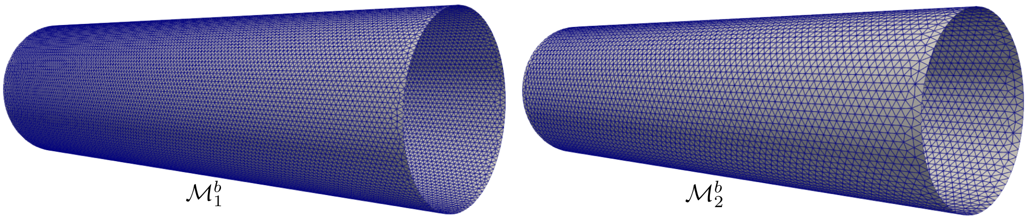


Figure 4.4: The two boundary meshes $\mathcal{M}_1^b$ and $\mathcal{M}_2^b$ that define the connection interface, where the two surface modal basis are computed.

Then the two surface modal bases $(\mathcal{K}_k^{(1)})$ and $(\mathcal{K}_k^{(2)})$ are computed respectively on $\mathcal{M}_1^b$ and $\mathcal{M}_2^b$. As explained in the previous section, and as illustrated in Fig. 4.5, the 2D acoustic modes of the surface can be degenerate.

It is noted that the 2D modes were computed with the same boundary condition (Dirichlet) on the right side, in order to comply with the boundary condition of the global system. As illustrated in Fig. 4.5, we observe that

`Figures/chap_BCs/degenerate_modes.pdf`

Figure 4.5: Examples of behaviors that can occur when computing the surface modal basis. Mode 5: longitudinal mode that is reversed from one mesh to the other. Mode 6 and 7: example of rotated modes which are not aligned between the two meshes.

modes 6 and 7 are rotated by a certain angle θ, and mode 5 is inverted. The matching matrix, which represents the link between these two basis, therefore takes the following form for its indices 5 to 7:

$$(M_m)_{5\leq k,l\leq 7} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) \\ 0 & -\sin(\theta) & \cos(\theta) \end{bmatrix} \tag{4.28}$$

The inner products between two quantities defined on different meshes are computed by linearly extrapolating the values from the mesh $\mathcal{M}_2^b$ onto the mesh $\mathcal{M}_1^b$. The FEniCS script used to perform this extrapolation is provided in Appendix D.2.1.

For this simulation, we used $N_1 = 100$ modes in the frame of the subdomain Ω_1 and $N_2 = 100$ modes for the frame of the subdomain Ω_2. We set the number of spectral modes to $N_K = 100$, but the selection algorithm only retains 56 modes, which are not necessarily the 56 first modes. The results of the first 12 frequencies of this configuration are presented in the Tab. 4.2.

Table 4.2: Frequencies of the case depicted in Fig. 4.3 using uniform fields. The mean relative error is computed using Eq. 4.29, with $N_{modes} = 12$.

Mode index	Reference solution [Hz]	Storm output [Hz]	Relative errors [%]
1	556	554	0.36
2	1669	1663	0.36
3	2784	2773	0.40
4	3033	3079	1.52
5	3033	3080	1.55
6	3419	3456	1.08
7	3419	3457	1.11
8	3901	3883	0.46
9	4084	4107	0.56
10	4084	4108	0.59
11	4917	4926	0.18
12	4917	4928	0.22
Mean relative error [%]			0.70

The results are very close to the reference values, with a mean relative error (MRE) of 0.7 % on the frequencies for the first 12 modes. To study the importance of the selection algorithm, different simulations have been performed, using different number of spectral modes from $N_K = 10$ up to $N_K = 100$ with a step of 10. When no selection algorithm is applied, the spectral modes are chosen according to the lowest N_K eigenfrequencies in ascending order. When the selection algorithm is used, it identifies the most relevant modes regardless of their frequency ranking. The Fig. 4.6 shows the mean relative error of the 12 first modes depending on N_K, computed as:

$$MRE = \frac{1}{N_{modes}} \sum_{n=1}^{N_{modes}} \left| \frac{f_{target,n} - f_{storm,n}}{f_{target,n}} \right| \tag{4.29}$$

where f_{target} and f_{storm} are the reference frequency and the the one returned by STORM.

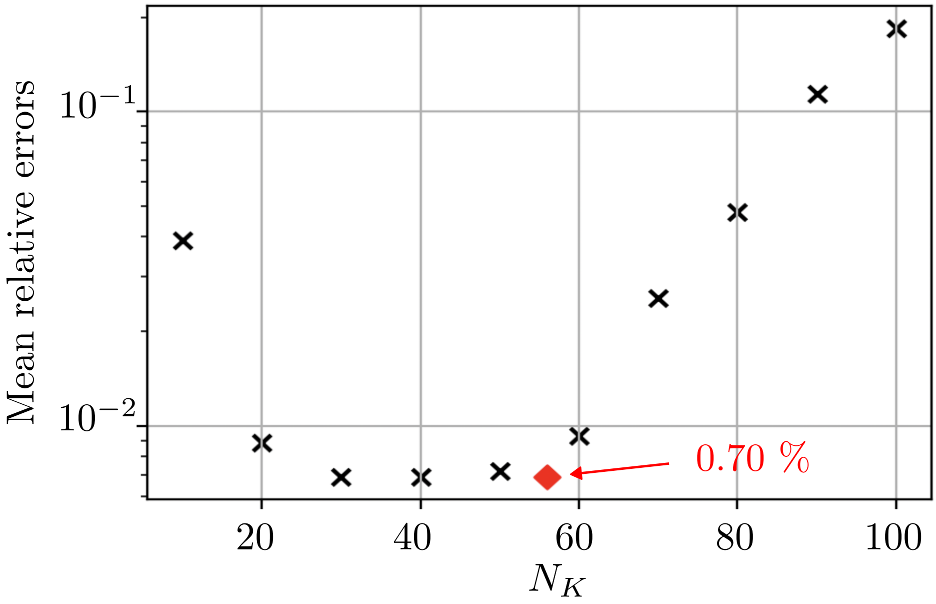


Figure 4.6: Black crosses: mean relative errors on the first 12 modes as a function of the number of spectral modes used for the connection, without applying the selection algorithm. The red diamond is the result when the selection algorithm is applied starting from an initial set of $N_K = 100$ modes, of which 56 are retained.

Note that for $N_K = 30$ 40 or 50, the mean relative error is also very low and thus comparable to those obtained from the simulation using the selection algorithm (red diamond in Fig. 4.6). It can be observed that for low values of N_K, increasing N_K reduces the error. However, this effect no longer holds for higher values of N_K, where adding more modes actually disturbs the overall result.

This section validated different key aspects: 1) the implementation of the 3D–3D connection state-space in the case of uniform mean fields, using the correction introduced by the matching matrix, and 2) the performance of the mode selection algorithm in this context.

4.3.2 Non-uniform fields

In a real combustion chamber, the temperature and sound speed fields are far from uniform. Typically, the gas density is low near the flame and increases as the gases cool down. To verify that the implementation remains efficient in this context, we introduce a quadratic profile for ρ_0 along the length of the tube, representing a temperature decrease. This profile is not based on any physical consideration; it is simply used to test the numerical method in the presence of a non-uniform field. The mean field ρ_0 used for this simulation is provided in Fig. 4.7.

For this simulation, we keep the same frame size that is $N_1 = 100$ in Ω_1 and $N_2 = 100$ in Ω_2. The number of spectral modes is set to $N_K = 100$ and the selection algorithm only retains 55 modes.

The results in terms of frequencies of the 12 first modes are presented in Tab. 4.3. In this case, the selection algorithm is used.

The frequencies match the reference results accurately, with low relative errors.

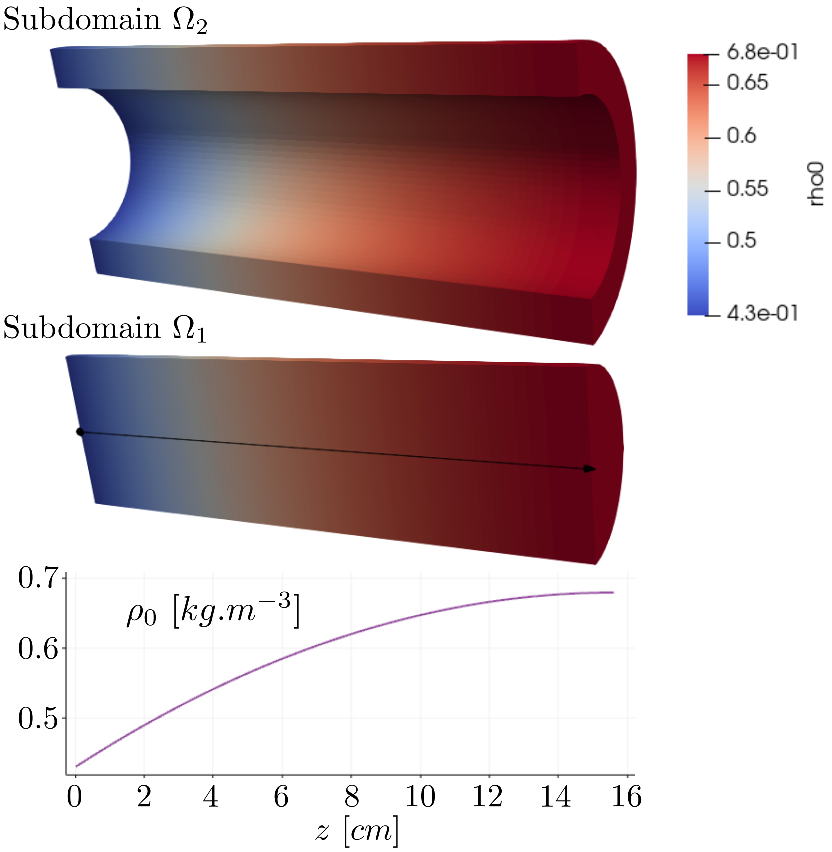


Figure 4.7: Quadratic profile of ρ_0 over both the subdomains Ω_1 and Ω_2. The graph in the last row is plotted along the black arrow shown in subdomain Ω_1.

Table 4.3: Frequencies of the case depicted in Fig. 4.3 using quadratic fields. The relative error is computed using Eq. 4.29.

Mode index	Reference solution [Hz]	Storm output [Hz]	Relative errors [%]
1	750	747	0.4
2	2335	2327	0.34
3	3911	3895	0.41
4	4351	4413	1.42
5	4351	4414	1.45
6	4902	4957	1.12
7	4902	4958	1.4
8	5487	5461	0.47
9	5775	5810	0.61
10	5775	5811	0.62
11	6934	6947	0.19
12	6934	6949	0.22
Mean relative error [%]			0.70

4.3.3 Temperature jump through the connection interface and uniform fields

A common feature of industrial combustion chambers is the use of multiperforated plates surrounding the chamber. Their purpose is to separate the hot gases inside the chamber from the fresh gases flowing around the plate. The small holes in the plate allow a thin film of fresh air to pass through, cooling the chamber. From a network modeling perspective, the system is divided into two subdomains: the inner domain, which contains hot gases, and the outer domain, where the fresh gases flow. In this context, a uniform mean field is applied, with low density (hot gases) in the inner domain and high density (fresh gases) in the outer domain. The mean field of ρ_0 used for this simulation is provided in Fig. 4.8.

For this simulation, the frame size is kept at $N_1 = 100$ in Ω_1 and $N_2 = 100$ in Ω_2. The number of spectral modes is also kept at $N_K = 100$ and the selection algorithm only retains 56 modes. The results are summarized in Tab. 4.4

We observe that the mean relative error remains very small, but it is 10 times greater than in the previous configuration without the temperature jump across the interface. This increase can be attributed to differences in the mean flow field: in the reference case, the full 3D domain is meshed, and there is no strict separation between the two subdomains, whereas in the test case the interface explicitly divides them. As a result, the mean density field (see Figure 4.8) is not defined exactly the same way in both configurations. Nevertheless, given the very low error levels, these results still validate the 3D3D connection method in the presence of a temperature and density discontinuity at the interface.

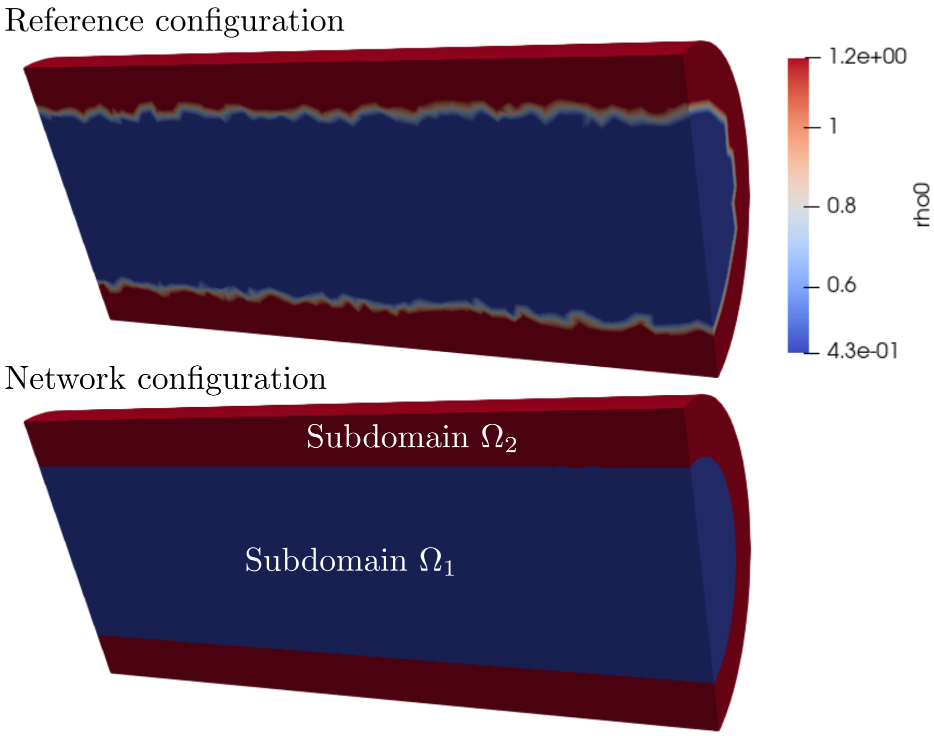


Figure 4.8: Mean fields with a temperature jump across the interface. The first figure is the reference configuration corresponding to a full 3D geometry. The second one corresponds to the network approach, where two distinct 3D subdomains are connected through the interface.

Influence of modal discretization in 3D subdomains

The modes 6 and 7 have a large relative frequency error compared to the reference case. This is not shown in Table 4.4, but if we look at the growth rates of these modes, they are not exactly zero in the STORM output: $\sigma_6 = 0.98$ s^{-1} and $\sigma_7 = -0.98$ s^{-1}. This is because these modes have a stronger radial component, which requires more modes in the frame to properly capture them. These two modes are degenerate — or more precisely, form a pair of complex conjugates — and the reason they do not exactly match the reference is the same for both. Therefore, we arbitrarily focus on mode 6 to

Table 4.4: Frequencies of the case depicted in Fig. 4.3 using a uniform temperature jump across the interface. The relative error is computed using Eq. 4.29.

Mode index	Reference solution [Hz]	Storm output [Hz]	Relative errors [%]
1	734	742	1.09
2	2193	2213	0.91
3	3614	3644	0.83
4	4239	4382	3.37
5	4239	4383	3.40
6	4612	4753	3.06
7	4612	4753	3.06
8	4983	5011	0.56
9	5273	5410	2.60
10	5274	5411	2.60
11	6114	6251	2.24
12	6116	6251	2.21
Mean relative error [%]			2.16

explain this.

The main phenomenon responsible for this error is the poor modal discretization of the geometric subdomains Ω_1 and Ω_2. As shown in Figure 4.9, when the number of modes in the geometrical domains increases, the error in the final mode 6 decreases. Note that the error displayed in the bottom row of the Fig. 4.9 accounts for the growth rate, which is particularly relevant for mode 6 as it has a non-null growth rate.

It is interesting to note that the convergence of the mode does not follow

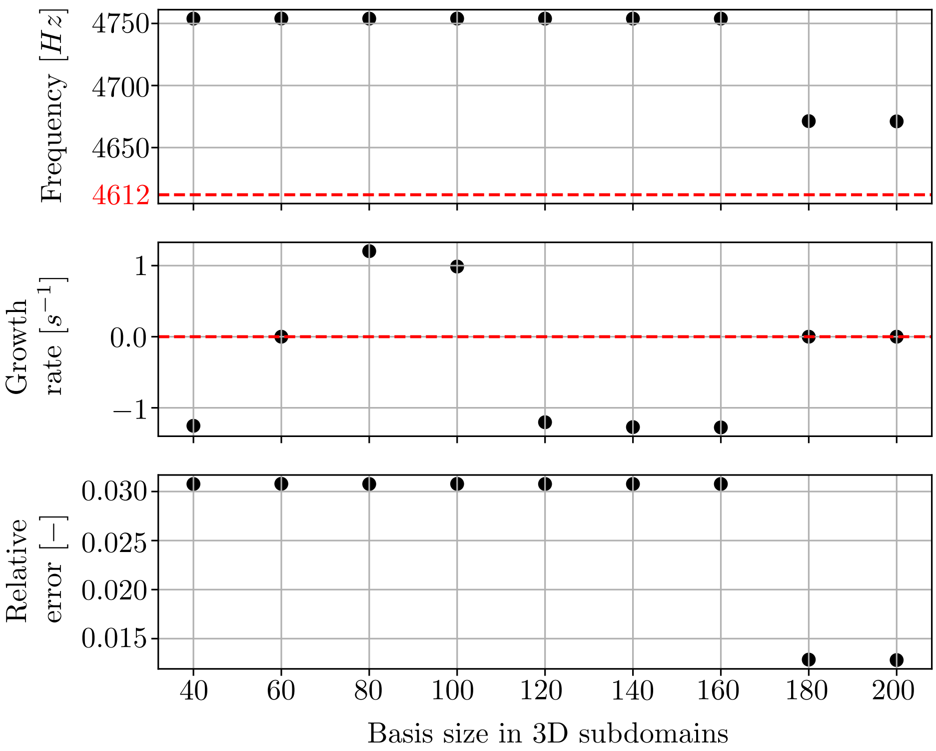


Figure 4.9: First two rows: Frequency and growth rate of the mode 6 in the configuration depicted in Fig. 4.3 using a uniform temperature jump across the interface. Bottom row: Relative error compared to the reference frequency, computed using Eq. 4.29, by including the growth rate contribution. Red dashed line: target frequency/growth rate.

any smooth or predictable trend. Instead, the solution abruptly converges once a certain frame size is reached, that is, $N_1 = N_2 = 180$. In fact, the mode in question (here, mode 6) requires a few specific components of the frame to be accurately captured. This behavior is clearly illustrated in Fig. 4.10.

Figure 4.10 shows the absolute value of modal amplitudes $A_n^{\Omega_i}$ of mode 6 at $\sim$ 4700 Hz, associated with each component of the frame $\phi_n^{\Omega_i}$, in each subdomain (Ω_1 and Ω_2). Recall that in each subdomain Ω_i, the pressure mode 6 is written: $p_6^{\Omega_i}(\boldsymbol{x}) = \sum_{n=1}^{N_i} A_n^{\Omega_i}\phi_n^{\Omega_i}(\boldsymbol{x})$. Actually, since the pressure

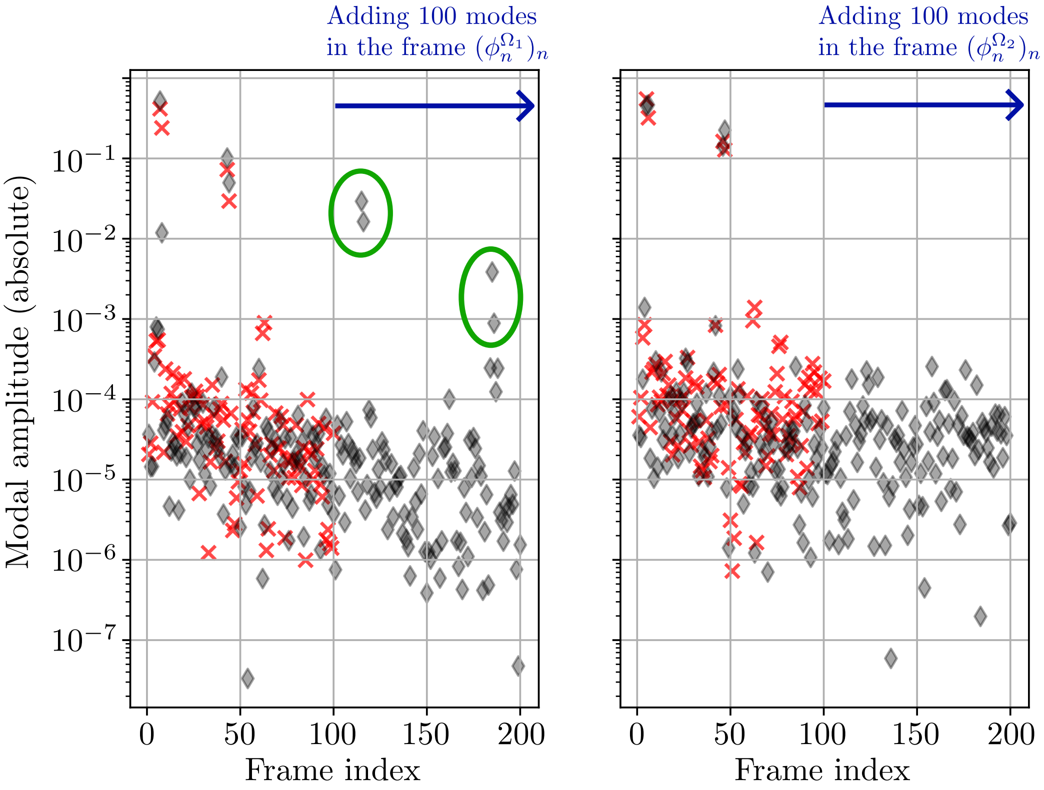


Figure 4.10: Left: absolute value of the modal amplitudes of the mode 6 at ~ 4700 Hz in the inner domain (Ω_1), associated with each component of the frame $\phi_n^{\Omega_1}$. Right: same for the outer domain (Ω_2). Two different frame size are tested, which are $N = 100$ (red crosses), and $N = 200$ (black diamonds).

field is decomposed onto an orthonormal basis of the frame (the U-basis, see Sec. 3.5.1), some processing of the output eigenvector is required to obtain the modal amplitudes associated with the initial frame components. This procedure is detailed later in the text.

In the Fig. 4.10, the components of each frame (with $N = 100$ or $N = 200$) are individually sorted in increasing order of frequency. We observed that the error previously seen on mode 6 comes from an insufficient number of components in the frame to accurately represent it. In particular, in domain Ω_1, increasing the frame size from 100 to 200 modes reveals that mode

6 requires higher-frequency components from the frame to be properly reconstructed. This is illustrated by the green-circled points on the graph, which correspond to non-negligible modal amplitudes associated with high-frequency modes. Interestingly, although mode 6 is around 4600 Hz, it relies on frame components at much higher frequencies, around $\sim$ 20000 Hz (frequency of frame components in the leftmost green circle). This is not an intuitive result, as one would typically expect a mode to be well represented by frame components of similar frequency.

Actually, the explanation lies not in the frequency content, but rather in the spatial structure of the modes. Mode 6 exhibits a strong radial component, and accurately capturing its shape requires basis modes that share similar spatial structure, regardless of their frequency. This is precisely what happens here: the high-frequency basis modes that contribute a lot to mode 6 (marked with a green circle in Fig. 4.10) also exhibit a strong radial structure, as illustrated in Fig. 4.11. Its spatial alignment with the target mode makes it an important component for the reconstruction, despite the large frequency gap.

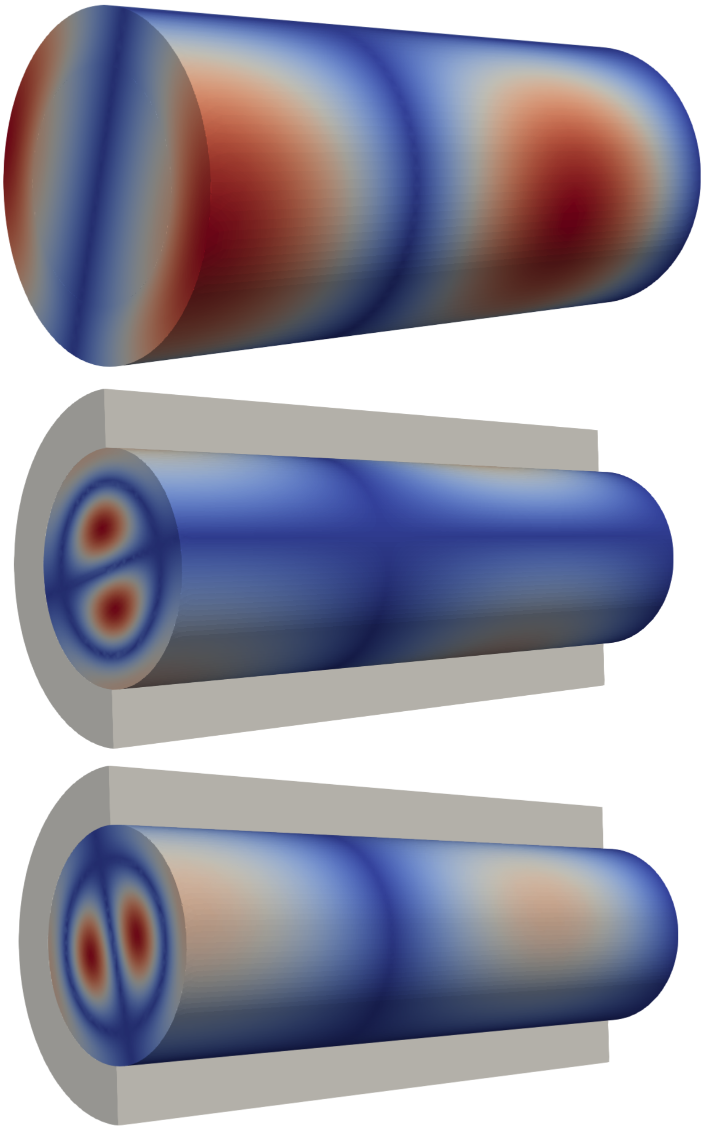


Figure 4.11: Spatial structure of mode 6 and contributing high-frequency basis modes with strong radial components. Top row: mode 6 ($\sim 4700\ Hz$), plotted over both subdomains to visualize the complete spatial structure of the global mode. Bottom rows: high-frequency basis modes ($\sim 20000\ Hz$), in subdomain Ω_1, where a strong radial component is required to discretize mode 6 accurately. This issue does not appear in subdomain 2, where the modal basis already captures the radial structure sufficiently with low-frequency basis modes. The selected modes correspond to the leftmost green circles in Fig. 4.10.

The note below details how to get the modal amplitudes $A_n^{\Omega_i}$ plotted in Fig. 4.10

How to get modal amplitudes on frame components ?

Due to the SVD that is performed in each geometrical subdomain (see section 3.5.1), the amplitude coefficients of mode 6 that are plotted in Fig. 4.10 are computed from the global solution eigenvector $\boldsymbol{\mathcal{V}_6}$. As 3 subdomains are present in the configuration, this global eigenvector can be noted by $\boldsymbol{\mathcal{V}_6} = \left[\boldsymbol{\mathcal{V}_6^{\Omega_1}}, \boldsymbol{\mathcal{V}_6^{\Omega_c}}, \boldsymbol{\mathcal{V}_6^{\Omega_2}}\right]$. The portion of this global eigenvector corresponding to the geometrical subdomain Ω_i contains all the information to build the pressure and the velocity shape of mode 6. If we note $\boldsymbol{\mathcal{V}_6^{\Omega_i}} = \left[a_1^{\Omega_i}, b_1^{\Omega_i}, \cdots, a_{N_i}^{\Omega_i}, b_{N_i}^{\Omega_i}\right]$, the pressure in Ω_i is built as follows:

$$p_6^{\Omega_i}(\boldsymbol{x}) = \sum_{n=1}^{N_i} b_n^{\Omega_i} U_n^{\Omega_i}(\boldsymbol{x}) = \boldsymbol{U^{\Omega_i} b^{\Omega_i}} \tag{4.30}$$

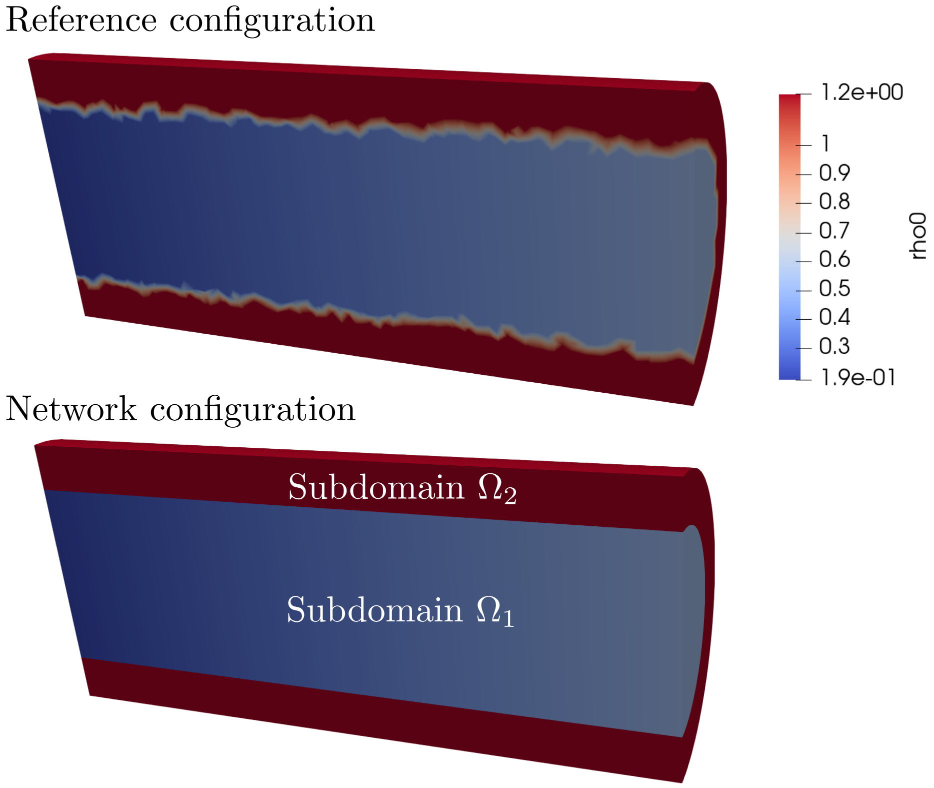
where $\boldsymbol{U^{\Omega_i}}$ is given by the SVD of the frame: $\boldsymbol{\phi^{\Omega_i}} = \boldsymbol{U^{\Omega_i} \Sigma^{\Omega_i} V^{T,\Omega_i}}$, and $\boldsymbol{b^{\Omega_i}}$ is the column vector that contains the coefficients $b_n^{\Omega_i}$. This allows us to express the modal decomposition of mode 6 in terms of the frame modes rather than the orthonormalized modes, the latter having little physical meaning, such as:

$$p_6^{\Omega_i}(\boldsymbol{x}) = \boldsymbol{\phi^{\Omega_i} V^{\Omega_i} (\Sigma^{\Omega_i})^{-1} b^{\Omega_i}} \tag{4.31}$$

In the same way that $\boldsymbol{b^{\Omega_i}}$ corresponds to the modal amplitudes of the decomposition on $\boldsymbol{U^{\Omega_i}}$, the product $\boldsymbol{V^{\Omega_i} (\Sigma^{\Omega_i})^{-1} b^{\Omega_i}}$ corresponds to the modal amplitudes of the decomposition on $\boldsymbol{\phi^{\Omega_i}}$.

4.3.4 Temperature jump through the connection interface and non-uniform fields

Building on the previous two sections, we focus on a test case that brings everything together: a non-uniform temperature field along the interface, combined with a sharp jump in temperature in the direction normal to it. This setup closely resembles the typical conditions encountered in post-combustion chambers. The mean fields used for this section are displayed in Fig. 4.12.


Reference configuration

Network configuration

Figure 4.12: Mean fields with a temperature jump across the interface, and a linear profile of the density in Ω_1. The first figure is the reference configuration corresponding to a full 3D geometry. The second one corresponds to the network approach, where two distinct 3D subdomains are connected through the interface.

As explained in the previous section, a frame that is too small may fail to properly resolve the modes of interest, even at low frequencies. For this reason, a sufficiently large frame size is chosen for this case, with $N_1 = N_2 = 200$. The resulting frequencies are listed in Tab. 4.5.

Table 4.5: Frequencies of the case depicted in Fig. 4.3 using the fields shown in Fig. 4.12. The relative error is computed using Eq. 4.29.

Mode index	Reference solution [Hz]	Storm output [Hz]	Relative errors [%]
1	741	748	0.94
2	2338	2369	1.33
3	3855	3900	1.17
4	4579	4647	1.49
5	4580	4648	1.48
6	5106	5241	2.64
7	5107	5243	2.66
8	5279	5321	0.80
9	5651	5789	2.44
10	5652	5790	2.44
11	6424	6551	1.98
12	6427	6552	1.94
Mean relative error [%]			1.78

For this simulation, although the number of 3D modes was doubled compared to previous cases (200 instead of 100), we kept $N_K = 100$ spectral modes to discretize the 2D interface. As a result, the selection algorithm retained all the spectral modes to represent the 200 3D modes present in each 3D subdomain. This means that, according to the algorithm's criteria, some 2D modes are missing. However, as seen in the results, the representation remains accurate, at least for the first 12 modes of the geometry.

4.4 Multiperforated liners

The series of test cases presented in the previous sections have demonstrated the validity of the numerical method based on spectral connections used to enforce continuity conditions across interfaces between geometrical subdomains. These tests have covered a variety of mean field configurations and confirmed the robustness of the approach in handling both uniform and non-uniform conditions.

Building on these results, we now turn to a practical application of this framework: the modeling of multiperforated plates. The role of multiperforated plates is to separate the hot combustion gases inside the chamber from the cooler air circulating around it. Thanks to the small holes distributed across the plate, a thin film of fresh air can pass through and flow along the chamber walls. This film acts as a thermal barrier, cooling the walls and protecting them from high-temperature gases. In addition to their thermal behavior, these plates also influence the acoustic of the chamber.

To model the acoustic influence of multiperforated plates, we use the Howe model [154, 161] in the STORM framework. This model introduces a pressure jump condition across the interface, which depends on Rayleigh conductivity (noted K_R, in $[m]$) that depends on several physical and geometrical parameters such as the aperture radius, plate thickness, frequency. The pressure jump is expressed as:

$$p^{\Omega_1} - p^{\Omega_2} = -\frac{\rho_u d^2}{K_R} \frac{\partial u_n^{\Omega_2}}{\partial t} \tag{4.32}$$

where $p^{\Omega_1} - p^{\Omega_2}$ is the pressure jump through the multiperforated plate and $u_n^{\Omega_2}$ is the upstream fluctuating velocity normal to the plate. The hole spacing is noted d, and the upstream density ρ_u.

This model was successfully implemented and validated by Laurent during his Ph.D. [94]. In the following, we describe how the 3D3D connection formalism is extended to incorporate this pressure jump condition. For clarity, here we recall only the form of the state equations:

$$\left\{\begin{array}{rcl} \ddot{\nu}_k + \omega_k^2 \nu_k & = & -\dfrac{\overline{c_0}^2}{\lambda_k L_t} \left(s_i \langle u_n^i | \mathcal{K}_k \rangle + s_j \langle u_n^j | \mathcal{K}_k \rangle \right) \\ \\ \dot{\mu}_k & = & \dfrac{1}{\lambda_k L_t \overline{\rho_0}} \left(s_i \langle p^i | \mathcal{K}_k \rangle + s_j \langle p^j | \mathcal{K}_k \rangle - s_i \dfrac{\rho_u d^2}{K_R} \langle \dot{u}_n^i | \mathcal{K}_k \rangle \right) \end{array}\right. \quad (4.33)$$

Note that these equations are similar to those of system 4.19, with the addition of a new term that accounts for the pressure jump induced by the multiperforated plate model. The complete state-space realization of this formalism is provided in Appendix A.4.

4.5 New strategy for handling discontinuities in thermoacoustic systems

In all the previous developments, jump conditions (based on 3D3D connections) have been implemented by explicitly splitting the geometry at the location of the discontinuity, so that each side of the interface is treated as a separate subdomain. Although this approach is effective within the state-space formalism, it raises several conceptual and practical concerns.

First, one may question the relevance of building global thermoacoustic modes as linear combinations of local solutions defined on individual subdomains rather than on the entire geometry. This is more of a philosophical remark, as the method remains mathematically justified. However, from a numerical perspective, this decomposition required the introduction of frames to ensure compatibility at the interfaces. As discussed in Chapter 3, frames can lead to conditioning issues that have been specifically addressed.

More critically, 3D3D connections do not always behave as expected, especially in complex configurations. For instance, in the post-combustion geometry discussed in Chapter 5, non-physical modes appear due to a 3D3D connection. These limitations motivated us to reconsider the entire approach based on geometrical splitting.

Instead, we propose a new modeling strategy for jump conditions, in which the geometry is no longer split at the location of jump conditions. The full domain is treated as a single, unified geometry, and jump conditions (e.g., multiperforated liners) are modeled as volumetric source terms in the Helmholtz equation, just like flame models. The state-space formulation is still used, but the only subdomains are now flame-like or source-like elements that impose the desired jump in acoustic pressure and/or velocity. This new vision eliminates the need for frames at the interface, since no explicit interface is defined, and requires only a single modal basis to describe the solution across the entire domain.

Consider a given volume where a jump condition on pressure or velocity fluctuations is imposed through an interface S_d (See Fig. 4.13).

The surface S_d where the discontinuity is imposed is parametrized by $\boldsymbol{x} = \boldsymbol{x_0}$. The unitary normal vector to the surface is noted $\boldsymbol{n}$. The goal is to properly impose a pressure jump $[p]$ and/or a velocity jump $[u]$ at the position $\boldsymbol{x_0}$.

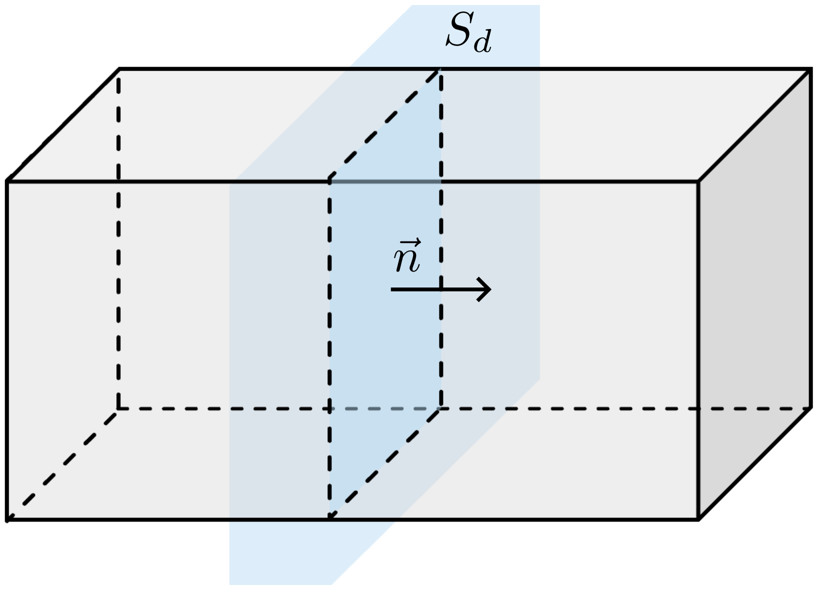


Figure 4.13: Schematic a flow domain where a pressure or velocity jump is imposed at the interface S_d.

4.5.1 Modified Helmholtz equation

The classical Helmholtz equation Eq. 2.3 must be modified to consider a pressure or velocity jump at the position $\boldsymbol{x_0}$. We start from the mass and momentum equations, where the source term associated with the flame is omitted for simplicity:

- Mass equation:

$$\frac{\partial \rho'}{\partial t} + \rho_0 \boldsymbol{\nabla} \cdot \boldsymbol{u'} = m'_u \delta(\boldsymbol{x} - \boldsymbol{x_0}) \tag{4.34}$$

- Momentum equation:

$$\rho_0 \frac{\partial \boldsymbol{u'}}{\partial t} + \boldsymbol{\nabla p'} = \boldsymbol{f'_c} \delta(\boldsymbol{x} - \boldsymbol{x_0}) \tag{4.35}$$

The flow rate induced by the velocity jump and the force induced by the pressure jump are noted m'_u of the unit $[kg.s^{-1}]$ and $\boldsymbol{f'_c}$ of the unit

$[N]$, respectively. To be consistent with units of force $\boldsymbol{f_c'}$ and flow rate m_u', the Dirac function $\delta(\boldsymbol{x}-\boldsymbol{x_0})$ introduced here is 3-dimensional, with unit of $[m^{-3}]$, and normalized to 1 over the whole volume. Therefore, we also introduce $\delta_n(\boldsymbol{x}-\boldsymbol{x_0}) = S_d\delta(\boldsymbol{x}-\boldsymbol{x_0})$ the 1-dimensional Dirac function which is normalized to 1 over the normal direction to the surface S_d.

Assuming a given mean flow (ρ_0, p_0 and γ) and taking the difference between the time derivative of Eq. 4.34 and the divergence of Eq. 4.35, a Helmholtz equation can be obtained for the pressure fluctuation (isentropicity is used to relate the density fluctuation to the pressure oscillation, i.e. $p' = \rho' c_0^2$):

$$\begin{aligned}\boldsymbol{\nabla}\cdot\left(\frac{1}{\rho_0}\boldsymbol{\nabla p'}\right) - \frac{1}{\gamma p_0}\frac{\partial^2 p'}{\partial t^2} &= -\frac{1}{\rho_0}\frac{\partial m_u'}{\partial t}\delta(\boldsymbol{x}-\boldsymbol{x_0}) \\ &+ \boldsymbol{\nabla}\cdot\left(\frac{1}{\rho_0}\boldsymbol{f_c'}\delta(\boldsymbol{x}-\boldsymbol{x_0})\right)\end{aligned} \tag{4.36}$$

The explicit expressions of m_u' and $\boldsymbol{f_c'}$ must now be derived. Consider a small volume element $\Delta_V = S_d\Delta_x$ of surface S_d and of width Δ_x surrounding the position where the jump occurs. Integrating Eq. 4.34 over this volume:

$$\frac{\partial}{\partial t}\iiint_{\Delta_V}\frac{1}{\rho_0}\rho' \mathrm{d}V + \iiint_{\Delta_V}\boldsymbol{\nabla}\cdot\boldsymbol{u}\ \mathrm{d}V = \frac{m_u'}{\rho_0} \tag{4.37}$$

The first term tends to 0 when the volume Δ_V tends to 0. By using the Green-Ostrogradsky theorem, it comes:

$$m_u' = \rho_0[u]S_d \tag{4.38}$$

Integrating 4.35 over the same volume Δ_V

$$\iiint_{\Delta_V} \rho_0 \frac{\partial \boldsymbol{u'}}{\partial t} \mathrm{d}V + \iiint_{\Delta_V} \boldsymbol{\nabla p'} \mathrm{d}V = \boldsymbol{f'_c} \tag{4.39}$$

The first term tends to 0 when the volume Δ_V tends to 0. This equation is projected onto the normal of the surface $\boldsymbol{n}$ It comes:

$$\iiint_{\Delta_V} \nabla_n p' \mathrm{d}V = \boldsymbol{f'_c} \cdot \boldsymbol{n} \tag{4.40}$$

where ∇_n is the gradient operator projector onto the normal direction to the surface. So the only component of the force $\boldsymbol{f'_c}$ is normal to the surface and:

$$S_d[p] = \boldsymbol{f'_c} \cdot \boldsymbol{n} \tag{4.41}$$

Replacing $\delta_n(\boldsymbol{x} - \boldsymbol{x_0}) = S_d \delta(\boldsymbol{x} - \boldsymbol{x_0})$, the explicit form of the Helmholtz equation including flames and jump condition on p' and u' is written:

$$\begin{aligned} \boldsymbol{\nabla} \cdot \left(\frac{1}{\rho_0} \boldsymbol{\nabla} p' \right) - \frac{1}{\gamma p_0} \frac{\partial^2 p'}{\partial t} \quad = \quad & \underbrace{-\frac{\gamma - 1}{\gamma p_0} \frac{\partial \omega'_T}{\partial t}}_{flame} \\ & \underbrace{-\frac{\partial [u]}{\partial t} \delta_n(\boldsymbol{x} - \boldsymbol{x_0})}_{jump\ of\ u'} \\ & + \underbrace{[p] \nabla_n \cdot \left(\frac{1}{\rho_0} \delta_n(\boldsymbol{x} - \boldsymbol{x_0}) \right)}_{jump\ of\ p'} \end{aligned} \tag{4.42}$$

Applying the Fourier transform to Eq. 4.42 leads to the following frequency-domain Helmholtz equation:

$$
\begin{aligned}
\boldsymbol{\nabla}\cdot\left(\frac{1}{\rho_0}\boldsymbol{\nabla}\hat{p}\right)+\frac{\omega^2}{\gamma p_0}\hat{p} &= \underbrace{-j\omega\frac{\gamma-1}{\gamma p_0}\hat{\omega}_T}_{flame} \\
& \underbrace{-j\omega[u]\delta_n(\boldsymbol{x}-\boldsymbol{x_0})}_{jump\ u'} \\
& \underbrace{+\frac{[p]}{\rho_0}\delta'_n(\boldsymbol{x}-\boldsymbol{x_0})}_{jump\ p}
\end{aligned}
\tag{4.43}
$$

where $\delta'_n = \nabla_n\delta_n$ is the derivative of the Dirac function in the normal direction to the surface. In the sense of distribution, the Dirac $\delta_n(\boldsymbol{x})$ is defined by the following property:

$$
\int_{-\infty}^{+\infty}\delta_n(\boldsymbol{x}-\boldsymbol{x_0})f(\boldsymbol{x})\mathrm{d}x = f(\boldsymbol{x_0}) \tag{4.44}
$$

for any test function $f(\boldsymbol{x})$ that is continuous at $\boldsymbol{x}=\boldsymbol{x_0}$. Note that the integration is performed in the normal direction to the surface. Applying integration by parts to Eq. 4.44 leads to the expression for the first derivative of the Dirac function, denoted $\delta'_n(\boldsymbol{x})$. Its projection on the normal to the surface is defined in the sense of distributions by:

$$
\int_{-\infty}^{+\infty}\delta'_n(\boldsymbol{x}-\boldsymbol{x_0})f(\boldsymbol{x})\mathrm{d}x = -\nabla_n f(\boldsymbol{x_0}) \tag{4.45}
$$

4.5.2 State-space of geometrical domain

Introducing the pressure decomposition Eq. 3.16 in the modified Helmholtz equation 4.43 and projecting it on the basis mode U_n leads to the follwing temporal evolution of modal amplitudes $\Upsilon_n(t)$:

$$\left\{\begin{array}{rcl} p'(\boldsymbol{x},t) & = & \sum_n^N \dot{\Upsilon}_n(t)U_n(x) \\ \\ \ddot{\Upsilon}_n(t) & = & -\sum_k^N [\boldsymbol{\Sigma V}^T\boldsymbol{\Omega^2 V\Sigma}^{-1}]_{n,k}\Upsilon_k(t) \\ \\ & - & \langle H'|U_n\rangle c_0^2 \displaystyle\int^t [p](\boldsymbol{x_0},t')\mathrm{d}t' \\ \\ & + & \langle H|U_n\rangle \rho_0 c_0^2 [u](\boldsymbol{x_0},t) \end{array}\right. \tag{4.46}$$

In the same way as for the definition of flame shapes, a sharp Gaussian shape $H(\boldsymbol{x})$, of unit $[m^{-1}]$ here, and of finite thickness σ is introduced to approximate the Dirac function $\delta_n(\boldsymbol{x}-\boldsymbol{x_0})$, to avoid the Gibbs oscillation at the pressure or velocity jump position. This equation is valid in any 3-dimensional case where no flame is present in the system. Depending on the modeling of the pressure or velocity jump, the state-space representation may vary. Different cases are provided in Appendices A.8, A.9, and A.10.

When a pressure jump is present in a geometrical domain, it is also necessary to modify the modal expansion of velocity fluctuations $\boldsymbol{u'}$ according to Eq. 4.35:

$$\frac{\partial}{\partial t}\boldsymbol{u'}(\boldsymbol{x},t) = -\frac{1}{\rho_0}\boldsymbol{\nabla p'} + \frac{1}{\rho_0}[p]\delta_n(\boldsymbol{x}-\boldsymbol{x_0})\boldsymbol{n} \tag{4.47}$$

The modal expansion of the pressure field is never modified and remains $p'(\boldsymbol{x},t) = \sum_{n=1}^N \dot{\Upsilon}_n U_n(\boldsymbol{x})$. Equation 4.47 must be considered when defining the output vector of the geometrical subdomain (see Appendices A.9 and A.10) and when constructing the velocity mode shape at the end of the method.

4.5.3 Validation

To validate the newly introduced approach for modeling jump conditions, the configuration shown in Fig. 4.14 is considered. It is a 1-dimensional duct where a pressure or a velocity jump is applied at the position $x = x_0$.

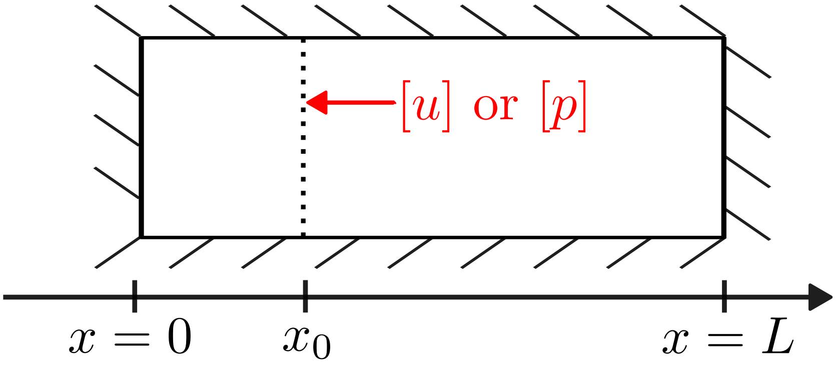


Figure 4.14: Scheme of the academic configuration of a single 1-dimensional duct of length $L = 1\ m$, where either a pressure jump $[p]$ or velocity jump $[u]$ is applied at the position $x_0 = 0.32\ m$.

In this section, three distinct test cases are considered. Each case targets a specific type of discontinuity and serves to evaluate the method:

- A velocity jump condition of the form $[u] = \alpha u(x_0)$. Depending on the value of α, this condition typically models either a cross sectional change or the effect of a flame.
- A pressure jump condition of the form $[p] = \alpha p(x_0)$. While this condition may not correspond directly to a physical interface, it is used here as a mathematical test to assess the general applicability of the formulation.
- A pressure jump of the form $[p] = -\rho_0/K_R \cdot \partial u_n(x_0, t)/\partial t$, which is the Howe model for multiperforated liners [154, 161]. In this notation, K_R is the Rayleigh conductivity with unit of $[m^{-1}]$.

In all test cases, the density and speed of sound are constant and set to $\rho_0 = 1.177\ kg \cdot m^{-3}$ and $c_0 = 347.12\ m \cdot s^{-1}$, respectively. Neumann boundary conditions (i.e., $u'_n = 0$) are applied at both ends of the domain.

Since the geometry is not split into multiple subdomains, no frame is required. A single orthogonal modal basis satisfying the Neumann boundary conditions is sufficient to represent the solution across the entire domain. This simplification highlights one of the key advantages of the new method.

For each test case, a pseudo-analytical reference solution is provided in the corresponding appendix (see App. B.1.1, B.1.2, and B.1.3). The associated state-space realizations for both the geometrical domain and the source domain are also detailed in Appendices A.8, A.9 and A.10.

4.5.3.1 Velocity jump condition: $[u] = \alpha u(x_0)$

This first test case focuses on a velocity jump condition of the form $[u] = \alpha u(x_0)$. As for flame definition, the Dirac function acting as support for the source term must be approximated by a Gaussian function to avoid any Gibbs perturbation (see Sec. 3.7.2 for more details). The Gaussian thickness is set to $\sigma = 0.015\ m$. The size of the modal basis is set to $N = 100$, leading to a projection error of $\varepsilon_{proj} \simeq 1.7 \times 10^{-3}$. This projection error is calculated in the same way as for flames, that is, using Eq. 3.27 in Section 3.7.1. The velocity jump is set to mimic a jump section that is: $\alpha = S_1/S_2 - 1$, where $S_1 = 10^{-4}\ m^2$ and $S_2 = 2.10^{-4}\ m^2$.

The comparison in terms of frequencies between the pseudo-analytical solution and the modal expansion approach is given in Tab. 4.6

Figure 4.15 shows the shape of the pressure and velocity of the first three modes.

The results of the modal expansion method, including the velocity jump

Table 4.6: Frequencies of the first five modes of the case depicted in Fig. 4.14, using the pseudo-analytical method or the modal expansion approach.

Mode index	Pseudo-analytical [Hz]	Network-based [Hz]
1	191	189
2	331	331
3	516	512
4	712	708
5	855	856

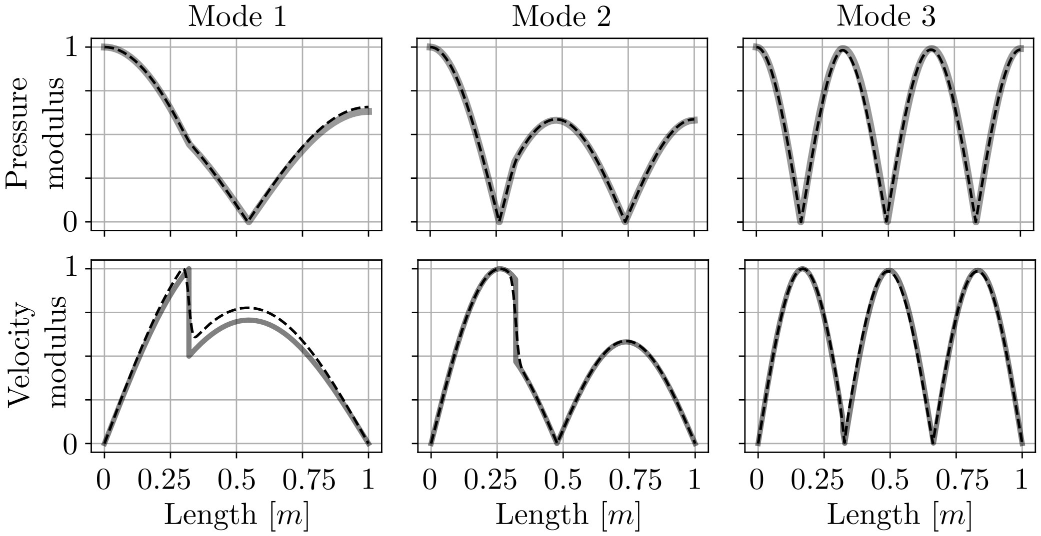


Figure 4.15: Shape of the pressure and velocity fluctuations (modulus) of the first three modes in the case depicted in Fig. 4.14. Dashed lines: results of the modal expansion method. Full line: pseudo-analytical results, computed with Eq. B.5.

term as a source term in the Helmholtz equation, are correct, which validates the capability of this method. However, a slight deviation is observed in the mode shapes, particularly in the velocity of mode 1. Indeed, it is not possible to represent a discontinuity using a modal expansion formalism unless a very large basis size N, which is not the goal here. Figure 4.16 shows that this discontinuity is smoothed by the modal expansion. To obtain a steeper slope at the velocity jump position, one simply needs to reduce the width of the Gaussian function used to represent the analytical Dirac function. To keep a relatively low projection error, the size of the modal basis is increased when the Gaussian function is sharpened. To illustrate this convergence, four different cases were performed by keeping a relatively low projection error. They are summarized in Tab. 4.7.

Table 4.7: The four cases performed to illustrate the convergence by sharpening the Gaussian shape.

Frame size N	Gaussian thickness σ $[m]$	Projection error ε_{proj} $[-]$
100	0.015	1.7×10^{-3}
150	0.01	1.6×10^{-3}
250	0.005	9.5×10^{-3}
500	0.002	4.1×10^{-2}

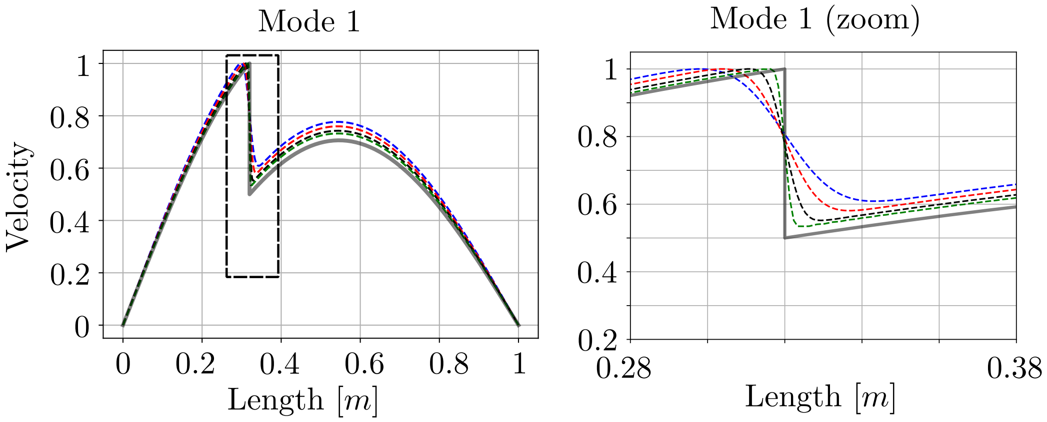


Figure 4.16: Left: Shape of the velocity fluctuations (modulus) of mode 1. Right: Zoom around the velocity jump position. Full line: analytical solution. The blue, red, black and green dashed lines correspond to the cases parametrized by lines 1, 2, 3 and 4 of Tab. 4.7, respectively.

4.5.3.2 Pressure jump condition: $[p] = \alpha p(x_0)$

This second test case introduces a pressure jump condition of the form $[p] = \alpha p(x_0)$ applied at a given interface location x_0. Unlike the previous velocity-jump case, this configuration requires a modification of the velocity modal expansion.

As explained earlier (see Eq. 4.47), the presence of a pressure discontinuity modifies the modal expansion of the velocity field. For clarity, the corrected velocity expansion used in this test case is recalled below. The full derivation of this formulation is provided in Appendix A.9.

$$\begin{aligned} \boldsymbol{u'}(\boldsymbol{x},t) &= -\frac{1}{\rho_0}\textstyle\sum_{n=1}^{N} \Upsilon_n(t)\boldsymbol{\nabla U_n(x)} \\ &+ \frac{1}{\rho_0}\alpha\textstyle\sum_{n=1}^{N} \Upsilon_n(t)U_n(\boldsymbol{x_0})\widetilde{H}(\boldsymbol{x})\boldsymbol{n} \end{aligned} \tag{4.48}$$

where $\boldsymbol{n}$ is the unitary vector normal to the surface where the jump occurs. As before, the analytical reference solution and the corresponding state-space models used in this configuration are provided in Appendices B.1.2 and A.9, respectively.

For this test case the frame size is set to $N = 100$ and the Gaussian thickness to $\sigma = 0.015\ m$, as in the previous section. However, as explicitly stated in Eq. 4.43, the support for the pressure jump is the derivative of the Dirac function, which is therefore approximated by the derivative of the Gaussian. For the same value of the frame size $N = 100$, the projection error is increased up to $\varepsilon_{proj} = 7.6 \times 10^{-3}$ when considering a pressure jump, instead of $\varepsilon_{proj} = 1.7 \times 10^{-3}$ for the velocity jump.

The shapes of the first three modes are presented in Fig. 4.17 and the frequencies are given in Tab. 4.8.

Despite the non-physical nature of the imposed pressure jump condition,

Table 4.8: Frequencies of the first five modes of the case depicted in Fig. 4.14, when considering a pressure jump of the form $[p] = \alpha p(x_0)$, and using the pseudo-analytical method or the network-based approach.

Mode index	Pseudo-analytical [Hz]	Network-based [Hz]
1	183	185
2	338	339
3	518	518
4	705	707
5	860	865

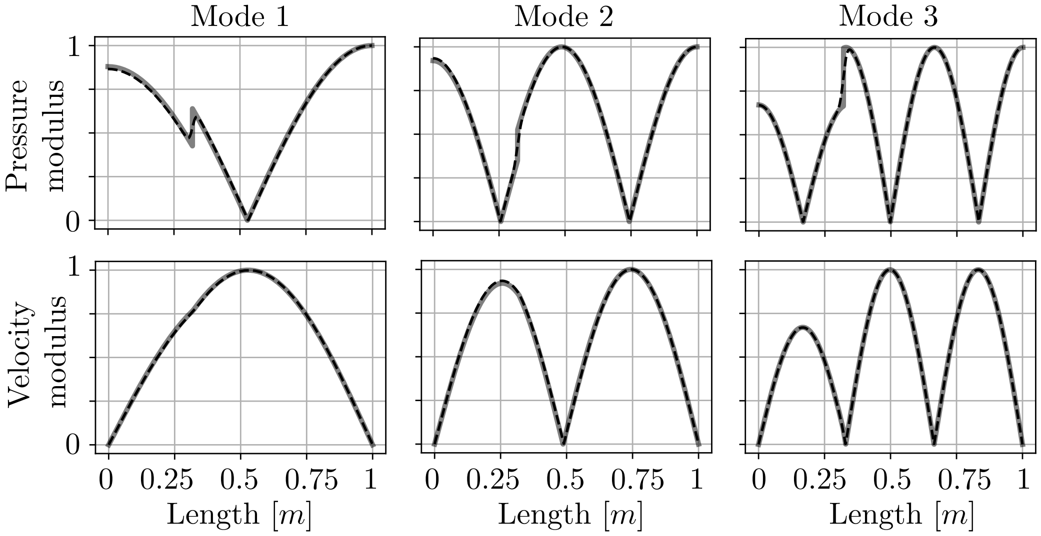


Figure 4.17: Shape of the pressure and velocity fluctuations (modulus) of the first three modes in the case depicted in Fig. 4.14, when considering a pressure jump of the form $[p] = \alpha p(x_0)$. Dashed lines: results of the modal expansion method. Full line: pseudo-analytical results, computed with Eq. B.7.

this test case provides a valuable verification of the proposed modeling strategy. The results obtained for both the frequencies and the mode shapes show excellent agreement with the reference solution. This confirms that the formalism accurately captures the correct global modes.

4.5.4 Multiperforated liner jump condition

The last test case investigates a pressure jump condition representative of a multiperforated liner, a common acoustic treatment in aeronautical combustion chambers. The discontinuity is modeled using Howe's formulation [154, 161], in which the pressure jump is proportional to the time derivative of the normal velocity:

$$[p] = -\frac{1}{K_R}\rho_0 \frac{\partial u_n'(x_0,t)}{\partial t} \tag{4.49}$$

This type of condition introduces both a reactive and a dissipative behavior depending on the value and frequency dependence of the Rayleigh conductivity K_R. In this simplified validation case, a constant value of K_R is considered. However, if K_R depends on pulsation ω, the function must be fitted using the Vector Fitting algorithm, similarly to what is done for flame transfer functions or impedance models.

Because of the pressure discontinuity, the modal expansion of the velocity field must again be modified. Only the final corrected expression is recalled below, while the full derivation is provided in App. A.10.

$$\left\{\begin{array}{rcl} \boldsymbol{u'}(\boldsymbol{x},t) & = & -\dfrac{1}{\rho_0}\sum_{n=1}^{N}\Upsilon_n(t)\boldsymbol{\nabla U_n(x)} \\ \\ & + & \dfrac{\widetilde{H}(\boldsymbol{x})\boldsymbol{n}}{\rho_0\left(K_R+\widetilde{H}(x_0)\right)}\sum_{n=1}^{N}\Upsilon_n(t)\nabla_n U_n(\boldsymbol{x_0}) \end{array}\right. \tag{4.50}$$

where $\boldsymbol{n}$ is the unitary vector normal to the surface where the jump occurs. As in the previous configurations, the method is validated by comparing the frequencies and mode shapes against pseudo-analytical solutions given in App. B.1.3.

For this test case we set the frame size to $N = 100$ and the Gaussian thickness to $\sigma = 0.015\ m$, as in the previous section. The Rayleigh conductivity is arbitrary set to $K_R = 0.5\ m^{-1}$. Comparison between the modal expansion method and the pseudo-analytical one in terms of frequency is presented in Tab. 4.9

Table 4.9: Frequencies of the first five modes of the case depicted in Fig. 4.14, when considering a pressure jump of the form $[p] = -\rho_0/K_R \times \partial u_n'(x_0, t)/\partial t$ that models a liner, and using the pseudo-analytical method or the network-based approach.

Mode index	Pseudo-analytical [Hz]	Network-based [Hz]
1	77	79
2	264	264
3	514	513
4	551	580
5	769	769

The shapes of the first three modes are presented in Fig. 4.18

The difference in the pressure jump for mode 1 is only due to the graph display, as there is a change of sign in the real part at the jump position, that is smoothed by the modal expansion. If we look at the real part instead of the absolute value, this difference disappears, as shown in Fig. 4.19. However, the differences observed between the pressure and velocity shapes for mode 3 are due to the non-zero thickness of the Gaussian profile. As explained in Sec. 4.5.3.1, the error can be reduced by sharpening the Gaussian function and increasing the frame size (similar trends can be seen in Fig. 4.16).

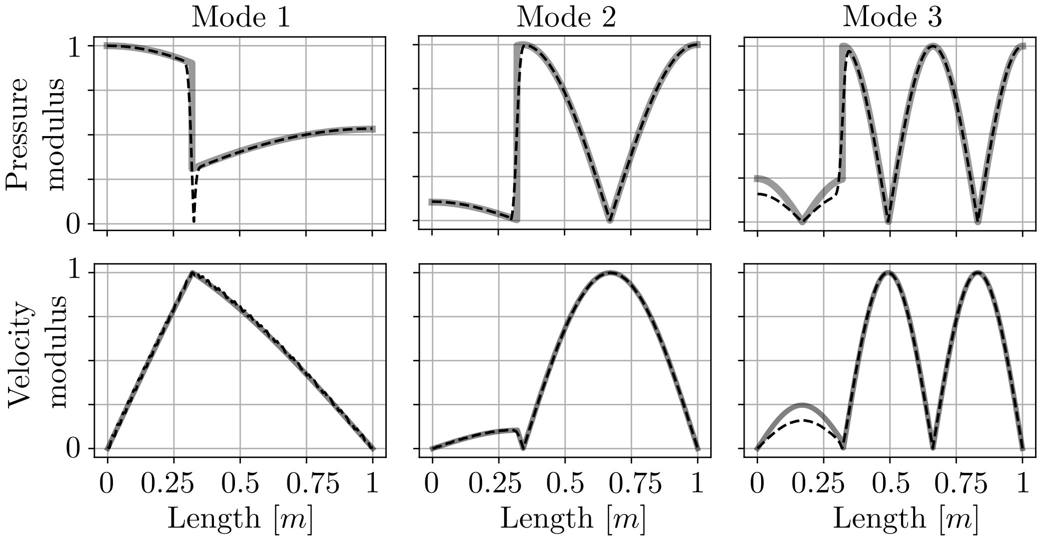


Figure 4.18: Shape of the pressure and velocity fluctuations of the first three modes (modulus) in the case depicted in Fig. 4.14, when considering a pressure jump of the form $[p] = -\rho_0/K_R \times \partial u'_n(x_0,t)/\partial t$ that models a liner. Dashed lines: results of the modal expansion method. Full line: pseudo-analytical results, computed with Eq. B.9.

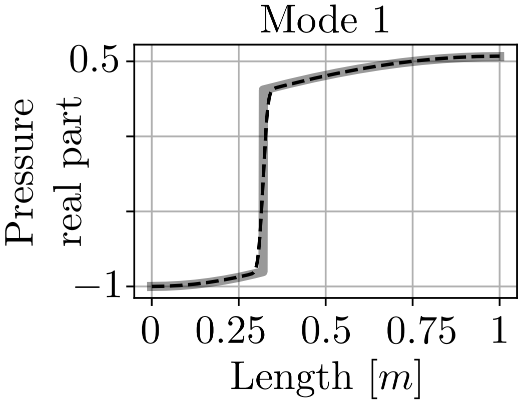


Figure 4.19: Shape of the pressure (real part) of the mode 1 in the case depicted in Fig. 4.14, when considering a pressure jump of the form $[p] = -\rho_0/K_R \times \partial u'_n(x_0,t)/\partial t$ that models a liner. Dashed lines: results of the modal expansion method. Full line: pseudo-analytical results, computed with Eq. B.9.

According to Tab. 4.9, the frequencies are correct except for mode 4. The discrepancy with the pseudo-analytical method arises from the nonzero thickness of the Gaussian and its projection onto the modal basis, or more precisely the projection of its derivative. Indeed, increasing the number of modes in the geometrical domain reduces this error and allows convergence toward the analytical solution. Two competing phenomena are at play here. The first is that the Gaussian thickness must be sufficiently small to accurately represent the pressure jump. The second is that the finer the Gaussian, the larger the number of modes required in the modal basis for an accurate projection. If the number of modes is insufficient, the projection of the Gaussian shape is poor, leading to spurious fluctuations in the output modes, as illustrated in Fig. 4.20.

Table 4.10: Summary of the three cases used to illustrate spurious oscillations when sharpening the Gaussian profile while keeping a constant frame size. The projection error is computed on the derivative of the Gaussian function.

Frame size N	Gaussian thickness σ $[m]$	Projection error ε_{proj} $[-]$
100	0.015	7.6×10^{-3}
100	0.0125	3.8×10^{-2}
100	0.010	1.4×10^{-1}

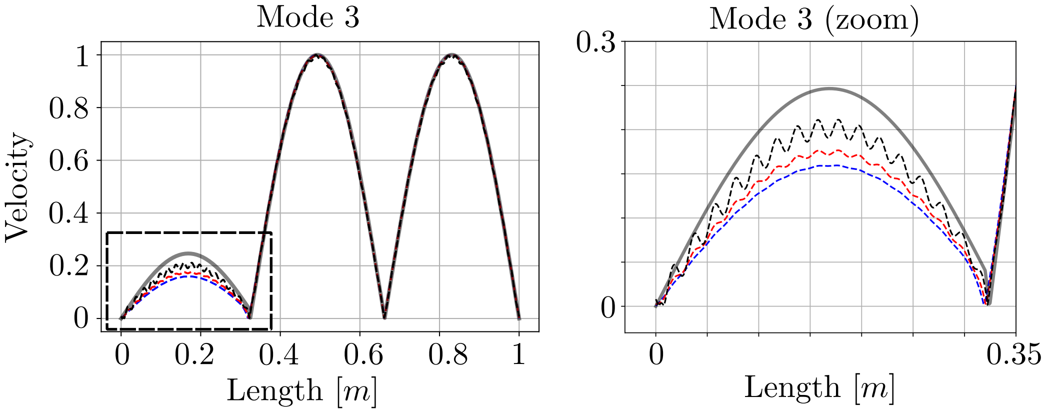


Figure 4.20: Left: Shape of the velocity fluctuations (modulus) of mode 3. Right: Zoom. Full line: analytical solution. The blue, red and black dashed lines correspond to the cases parametrized by lines 1, 2 and 3 of Tab. 4.10, respectively.

4.6 Conclusion

This chapter first recalled how impedance boundary conditions are imposed in the 1D case, to illustrate the general method for deriving a state-space representation of such conditions. We also adapted the spectral connection methodology, introduced and implemented by C. Laurent within the STORM framework, which allows boundary or jump conditions to be enforced in 3D cases. Due to the numerical challenges of these 3D-3D connections, aligning the spectral modes becomes necessary - as explained in Sec. 4.2.3 - using the matching matrix, which modifies the initial state-space.

The 3D-3D connection framework was then successfully tested and validated on a simple geometry under various mean field configurations, to demonstrate the robustness and applicability of the method. Ensuring a robust and accurate implementation of spectral connections is essential, as multiperforated plate models are typically based on this formalism. The pressure jump induced by the liner is added as an extra term in the state-space formulation of the spectral connection.

In a second step, an alternative approach is proposed to impose jump conditions in a thermoacoustic system. Instead of splitting the domain into two subdomains (as explained in Chap. 3), the system is treated as a single global domain, and the jump condition is introduced as a discontinuity in the Helmholtz equation itself. This leads to additional source terms and modifies the modal expansions and therefore the state-space equations.

This new formalism was tested and validated in a 1D configuration for various types of jump conditions:

- A velocity jump proportional to the upstream velocity, which typically represents a cross-section area change or a flame.

- A pressure jump proportional to upstream pressure — not physically meaningful, but still numerically verified.
- A pressure jump proportional to the time derivative of upstream velocity, as in the Howe's model for multiperforated liners.

In the case of internal discontinuities, particular care must be taken when projecting the Gaussian function used to localize the discontinuity onto the modal frame. If the projection is not accurate, spurious, non-physical oscillations may appear in the reconstructed mode shapes. Moreover, the Gaussian function must be sufficiently narrow to represent the discontinuity accurately. However, the sharper the Gaussian, the larger the number of frame modes required to project it correctly. A balance must therefore be found between the sharpness of the Gaussian function and the size of the modal frame.

Extending this formalism to three-dimensional configurations on complex meshes would be a natural but challenging next step, and is therefore not addressed in the present work. In three dimensions, additional constraints arise from the spatial discretization of the computational mesh. In particular, an accurate enforcement of jump conditions requires sufficient mesh resolution at the interface to ensure a precise projection of the support function used to define the discontinuity. Inadequate discretization at this interface may lead to significant projection errors and, consequently, to inaccurately imposed jump conditions.

Moreover, an additional difficulty specific to three-dimensional configurations lies in the definition of the Gaussian function used to localize the jump condition. Unlike the one-dimensional case, where the discontinuity is associated with a single spatial location, the jump condition in 3D is supported on a two-dimensional patch that may exhibit a complex and non-planar geometry. Defining a Gaussian function consistently centered on such a surface, while maintaining both sufficient spatial localization and numerical robustness, is a non-trivial task that would require dedicated developments. Addressing these challenges would demand substantial methodological and computational efforts.

Chapter 5

Industrial and Experimental Applications

Contents

5.1 The MIRADAS conifguration

The methodology to make the frame-modal expansion approach more reliable and predictive is tested on an experimental configuration studied at the Institut de Mecanique des Fluides de Toulouse (IMFT), named MIRADAS. The configuration setup is detailed in Fig. 1 and Fig. 2 in [162] and is shown in Fig. 5.1.

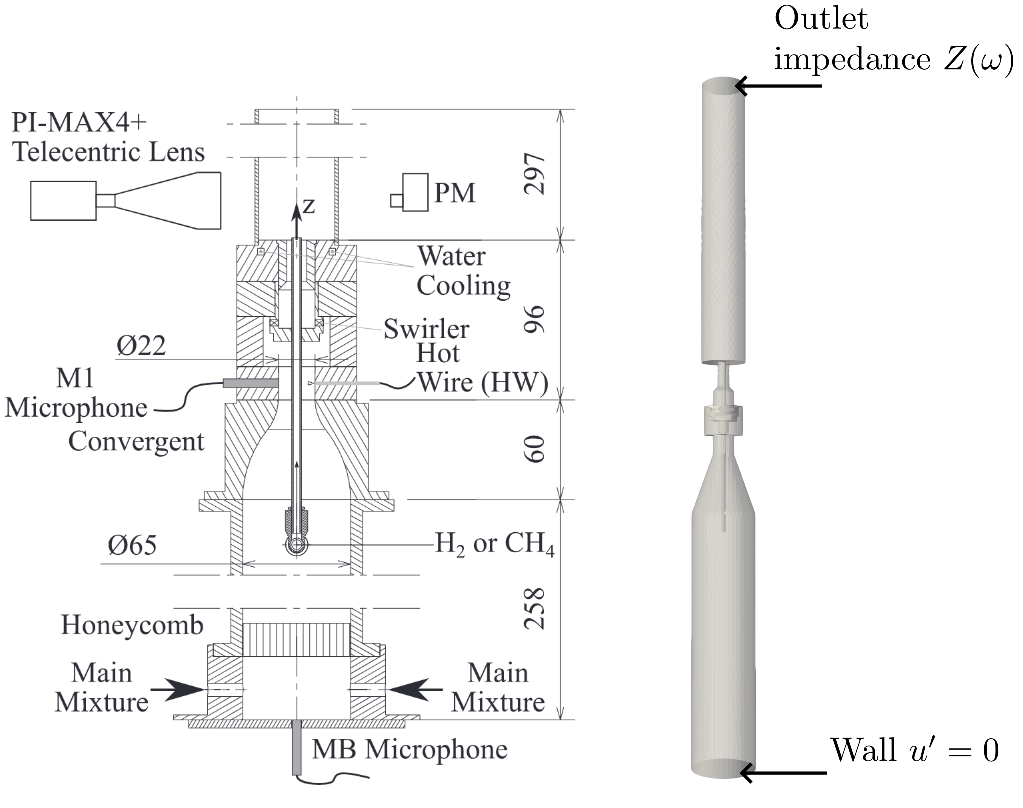


Figure 5.1: Left: Experimental setup of the MIRADAS configuration. Taken from [162]. Right: Computational domain and boundary conditions used in the simulation.

Oztarlik *et al.* have notably measured the Flame Transfer Function for different operating points (see Fig. 14 in [162]). In this section, the operating point named "Ref" is studied, which presents an unstable mode around $f = 591$ Hz (see Tab. 2 in [162]). The FTF and its fit with the Vector

Fitting algorithm are depicted in the left part of Fig. 5.2. The reflection coefficient $R(\omega)$ at the output of the combustion chamber was also measured by Oztarlik, given in Fig. 4.7 in [163]. As this reflection coefficient depends on ω, the outlet impedance $Z(\omega) = (1+R)/(1-R)$ must also be fitted with the Vector Fitting algorithm (see right part of Fig. 5.2)

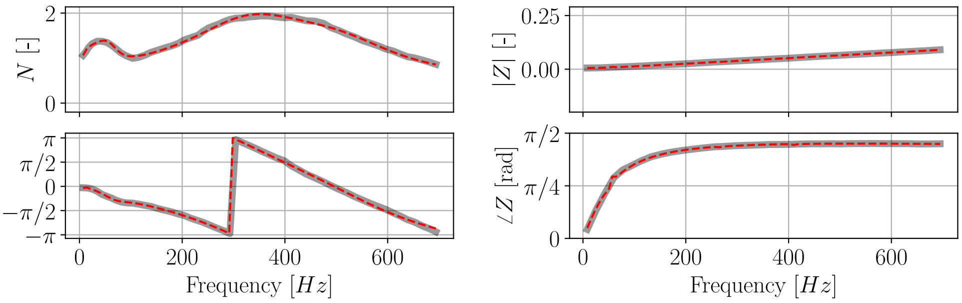


Figure 5.2: Left: Modulus and phase of the flame Transfer Function associated with the "Ref" operating point in Oztarlik *et al* [163]. Right: Modulus and phase of the impedance boundary condition at the combustion chamber outlet. Gray lines: experimental measurements from [162, 163]. Red dashed lines: the fits reconstructed using Eq. 3.31 and the Vector Fitting algorithm [105]. The number of Pole Base Function is set to $N_{PBF}^{fl} = 6$ for the FTF fit and to $N_{PBF}^{imp} = 9$ for the impedance fit.

The mean physical fields $\gamma(\boldsymbol{x})$, $\rho_0(\boldsymbol{x})$ and the value of p_0 used to compute the frame modal basis (see Eq. 3.6) are determined by the LES solver AVBP [164] developed at CERFACS and used for the presented thermoacoustic study (see Fig. 5.3). The value of the mean pressure is $p_0 = 101400\ Pa$.

As an impedance boundary condition has to be applied at the outlet of the combustion chamber (see Fig. 5.1), the two orthogonal basis constituting the frame are computed by applying either a Dirichlet or a Neumann boundary condition at the outlet. The modes are computed with the FEniCS software [129, 165], using Krylov-Schur method. The number of modes in each basis is set to 30, leading to a total of 60 modes in the initial frame. The solver

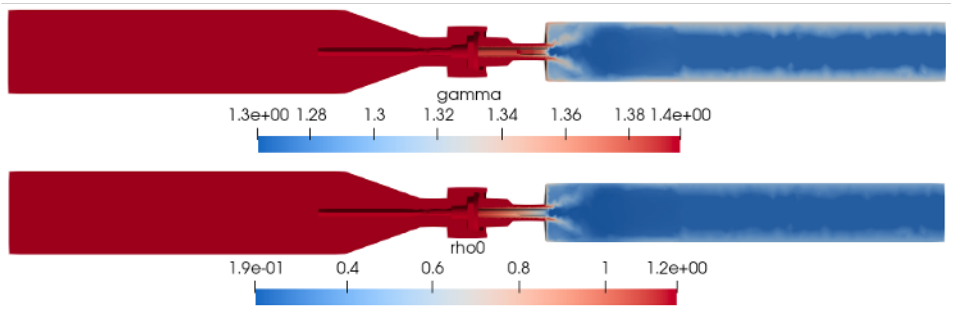


Figure 5.3: Mean fields γ and ρ_0 used to compute the frame modal basis, determined using Large Eddy Simulation.

tolerance is set to 10^{-7}. As detailed in Appendix C.2, the frame truncation threshold should theoretically be set close to the solver tolerance. To ensure the quality of the truncated frame, the truncation threshold is set just above the solver tolerance: $\varepsilon_t = 10^{-6}$ (see the singular value distribution in Fig. 5.4). All the modes associated with a singular value lower than the threshold, that is to say a total of 23 modes, were removed. This process effectively eliminates 23 non-physical modes from the overall output of the method.

The presented method is compared to the well-established thermoacoustic code AVSP developed at CERFACS [166]. As two different fit are used in this configuration (impedance and flame), the criterion to identify the non-physical modes uses a cross-comparison based on the Eq. 3.33. A total of nine simulations are performed, using three different values for the impedance fit order, $N_{PBF}^{imp} = 9,\ 10,$ and 11, and three different values for the flame fit order, $N_{PBF}^{fl} = 6,\ 7,$ and 8. The values $N_{PBF}^{imp} = 9$ and $N_{PBF}^{fl} = 6$ are chosen as the base orders since they provide a good fit of the impedance and the FTF. The orders are then incremented by one to identify non-physical modes. The case $N_{PBF}^{fl} = 6$ and $N_{PBF}^{imp} = 9$ is defined as the main case and eight sets of errors $\epsilon_s(n)$ (see Eq. 3.33) are computed using the eight other simulations. As the cost of one fit followed by one simulation is about 2 seconds, we are

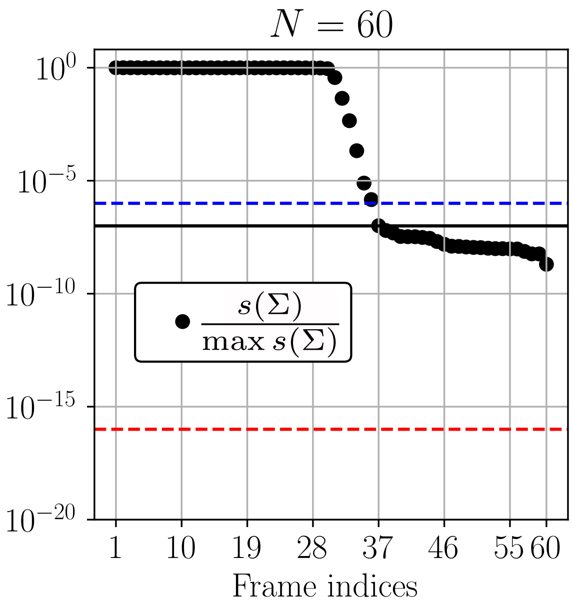


Figure 5.4: Singular value distribution of the Σ matrix normalized by their maximum, associated with the MIRADAS configuration. Σ is given by the SVD of the frame, Eq. 3.15. The red dashed line corresponds to 10^{-16}, an indication of the machine precision. The blue dashed line corresponds to the truncation threshold set to $\varepsilon_t = 10^{-6}$, just above the solver tolerance at 10^{-7} (horizontal black line).

not limited by the number of simulations in terms of computational cost. These sets of errors are added to define a global error: $\epsilon_g(n) = \sum_{i=1}^{8} \epsilon_s^i(n)$, where the index i refers to the eight different cases summarized in Tab. 5.1. The logarithm of $\epsilon_g(n)$ is plotted in Fig. 5.5.

Simulation index (i)	N_{PBF}^{imp}	N_{PBF}^{fl}
Main	9	6
1	9	7
2	9	8
3	10	6
4	10	7
5	10	8
6	11	6
7	11	7
8	11	8

Table 5.1: Value of the fit order in the nine different simulations performed used to identify non-physical components in the output.

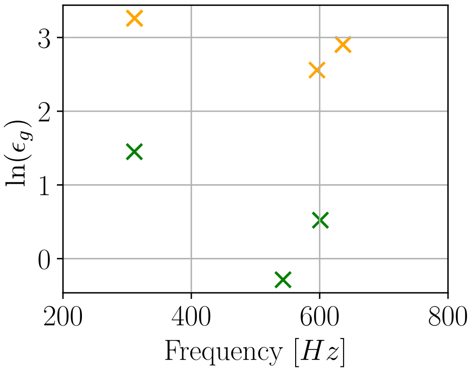


Figure 5.5: Log of the global error ϵ_g, computed as the sum of eight sets of errors ϵ_s. The coloring is based on the K-means algorithm with two clusters.

The frequency and growth rate of the three modes with the lowest global errors $\epsilon_g(n)$ (green crosses in Fig. 5.5) are plotted in the frequency plane in the right part of Fig. 5.6.

The presented method found an unstable mode (mode 3) at $f_3 = 606$ Hz, $\sigma_3 = 25\ s^{-1}$ which is consistent with the mode measured in the experiment at 591 Hz. It corresponds to the coupling between the flame and the second pure acoustic mode (at 602 Hz). The second mode at $f_2 = 546$ Hz is computed as a marginally unstable mode (low growth rate: $\sigma_2 = 1.2\ s^{-1}$), and corresponds to a small coupling between the flame and the first pure acoustic mode of the

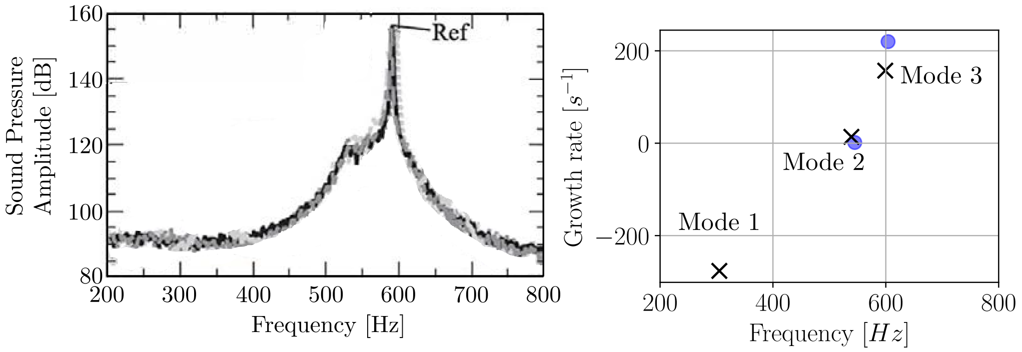


Figure 5.6: Left: Experimental measurements of the pressure amplitude (adapted from Oztarlik [162]). Right: Numerical results using the truncated SVD modal expansion method (black crosses) and the FEM solver AVSP (blue dots). Modes are indexed in increasing order of frequency.

geometry, which is at 558 Hz. The first mode at $f_1 = 297$ Hz is only found by the presented method. The Finite Element solver AVSP requires an initial guess f_0 to compute one by one the thermoacoustic modes. By setting the initial guess at f_1, AVSP converges to the mode 2. As the phase difference of the mode 1 between the velocity upstream and downstream of the flame is $-\pi$ (see Fig. 5.7), the mode can be classified as an Intrinsic Thermoacoustic (ITA) mode [167].

Furthermore, as the basin of attraction of ITA modes is very small when using iterative algorithms to solve the non-linear eigenvalue problem [85], AVSP failed to find this mode. Fortunately, the modal expansion method does not require an initial guess, and all the thermoacoustic modes are computed in one step, as the eigenproblem which defines the global system is linear.

To give an order of magnitude for the computation times in this configuration (approximately $30,000$ degrees of freedom), the nonlinear eigensolver AVSP requires about 30 minutes to 1 hour to compute the two thermoacous-

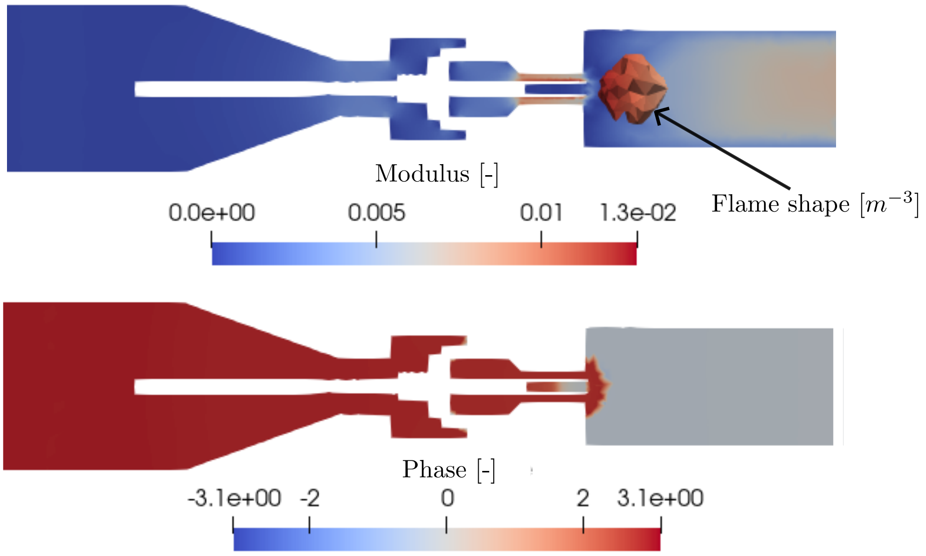


Figure 5.7: Top: Modulus of velocity fluctuations associated with the mode 1, normalized by its L2-norm. Flame shape $H(\boldsymbol{x})$ used in the simulation that satisfies $\int_V H(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = 1$. Bottom: Phase of the mode 1.

tic modes (excluding the ITA mode). In contrast, the presented frame-based modal expansion method requires an initial frame construction step. Specifically, 60 pure acoustic modes are computed using the Finite Element Method ($\sim$ 5 minutes), after which the global thermoacoustic system is assembled using the state-space formalism. This results in a linear eigenvalue problem of size $\sim$ 100, which is solved using a direct solver in about 1 second. To identify the three physical modes among the results, eight additional simulations (each taking 1 second) are performed, so the identification process takes approximately 10 seconds in total. Furthermore, once the frame is built, the method allows instantaneous recomputation of the system in response to small changes in the Flame Transfer Function (FTF) or boundary conditions, at no additional computational cost.

5.2 The HYLON configuration

The MIRADAS setup is primarily designed to study methane-air flames with a small percentage of hydrogen, and it is particularly interesting to observe that the addition of hydrogen can stabilize the flame. However, this geometry is not suitable for studying hydrogen-air flames. A new configuration, called HYLON (HYdrogen LOw Nox), has therefore been developed to investigate non-premixed hydrogen-air flames [168–170]. The configuration is depicted in Fig. 5.8.

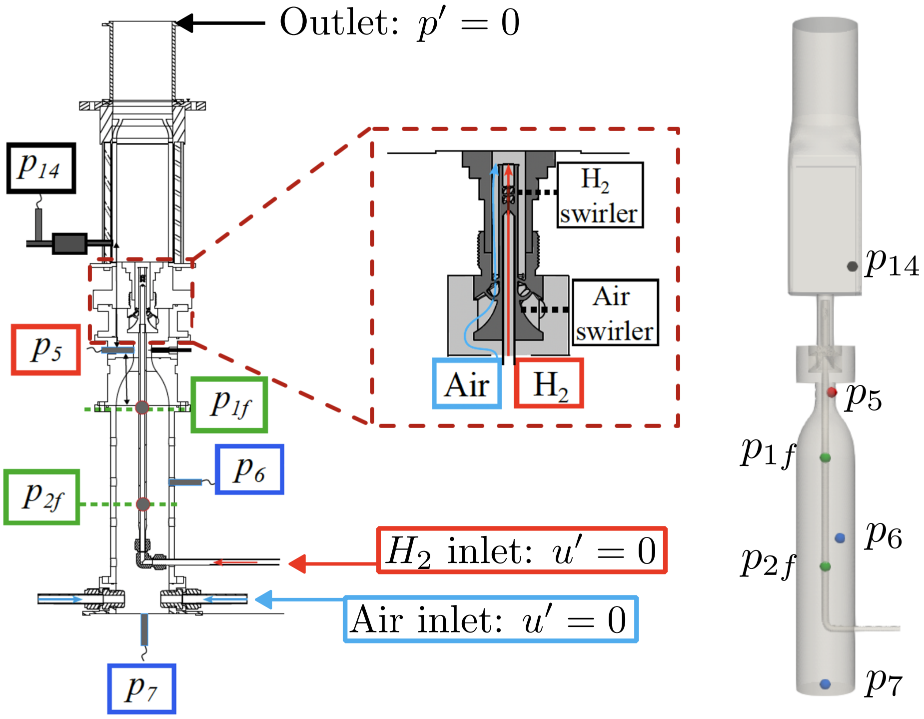


Figure 5.8: Left: Experimental setup studied at IMFT, including a detailed view of the HYLON burner. The inlet and outlet boundary conditions used in the simulation are also shown. Right: Computational domain with the locations of the microphones.

The two inlet lines Air/H_2 are well separated to prevent the gases from mixing before combustion. Several microphones are placed in the setup to measure the instantaneous pressure at different locations. Microphones p_6 and p_7 are located in the plenum, p_{1f} and p_{2f} are in the fuel line (H_2), p_5 is also in the plenum near the converging section just before the air swirler, and p_{14} is placed in the combustion chamber after the injector.

5.2.1 Cold case validation

We first compare the passive acoustic modes observed in the experiment with those predicted by STORM (specifically using FEniCS for the passive simulation). For this validation, both the experiment and the simulation involve only air in the domain (no H_2), and the setup is at ambient temperature, meaning there is no flame present. The value of sound speed is therefore set uniform, with value $c_0 = 347.12\ m.s^{-1}$. The boundary conditions used in the simulation (which are the same as those in the experiment) are given in Fig. 5.8.

The results in terms of frequencies for the first four acoustic modes are provided in Tab. 5.2.

Table 5.2: Frequencies of the first four acoustic modes in the passive simulation of the HYLON configuration, from both experimental measurements (by H. Paniez, unpublished) and FEniCS calculations.

Mode index N	Experimental measurements $[Hz]$	FEniCS solver $[Hz]$
1	66	64
2	166	189
3	256	263
4	420	446

The experimental pressure measurements at the different probe locations are compared with the pressure signals computed using FEniCS. The results are presented in Fig. 5.9.

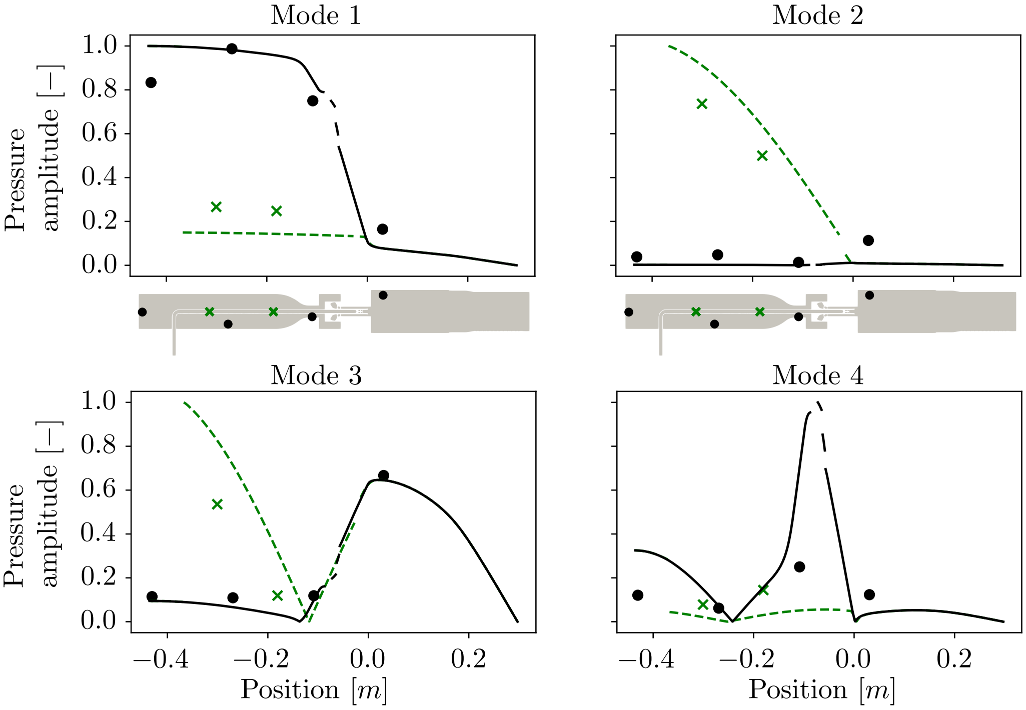


Figure 5.9: Green dashed line: Mode shape computed with FEniCS, plotted along a line that passes through the fuel line. Full black line: Same plot, but along a line passing through the plenum and the combustion chamber. Green crosses: Experimental pressure measurements at microphones p_{1f} and p_{2f} (fuel line). Black dots: Experimental pressure measurements at microphones p_7, p_6, p_5 and p_{14}. The middle figure shows the microphone positions within the geometry as a visual reference.

The discrepancies observed can be explained by some simplifying assumptions. For instance, acoustic losses in the geometry - especially around the injector - are not accounted for. Additionally, the boundary conditions in the simulation are fully reflective, whereas in the real experiment, partial reflections may occur. These could be better represented using impedance boundary conditions. Even though the experiment reflects the real physical behavior, some discrepancies may also arise from uncertainties in the pressure measurements themselves, due to the limitations and precision of the measurement devices. Overall, the experimental and numerical results are consistent, which allows us to proceed to the next part of the study: the flame response in hot conditions.

5.2.2 Double entry FTF

The combustion of hydrogen is an increasingly studied subject, particularly because it is considered "cleaner" from an emissions standpoint. The experimental burner HYLON has been built to study the behavior of non-premixed flames composed of air and hydrogen (H2). Besides the experimental and/or numerical challenges related to turbulent hydrogen combustion, the addition of a hydrogen injection line separate from the air injection questions the classical $N-\tau$ FTF model. Indeed, this model assumes that the unsteady rate of heat release Q' depends only on an acoustic velocity perturbation measured at a certain reference point. In the case of dual air/hydrogen injection, one can attempt to model the heat release in a way that accounts for two inlet velocities, each measured in the two different lines. In the work of Polifke [171], the flame is modeled as a Multiple Input Single Output (MISO) system, where Q' responds to equivalence ratio fluctuations on top of velocity fluctuations. ϕ' cannot be related in a straightforward manner to the velocity at the burner exit, but instead are coupled to the acoustics of the entire combustion system, including the fuel lines.

The double input / single output system has been implemented in the STORM framework (Eq. 5.1), but the methodology can be extended to any number of inputs. In the case considered, inputs are from the two inlet lines, air and H2.

$$Q' = \overline{Q}\left(N_{air}e^{j\omega\tau_{air}}\frac{u'(x_{air})}{\overline{u}_{air}} + N_{H_2}e^{j\omega\tau_{H_2}}\frac{u'(x_{H_2})}{\overline{u}_{H_2}}\right) \tag{5.1}$$

5.2.3 Pseudo-analytical solutions

Before performing a simulation of the double FTF with the modal expansion framework, we first computed a simplified configuration trackable analytically to verify the numerical implementation. The schematic representation of the case studied is depicted in Fig. 5.10.

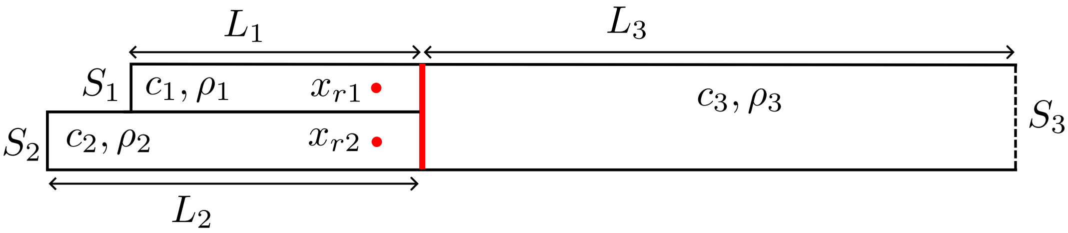


Figure 5.10: Schematic representation of the 1D test case used to validate the numerical implementation of the double entry FTF.

The single flame is positioned at the outlet of the two inlet lines, and is represented by the red line in Fig. 5.10. x_{r1} and x_{r2} are the two reference points for velocity perturbations in the double entry FTF. L_i and S_i are respectively the length and surface of each ducts. c_i and ρ_i are the sound speed and density. The value of all these parameters are summarized in Tab. 5.3. The boundary conditions are set as walls at the inlet of the two fuel lines, and open atmosphere and the outlet of the combustion chamber.

Table 5.3: Geometrical and physical parameters of the test case depicted in Fig. 5.10

Geometrical	Physical	FTF	
$L_1 = 308\ mm$	$c_1 = 1299.36\ m.s^{-1}$	$x_{r1} = 306\ mm$	$N_1 = 1$
$L_2 = 535\ mm$	$c_2 = 336.832\ m.s^{-1}$	$x_{r2} = 533\ mm$	$N_2 = 1$
$L_3 = 765\ mm$	$c_3 = 371.788\ m.s^{-1}$	$\overline{Q} = 250\ W$	$\tau_1 = 1\ ms$
$S_1 = 81.7\ cm^2$	$\rho_1 = 0.084\ kg.m^{-3}$	$\overline{u}_1 = 1\ m.s^{-1}$	$\tau_2 = 0.7\ ms$
$S_2 = 78.2\ cm^2$	$\rho_2 = 1.25\ kg.m^{-3}$	$\overline{u}_2 = 1\ m.s^{-1}$	
$S_3 = 200\ cm^2$	$\rho_3 = 1.026\ kg.m^{-3}$		

Details of the pseudo-analytical method used to compute the eigenmodes for this configuration are provided in Appendix B.2. These solutions serve as reference results for the State-space approach presented in the next section.

5.2.4 State-space and network approach

The state-space of the double entry FTF flame takes as input vector U_f the output vectors from the 2 connected ducts where the reference points are placed (x_{r1} and x_{r2}). It also takes as input the output from the third duct where the flame is located: x_l.

Since the double-entry FTF (Eq. 5.1) contains two frequency-dependent functions (corresponding to the responses of each input), each of these functions is fitted separately using the Vector Fitting algorithm.. The fluctuating heat release Q' then has the form:

$$\begin{aligned} Q'(t) &= \frac{\overline{Q}}{\overline{u}_1}\left(R_{0,1}+\sum_{k=1}^{N_{pbf1}}\frac{R_{1,k}}{j\omega - p_{1,k}}\right)u_1'(\boldsymbol{x_{r1}},t) \\ &+ \frac{\overline{Q}}{\overline{u}_2}\left(R_{0,2}+\sum_{k=1}^{N_{pbf2}}\frac{R_{2,k}}{j\omega - p_{2,k}}\right)u_2'(\boldsymbol{x_{r2}},t) \end{aligned} \tag{5.2}$$

Based on the formalism developed for the single input FTF, the state-space of the double entry FTF is of size $N_{pbf1}+N_{pbf2}+6$. It is written:

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{X}_{f_1}\\ \boldsymbol{X}_{f_2}\end{bmatrix} = \begin{bmatrix}\boldsymbol{A}_{f_1} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{A}_{f_2}\end{bmatrix}\begin{bmatrix}\boldsymbol{X}_{f_1}\\ \boldsymbol{X}_{f_2}\end{bmatrix} + \begin{bmatrix}\boldsymbol{B}_{f_1} & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{B}_{f_2} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}u_1^{ref}\\ p_1^{ref}\\ u_2^{ref}\\ p_2^{ref}\\ u_3^{loc}\\ p_3^{loc}\end{bmatrix} \tag{5.3}$$

The output equation is the following:

$$\begin{bmatrix}(\gamma-1)Q'\end{bmatrix} = \begin{bmatrix}\boldsymbol{C_{f_1}} & \boldsymbol{C_{f_2}}\end{bmatrix}\begin{bmatrix}\boldsymbol{X_{f_1}}\\ \boldsymbol{X_{f_2}}\end{bmatrix} \tag{5.4}$$

For $i=1,2$, the matrices $\boldsymbol{A_{f_i}}$, $\boldsymbol{B_{f_i}}$ and $\boldsymbol{C_{f_i}}$ and vectors $\boldsymbol{X_{fi}}$ respectively correspond to the state matrix, input matrix, output matrix and state vector of the state-space associated with a single input FTF. This state-space stands for the definition of either 1D or 3D flames. All the matrices involved in this state-space are detailed in the Appendix A.5. This formalism is implemented in the STORM framework, and the results in terms of frequencies of the configuration depicted in Fig. 5.10 using the parameters of Tab. 5.3 are presented in Tab. 5.4.

The results show good agreement. Minor discrepancies can be attributed to the finite flame thickness in STORM, modeled with a Gaussian profile,

Table 5.4: STORM vs. analytical results for the frequencies and growth rates of the first 5 modes of the 1D configuration described in Fig. 5.10. The last column shows the error, computed as the relative Cartesian distance between STORM and the analytical solution in the frequency/growth-rate plane.

Mode	Analytical		STORM		Errors
index	f $[Hz]$	σ $[s^{-1}]$	f $[Hz]$	σ $[s^{-1}]$	[%]
1	80	-4.2	81	-4.7	1.5
2	182	4.3	183	4.0	0.4
3	316	-13	320	-15	1.6
4	475	3.2	476	1.4	0.4
5	545	7.9	550	11.4	1.1

and to the fact that the reference points are taken 2 mm upstream of the flame. In contrast, the analytical model assumes an infinitely thin flame with reference points located exactly at the flame position. Overall, these results validate the numerical implementation of the double-entry flame state-space. This framework is now applied in the next section, using experimentally measured flame transfer functions on the HYLON test rig.

5.2.5 HYLON results

As part of the HYLON project, experiments were conducted at IMFT to measure the flame response to an acoustic velocity perturbation, either in the air line or the H_2 line. These experimental measurements, conducted by H. Paniez (unpublished data), are used to construct two transfer functions, which are fitted using the Vector Fitting algorithm, in the same manner as for single-input FTFs. The causality parameter, which enforces a negative

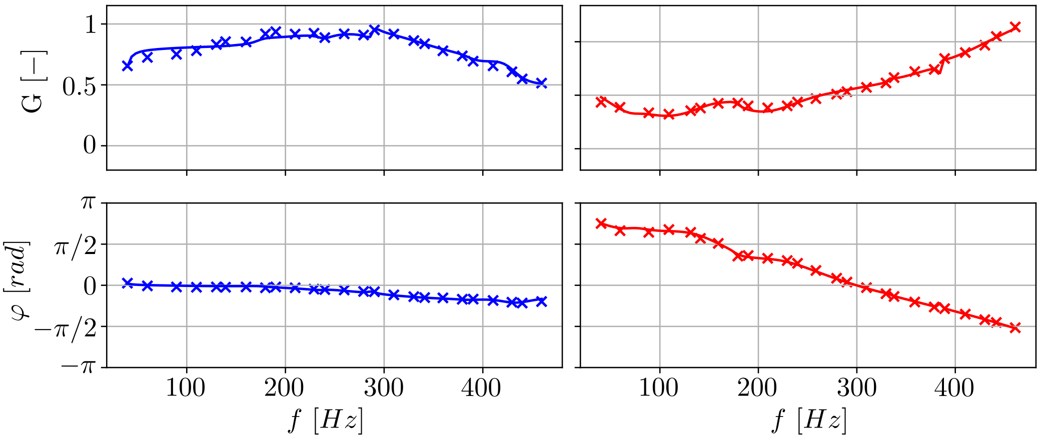


Figure 5.11: Crosses: Experimental measurements (conducted by H. Paniez at IMFT, unpublished). Solid lines: Fits obtained using the Vector Fitting algorithm. Left: Gain and phase of the flame response to an acoustic perturbation in the Air line. Right: Same quantities for an acoustic perturbation in the H_2 line. The fit orders are set to $N_{pbf,Air} = 16$ and $N_{pbf,H_2} = 15$.

real part for the poles, is activated. The experimental data and the resulting fits for the two transfer functions are presented in Fig. 5.11.

Since no impedance data are available at the outlet, an open-atmosphere boundary condition is applied at this location: $p' = 0$. All other boundaries are treated as solid walls, with a zero normal pressure gradient imposed: $\nabla_n p = 0$. The mean flow fields used in the simulation include a temperature jump from 293 K in the plenum to 900 K in the chamber. The resulting density field $\rho_0(\boldsymbol{x})$ is shown in Fig. 5.12. As the hydrogen line contains only H_2, its density remains low despite the relatively low temperature in that region.

The reference points $\boldsymbol{x_{r,Air}}$ and $\boldsymbol{x_{r,H_2}}$ for the flame response in the air and H_2 lines, respectively, are taken 6 mm upstream of the chamber in their corresponding line. The frame size is set to $N = 100$. Since only trivial boundary conditions are applied, the acoustic modes form an orthonormal

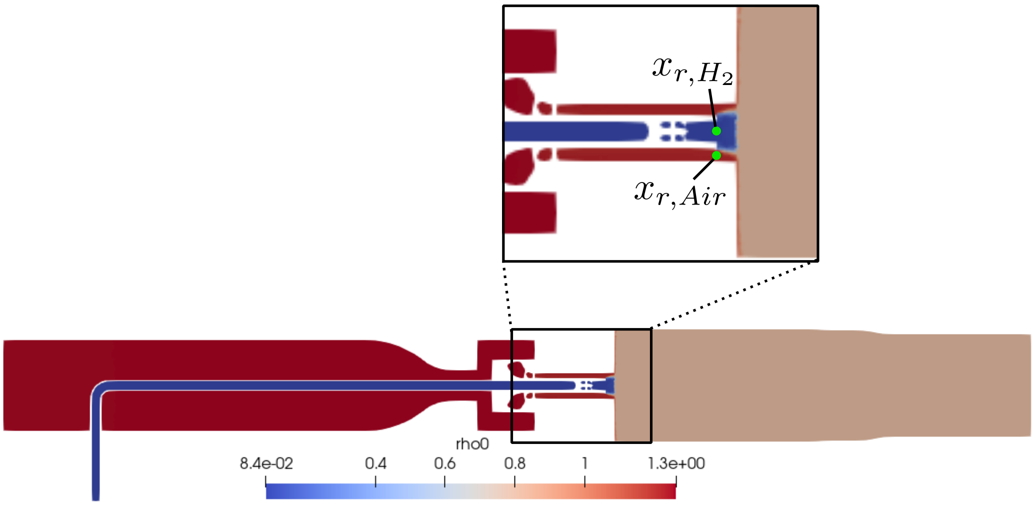


Figure 5.12: Density field $\rho_0(\boldsymbol{x})$ used in the simulation under hot conditions. The positions of the two reference points are also shown.

basis of the solution space. There is no need to construct a frame, and the use of SVD is not required. The mean flame power is $\overline{Q} = 5.2\ kW$, and the mean velocity in both lines are $\overline{u}_{Air} = 32.9\ m.s^{-1}$ and $\overline{u}_{H_2} = 18.9$ $m.s^{-1}$. Since two transfer functions are involved in the simulation, multiple fits must be performed with different orders (as explained in Chapter 3). A total of nine simulations are conducted to identify non-physical modes caused by the poles introduced by the fitting process. The fit orders used are: $N_{pbf,Air} = 14, 15, 16$ and $N_{pbf,H_2} = 14, 15, 16$.

The frequencies and growth rates then identified as physical by the method are presented in Tab. 5.5.

Although the experimental setup exhibits only one thermoacoustic mode around 430 Hz, the numerical simulation predicts three unstable modes at approximately 62 Hz, 302 Hz, and 447 Hz. The third mode matches well with the experimentally observed instability. The first mode, at 62 Hz, is only marginally unstable and could likely be stabilized by accounting for

Table 5.5: Frequency and growth rate obtained with STORM in the HYLON configuration, in the frequency range $[0, 500]$ Hz.

Mode index	f $[Hz]$	σ $[s^{-1}]$
1	62.5	3.52
2	302.9	13.57
3	447.8	14.33

acoustic losses that are neglected in the current model. The second mode, however, appears significantly unstable and is more difficult to explain. This could be attributed to several factors.

First, the numerical model uses idealized boundary conditions, such as fully reflective walls or pressure-released atmosphere, which may artificially enhance the acoustic feedback and promote instabilities that are otherwise damped in the real system. In reality, partial reflections and acoustic losses at the boundaries can stabilize certain modes.

Second, the mean flow used in the simulation is based on simplified assumptions, such as a stepwise temperature jump between the plenum and the combustion chamber. These approximations misrepresent the real temperature, density, or velocity gradients, thus modifying the coupling between pressure and heat release and potentially leading to instable modes.

Third, the flame model used in the simulation is based on a double-input Flame Transfer Function, built as the sum of the flame's responses to acoustic perturbations in the air and hydrogen lines, treated independently. In reality, there may exist cross-coupling effects between these two inputs that cannot be captured by a simple additive model. Introducing an explicit coupling term between the two inputs might improve the accuracy of the representation.

Fourth, the simplified treatment of the injector impedance may also contribute to the discrepancies. As discussed in Sec. 5.3 on the MICCA configuration, non-trivial acoustic impedances at the injector level can play a significant role in the stability of thermoacoustic modes. Neglecting these effects in the current model may lead to an inaccurate prediction of unstable modes.

Further investigations are required to fully validate these assumptions and better understand the origin and physical relevance of the additional unstable modes.

5.3 The MICCA test rig

The goal of this section is to investigate the MICCA configuration, a laboratory-scale annular combustor developed at the EM2C laboratory (CentraleSupélec, Université Paris-Saclay, CNRS). MICCA is specifically designed to study thermoacoustic instabilities in representative aero-engine geometries, particularly focusing on azimuthal modes. The experimental results used in this study are taken from the work of Latour et al. [172] and serve as a reference to validate the STORM simulation framework. The experimental setup is detailed in the paper previously cited.

In the experimental rig, each injector can be individually modified, and two types of injector are available: type U and type S. Since the flame response is affected by the injector characteristics, replacing one injector type with another (e.g., switching from S to U) allows for a parametric investigation of how the number and arrangement of each injector type influence the thermoacoustic instabilities. The different injector configurations tested experimentally are provided in Fig. 4 and Fig. 7 in [172] and are also recalled

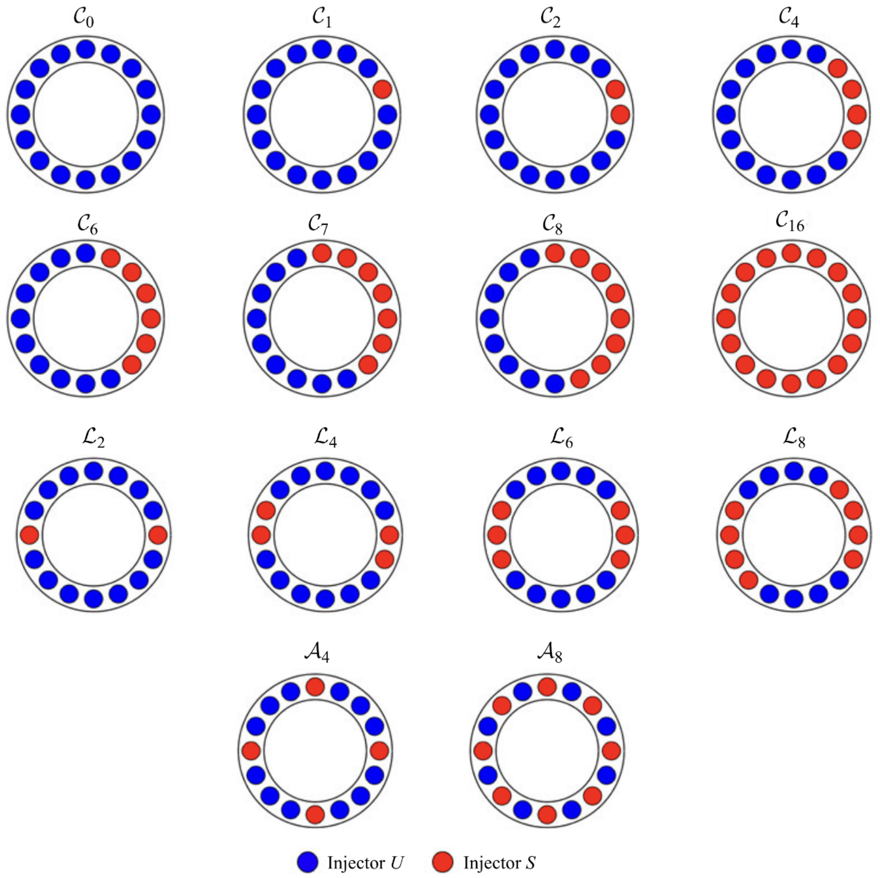


Figure 5.13: Injector configurations tested in the MICCA experiment. Image taken from [172].

in Fig. 5.13.

The objective is to assess the ability of STORM to reproduce the dominant thermoacoustic features observed experimentally in this laboratory configuration. The results using the STORM framework presented in this part was obtained in collaboration with Y. Gentil, a post doctoral researcher at CERFACS.

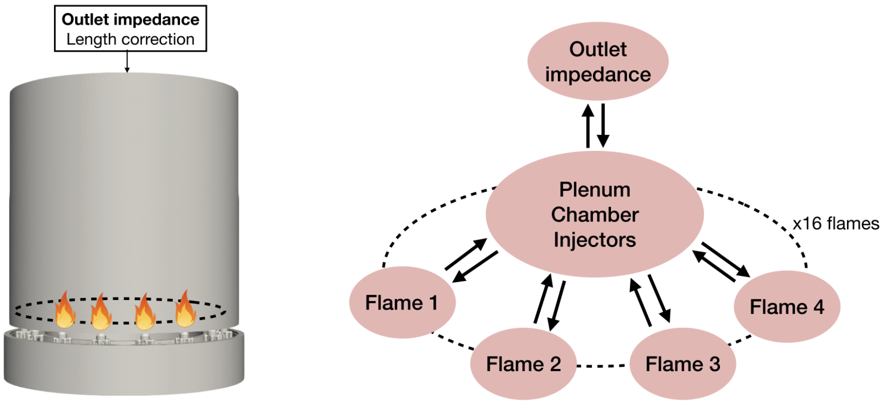


Figure 5.14: Illustration of the MICCA configuration: geometrical domain (left) and corresponding STORM network representation (right). In the left image, the flames are shown schematically. In reality, they are modeled as spherical regions located just downstream of each injector.

5.3.1 Geometry, FTFs, and network representation

A first simulation is performed under hot conditions. The network representation and the geometry of the MICCA configuration are shown in Fig. 5.14.

The mean fields used in this simulation include a temperature jump between the plenum and the chamber. The sound speed is set to $c_p = 346$ $m.s^{-1}$ in the plenum and $c_c = 776$ $m.s^{-1}$ in the combustion chamber. An end-length correction is applied at the chamber outlet, consistent with the treatment proposed by Latour *et al.* [172]. To model this boundary, the specific impedance expression from [173] is fitted using the Vector Fitting (VF) algorithm with two different orders, $N_{imp} = 4$ and 7. The resulting fit for order 4 is shown in Fig. 5.15. For this simulation, the frame size is set to $N = 400$, and the number of spectral modes at the outlet is set to $N_K = 15$.

The flame transfer functions used to model the U and S injectors are measured experimentally. These are fitted using the Vector Fitting algorithm with different orders $N_{ftf} \in [7, 8, 9]$, and the resulting fits for order 7 are

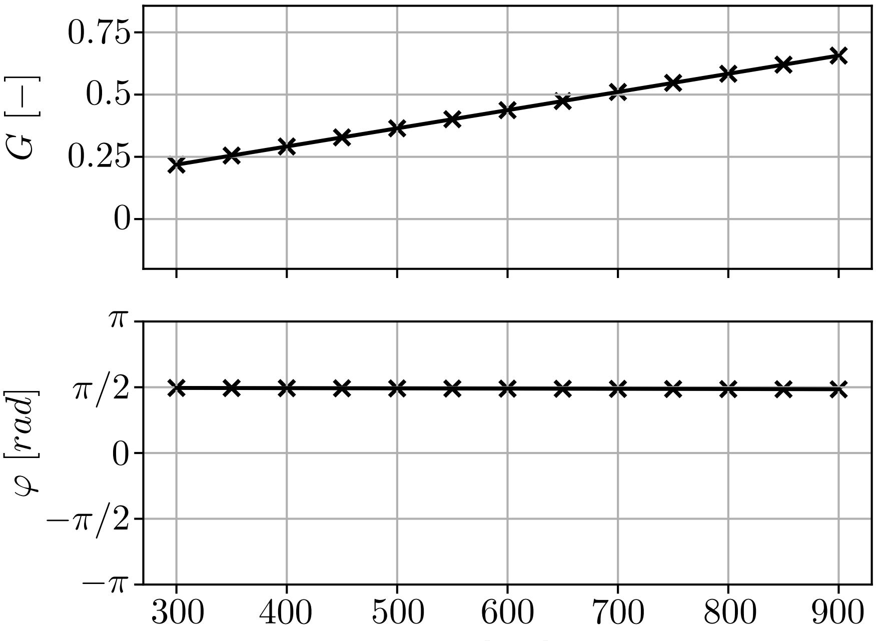


Figure 5.15: Impedance fitting for the outlet boundary condition using the Vector Fitting algorithm (order 4 shown). The solid line corresponds to the fitted model, while the crosses represent the experimental impedance data.

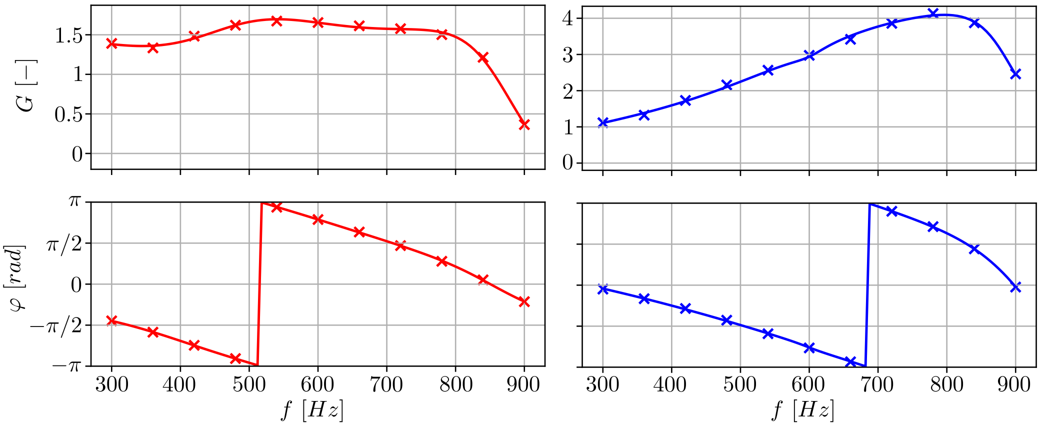


Figure 5.16: Flame Transfer Function fitting for U (right) and S (left) injectors using the Vector Fitting algorithm (order 7 shown for both cases). Solid lines show the fitted models, while the crosses represent the experimental data.

shown in Fig. 5.16.

5.3.2 Preliminary results

The results obtained from STORM are compared with both experimental data and an analytical model developed in [172]. In terms of frequency, the experimental setup allows for precise measurement of the thermoacoustic modes, which can be directly compared to the eigenfrequencies predicted by STORM. However, growth rates cannot be directly measured in the experiments, as only the amplitude of the limit cycle is observable. Since STORM does not predict limit cycle amplitudes, a direct comparison with growth rates is not possible.

Fortunately, the experimental study includes a validated ad hoc low order model that provides an estimate of the growth rates of the modes. This model serves as a reference for assessing the accuracy of the growth rates predicted by STORM. The comparison of frequencies (with respect to experimental measurements) and growth rates (with respect to the analytical model) is

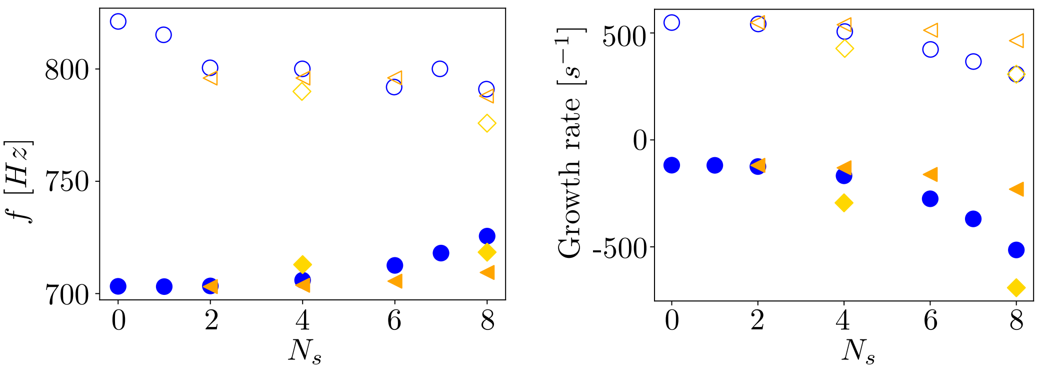


Figure 5.17: Comparison between STORM predictions and reference values for the MICCA configuration. Left: Frequencies compared with experimental measurements taken in [172]. Right: Growth rates compared with the analytical model from Latour *et al.* [172]. Empty markers correspond to the reference values from the cited paper, while filled markers show the results obtained using the STORM framework. All values are plotted as a function of N_s, the number of S-type injectors in the configuration. Blue circles represent configuration type C, orange triangles type L, and yellow diamonds type A. The different injector configurations are described in Figure5.14.

presented in Fig. 5.17.

Although the predicted frequencies are within the correct order of magnitude (700 vs 800 Hz), the trend observed when varying the number of S-type injectors is not correctly captured (STORM predicts that the frequency of oscillation increases with N_s while the measurements show the opposite), which is a significant issue. Moreover, the predicted growth rates follow the correct trend compared to the analytical model from Latour et al. [172], even though the absolute values appear to be shifted by a significant offset of order 600 s^{-1}.

One possible explanation for these discrepancies is the acoustic transfer function of the injectors themselves. Since the injectors are not acoustically transparent, their influence on wave propagation must be accounted for. This aspect is investigated in the following section.

5.3.3 Influence of Injector Acoustic Transfer

One important phenomenon that can affect acoustic propagation in combustors involves the interaction between acoustics and vorticity. This occurs in localized regions of low velocity, particularly near the injector trailing edges, where shear layers develop due to fluid viscosity. When subjected to acoustic forcing, these shear layers are modified, and energy is transferred from the acoustic field to vortices [174, 175], effectively introducing an acoustic damping mechanism.

This phenomenon is frequency dependent and exhibits a linear behavior at low acoustic amplitudes, transitioning to a nonlinear regime as amplitudes increase [176]. As pointed out by Latour *et al.* [177], such flow regions, especially within swirlers, are “weakly transparent to acoustic waves”, and accurately modeling their effect is critical for reliable thermoacoustic predictions.

Although this effect is not directly captured in the inviscid Helmholtz equation, it can be incorporated using an *Acoustic Transfer Matrix* (ATM) approach. The ATM framework connects upstream and downstream acoustic quantities (p' and u') across a compact element using a transfer matrix, and has been implemented in advanced Helmholtz solvers [81, 178]. In particular, Ni *et al.* [81] demonstrated the relevance of this model for swirlers, dilution holes, and perforated plates in a combustor geometry developed by SAFRAN.

While the ATM approach has already been incorporated into finite element Helmholtz solvers, it had not yet been fully implemented within the framework of modal-expansion-based methods. This section explores such an implementation within the STORM framework.

Acoustic Transfer Matrix implementation

Before applying the Acoustic Transfer Matrix (ATM) model to the full 3D MICCA configuration, a dedicated test case is used to validate its numerical implementation in the STORM framework. This verification step is essential to ensure that the model has been correctly integrated into the modal expansion solver.

The ATM formulation implemented in STORM is expressed as follows:

$$\begin{bmatrix} \hat{p}_d \\ \rho_0 c_0 \hat{u}_d \end{bmatrix} = \begin{bmatrix} T_{11}(\omega) & T_{12}(\omega) \\ T_{21}(\omega) & T_{22}(\omega) \end{bmatrix} \begin{bmatrix} \hat{p}_u \\ \rho_0 c_0 \hat{u}_u \end{bmatrix} \tag{5.5}$$

where $T_{11}(\omega)$, $T_{12}(\omega)$, $T_{21}(\omega)$, $T_{22}(\omega)$ are frequency-dependent transfer functions. The subscripts d and u refer to downstream and upstream quantities. These functions can be derived from experimental data, Large Eddy Simulations (LES), or analytical models. Since they are functions of pulsation ω, they are fitted using the Vector Fitting algorithm, following the same procedure as for impedances and Flame Transfer Functions. In this formulation, ρ_0 and c_0 denote the average density and sound speed between the two domains connected by the transfer matrix.

The state-space representation of this transfer relation is provided in Appendix A.7.

To validate this implementation, we use an academic 1D test case, illustrated in Fig. 5.18. The configuration is a 1D duct of length $L = 1\ m$, divided into two segments: a 0.4 m upstream duct and a 0.45 m downstream duct. These two segments are connected through a transfer matrix that models the acoustic propagation over the remaining 0.15 m of the domain, such that the total physical length remains 1 m. The boundary conditions, as well as the values of the density ρ_0 and sound speed c_0, are indicated in the figure. Since the transfer corresponds to a 1D wave propagation, the analytical expressions

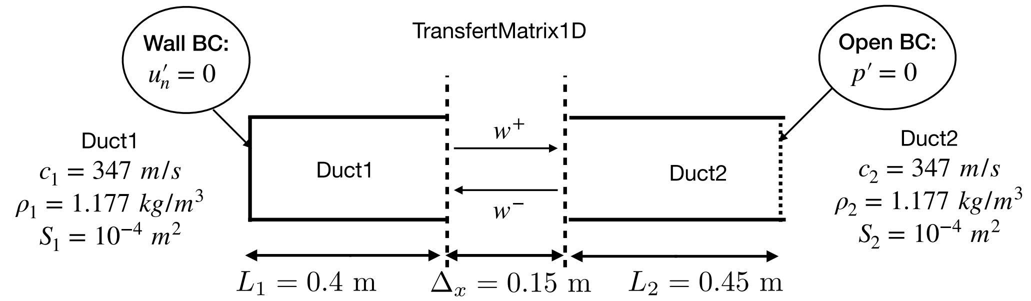


Figure 5.18: Schematic of the 1D validation configuration. The domain 1D duct of total length $L = 1\ m$ divided into two segments of $0.4\ m$ and $0.45\ m$, connected by an Acoustic Transfer Matrix modeling a $0.15\ m$ wave propagation. The values of density ρ_0 and sound speed c_0 are specified, along with the boundary conditions used for the simulation.

of the four transfer matrix coefficients are known and given by:

$$\begin{cases} T_{11}(\omega) & = & T_{22}(\omega) & = & \cos(k\Delta_x) \\ \\ T_{12}(\omega) & = & T_{21}(\omega) & = & -j\sin(k\Delta_x) \end{cases} \tag{5.6}$$

where $k = \omega/c_0$ is the wavenumber and $\Delta_x = 0.15\ m$ is the equivalent propagation length.

The results of this validation case are presented in Tab. 5.6. It reports the frequencies of the first seven acoustic modes computed using the STORM method with the implemented Acoustic Transfer Matrix model, and compares them to analytical solutions obtained for the full 1D tube.

The excellent agreement between both approaches confirms the correct numerical implementation of the state-space formulation associated with the transfer matrices.

Application of ATM model to MICCA

In this section, the Acoustic Transfer Matrix (ATM) model, numerically validated in the previous section, is applied to the MICCA configuration.

Table 5.6: Frequencies of the first seven acoustic modes computed with STORM using the Acoustic Transfer Matrix (ATM) model, compared with analytical results for the full 1D duct. Associated relative errors are shown in the last column.

Mode index	STORM [Hz]	Analytics [Hz]	Errors [%]
1	86.75	86.75	1.2×10^{-6}
2	260.25	260.25	4.8×10^{-9}
3	433.75	433.75	4.4×10^{-8}
4	607.25	607.25	7.7×10^{-9}
5	780.75	780.75	6.3×10^{-9}
6	954.25	954.25	3.1×10^{-9}
7	1127.75	1127.75	6.8×10^{-9}

This application is particularly relevant since, in the experimental study by Latour *et al.* [172], the four transfer matrix coefficients (T_{11}, T_{12}, T_{21}, T_{22}) were measured and are provided in the appendix of their article.

The dataset of their measurements, along with the corresponding Vector Fitting results, is shown in Fig. 5.19. In order to apply the non-physical mode identification criterion defined in Sec. 3.7.4, multiple fits with different orders were performed. The fitting orders used for the four coefficients T_{11}, T_{12}, T_{21} and T_{22} are: $N_{11} \in [4, 5, 6]$, $N_{12} \in [4, 5, 6]$, $N_{21} \in [5, 7, 9]$ and $N_{22} \in [4, 5, 6]$. These ranges were selected based on the low model-sample error observed between the Vector Fitting results and the original datasets.

To apply the acoustic transfer matrix formalism, the injectors must first be removed from the geometry, as illustrated in Fig. 5.20. This operation separates the 3D domain into two subdomains: the plenum and the combustion chamber. These two regions are then connected through the ATM model

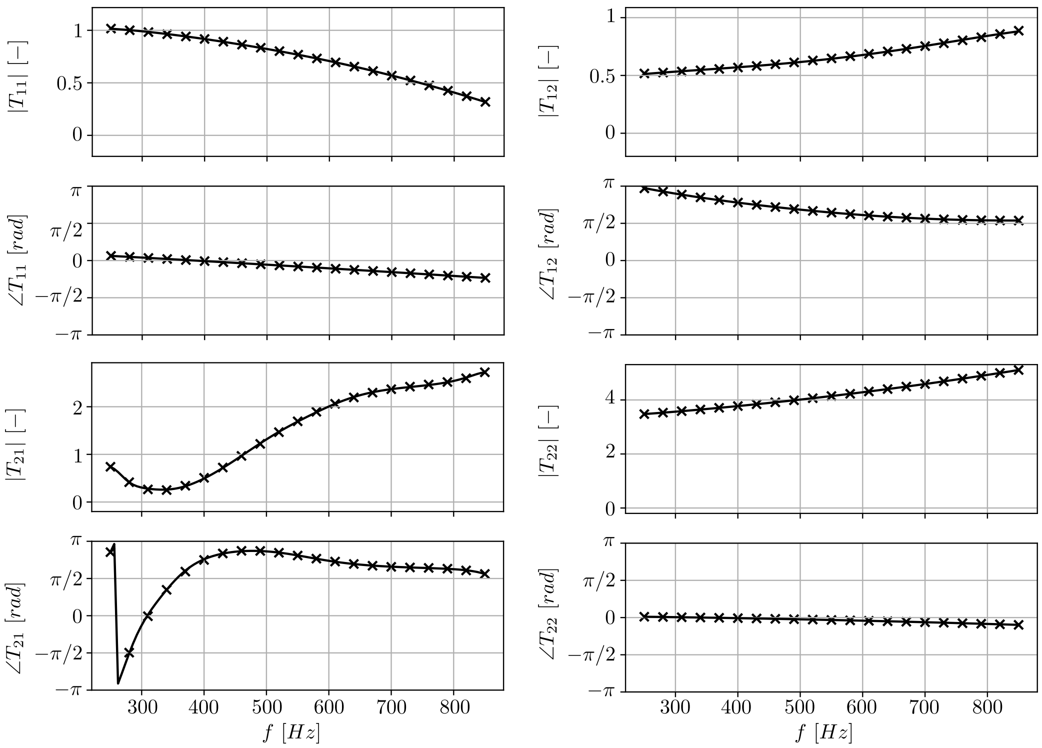


Figure 5.19: Transfer matrix coefficients (T_{11}, T_{12}, T_{21}, T_{22}) measured experimentally (crosses), and corresponding fits obtained using the Vector Fitting algorithm (solid lines). The fit orders are: 4 for T_{11}, 4 for T_{12}, 9 for T_{21}, and 4 for T_{22}.

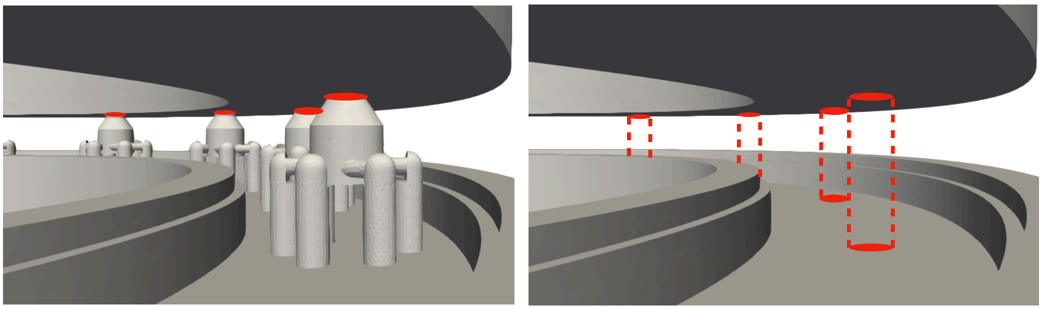


Figure 5.20: Modified MICCA geometry after removal of the injectors. This operation is necessary to apply the acoustic transfer matrix (ATM) formalism. The remaining domains are the plenum (upstream) and the combustion chamber (downstream), which will be connected via ATM blocks.

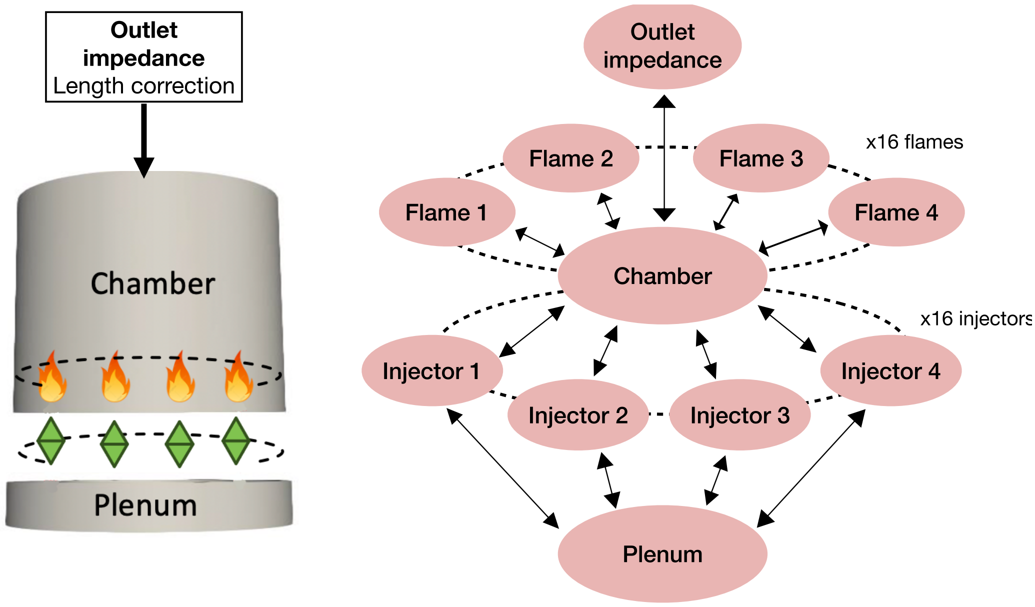


Figure 5.21: STORM network representation (right) and corresponding 3D geometry (left) after applying the ATM model between plenum and combustion chamber.

using two patches, following the connection method described in Sec. 4.2. The resulting STORM network, along with a visual representation of the split geometry, is shown in Fig. 5.21.

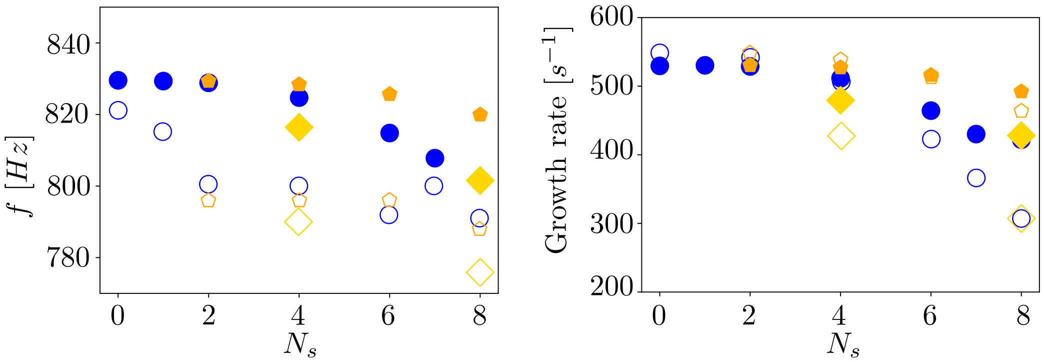


Figure 5.22: Comparison between STORM predictions using ATM and reference values for the MICCA configuration. Left: Frequencies compared with experimental measurements given in [172]. Right: Growth rates compared with the analytical model from Latour *et al.* [172]. Empty markers correspond to the reference values from the cited paper, while filled markers show the results obtained using the STORM framework. All values are plotted as a function of N_s, the number of S-type injectors in the configuration. Blue circles represent configuration type C, orange triangles type L, and yellow diamonds type A. The different injector configurations are described in Fig. 5.14.

The results obtained by including the acoustic transfer through the injectors are presented in Fig. 5.22. As in the comparison shown in Fig. 5.17, the predicted frequencies are compared against the experimental measurements, while the growth rates are compared to the analytical model developed and validated in the work by Latour *et al.* [172].

The results shown in Fig. 5.22 demonstrate a clear improvement once the non-acoustic behavior of the injectors is taken into account. This is particularly evident in the predicted frequencies. Not only they are closer in absolute value to the experimental measurements (with a maximum deviation of about 3%), but they also follow the correct decreasing trend as the number of S-type injectors increases - a behavior that was not captured without the transfer matrices (see left figure of Fig. 5.17). Similarly, the predicted growth rates are in better agreement with those obtained from the analytical model proposed by Latour *et al.* [172].

While in some practical applications such as MIRADAS (see Sec. 5.1) it may not be necessary to account for the acoustic transfer through the injectors, the MICCA setup provides a clear example where such effects cannot be neglected. This study has enabled the formulation and numerical implementation of the Acoustic Transfer Matrix (ATM) model, its validation on a simple 1D configuration, and its successful application to a complex experimental configuration.

5.4 Post combustion application

Post-combustion is a technique commonly used in military jet engines to provide a temporary thrust boost. It involves injecting additional fuel downstream of the turbine and burning it using the residual oxygen in the hot gases [179]. It is typically used for short durations during critical flight phases such as take-off, climb, landing, or during tactical manoeuvres. Post-combustion is a non-expensive solution for providing a thrust increase up to $\sim 70\%$. As the efficiency of such systems is comparatively lower than that of classical combustors, post-combustion systems are not an attractive option for civil aircraft applications, where fuel efficiency is of primary importance.

In addition to azimuthal and longitudinal instabilities [180], one of the most critical and damaging phenomena in afterburners is the occurrence of screech, a high-frequency self-excited thermoacoustic instability [181]. Screech in afterburners is typically associated with radial acoustic modes, unlike the azimuthal or longitudinal modes often observed in main combustors. This is due to the cylindrical geometry of the afterburner, which is completely filled with reactive gas. In contrast, main combustors usually have a toroidal or annular shape with central obstructions, which favor azimuthal/longitudinal modal structures.

5.4.1 Geometry

The configuration studied is a simplified geometry representative of the afterburner chamber of Safran's M88 engine. The geometry and the boundary conditions used for the simulation are provided in Fig. 5.23. This configuration is particularly relevant as it allows us to test the capabilities and limitations of the STORM method, and to explore how such a low-order approach can be applied to post-combustion systems. The mesh associated with this geometry contains approximately 1.5 million points.

The mean fields necessary to compute the frame was computed using LES by C. Mocquard. Details of the numerical model used for the LES can be found in [182]. The temperature and mean sound speed fields are shown in Fig. 5.24.

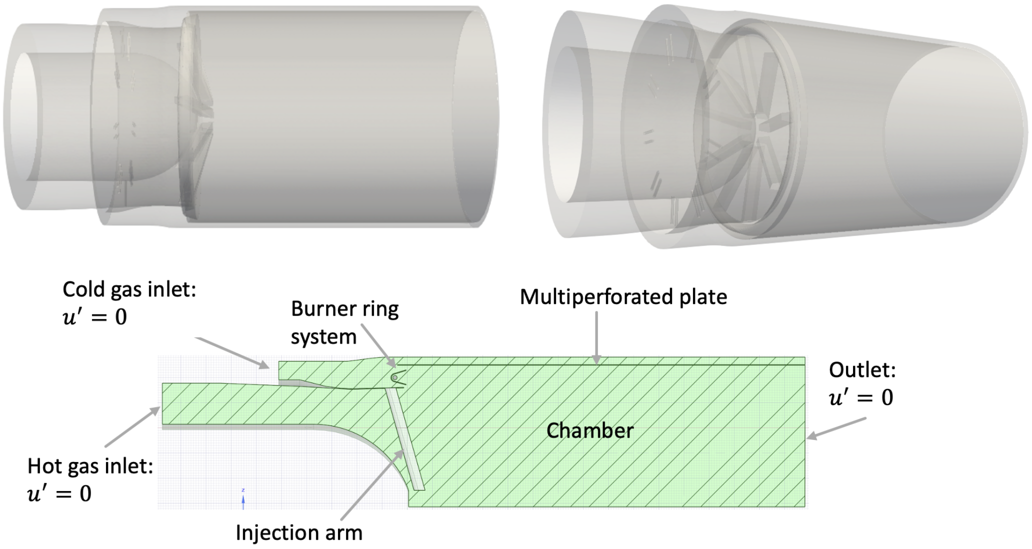


Figure 5.23: Geometry and boundary conditions used to simulate the afterburner chamber.

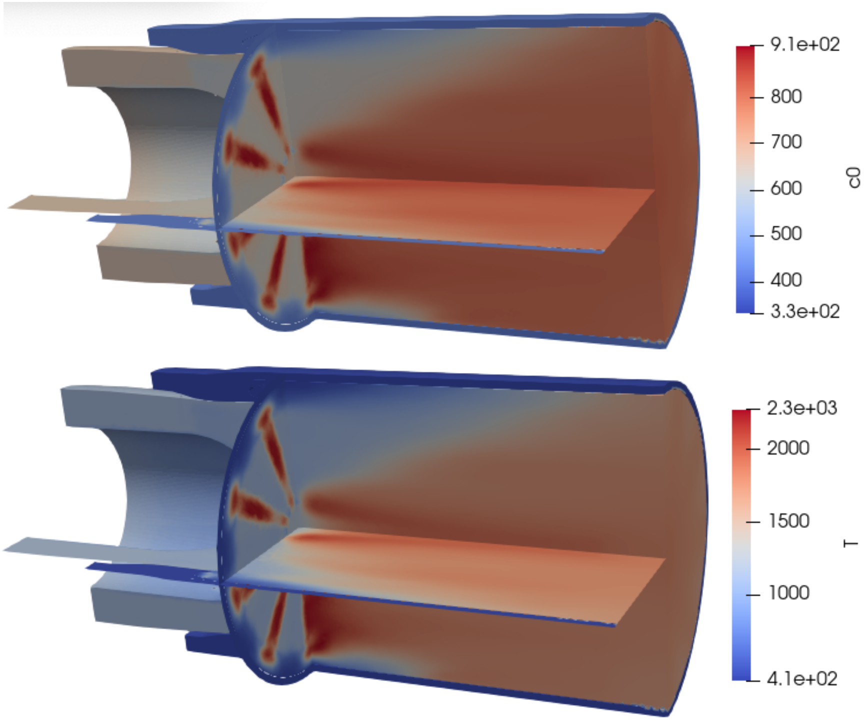


Figure 5.24: Mean temperature and sound speed computed using LES by the PhD student C. Mocquard, non-published.

5.4.2 Multiperforated plate modeling

As shown in Fig. 5.24, the sound speed field is non-uniform along the surface of the liner, and a temperature jump - and thus a jump in sound speed - is present across the liner interface. This configuration is in the scope of the validation previously performed on a simplified geometry (Sec. 4.3.4). However, a notable difference exists: the liner in this case does not separate two distinct 3D domains, but is instead embedded within a single continuous 3D domain. As a result, in the STORM network representation, there is only one 3D domain, which is connected to itself across the liner. This differs from the previously validated configuration, where the liner served as an interface between two clearly separated 3D domains.

The first step in assessing whether a liner model can be reliably applied is to verify that a simple 3D–3D connection, enforcing pressure and velocity continuity across the liner interface, works correctly. To do so, we compare the results obtained using the state-space 3D–3D connection based on spectral connections (Sec. 4.3.4) to those obtained from a simulation of the same geometry, with same mean fields but without the liner. The results in terms of frequencies are given in Tab. 5.7. Since no flame is present in this simulation and only trivial boundary conditions are applied, all growth rates are expected to be zero, and only the frequencies are considered for comparison. The frame size is set to $N = 1000$ and the number of spectral modes to $N_K = 200$.

The results show that the continuity condition across the liner interface is properly enforced, as the frequencies obtained with the connection closely match those from the reference case without the liner. This confirms that the 3D3D connection is properly implemented and operational, even in this complex configuration. However, for reasons not yet fully understood, non-

Table 5.7: Computed frequencies for the configuration shown in Fig. 5.23, based on the mean flow fields displayed in Fig. 5.24. Column 2: Results obtained by removing the liner interface from the global domain and serving as a reference. Column 3: Results obtained by applying a continuity condition (pressure/velocity) across the liner interface, via a 3D-3D connection. The x2 symbol indicates that the mode is degenerated.

Mode index	Reference case $f\ [Hz]$	Continuity condition $f\ [Hz]$
1	238.0	238.9
2	302.9	306.3
3	352.7 (x2)	356.4 (x2)
5	438.6	439.0
6	459.7 (x2)	460.5 (x2)
8	497.1 (x2)	500.1 (x2)
10	593.2 (x2)	600.6 (x2)
12	665.3 (x2)	667.9 (x2)
14	682.4 (x2)	689.3 (x2)
16	694.2	697.4
17	814.7 (x2)	817.1 (x2)

physical modes still appear in the output. This is illustrated in Fig. 5.25, which shows the frequencies and growth rates of all the modes returned by the STORM method in the range of frequency $[0, 950]$ Hz.

This graph shows that the physical modes are correctly captured by the modal expansion method when replacing the liner with a continuity condition (blue diamonds matching the red crosses). However, many non-physical modes appear in the output. These modes seem to occur in complex con-

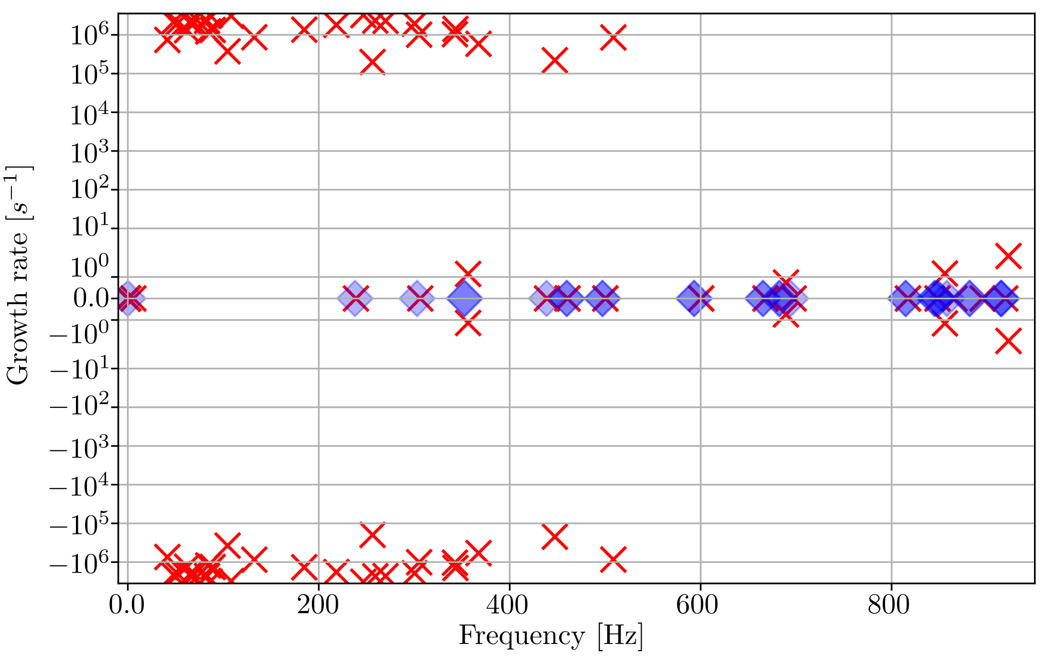


Figure 5.25: Frequencies and growth rates of modes in the configuration depicted in Fig. 5.23. Blue diamonds: Reference values. Red crosses: Results obtained with STORM by applying a continuity condition at the liner interface.

jugate pairs, i.e., each mode with a positive growth rate is mirrored by one with a negative growth rate of the same absolute value. Note that these non-physical modes still appear in the output, despite the application of the SVD-based frame conditioning described in Chapter 3.5.1, and the selection algorithm detailed in Section 4.2.2. This suggests that their origin lies elsewhere — possibly in the spectral mode conditioning rather than the frame itself. One hypothesis is that these spurious modes arise from the particular setup where the liner is embedded within the 3D geometry, instead of clearly separating it into two distinct domains. Additionally, geometric singularities at the corner of the liner may contribute to the emergence of these modes. This is visually evident when inspecting one of the non-physical modes (see Fig. 5.26), which exhibits a surface-like structure with high amplitude localized at the corner region.

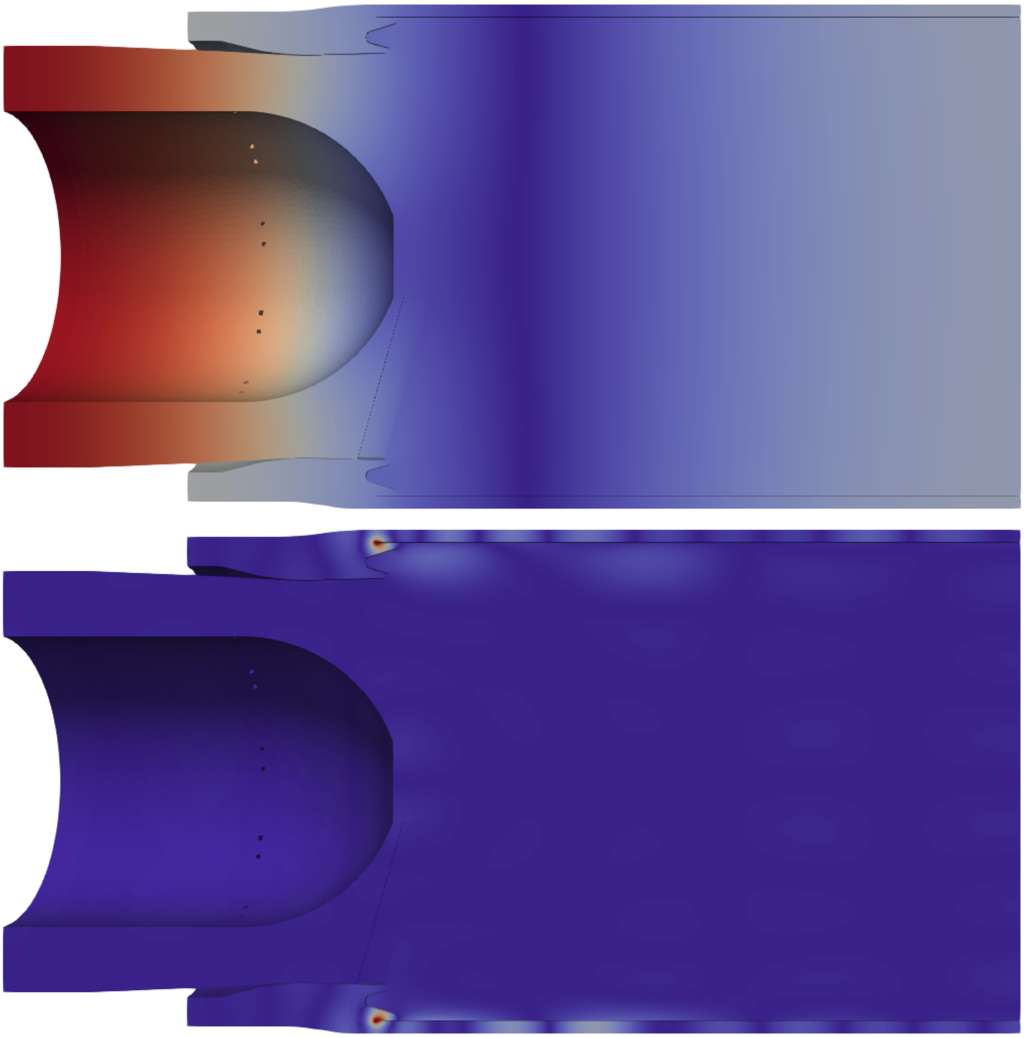


Figure 5.26: Top: Physical mode at 238 **Hz. Bottom: Non-physical mode. The figure highlights the singularity at the upstream edge of the liner in the geometry.**

Before these spurious modes are properly filtered out, it is not advisable to include a liner model at the interface, as it would introduce additional complexity and potentially amplify the influence of these non-physical modes. Similarly, adding a flame model at this stage would not be appropriate, since these spurious modes could couple with the flame response, making it even more difficult to identify and isolate non-physical modes a posteriori. Further investigations are necessary to understand the origin of these modes and to develop a robust method to remove them from the output.

However, the next section presents a flame simulation in such a complex case, where no Flame Transfer Function (FTF) is available and the flame present in the system is not compact.

5.4.3 Flame modeling - Parametric study

In the previous section, we discussed the challenges introduced by the presence of a multiperforated plate in the configuration, particularly from a numerical standpoint. In order to carry out a study focused on the flame dynamics and to avoid potential numerical artifacts associated with the plate, we consider a simplified geometry in which the multiperforated plate is removed. The next step consists in modeling the flame. The first requirement is to define its shape, which in this case is extracted from the LES simulation of C. Mocquard, where the heat release field is available. This field serves as the basis for identifying the flame region. A heat release threshold is applied: regions where the heat release exceeds a given value (here 10^8 $W.m^{-3}$) are considered to belong to the flame. This threshold is arbitrary and can be adjusted by the user. The resulting flame shape is shown in Fig. 5.27.

The post-combustor includes nine main injection arms arranged azimuthally, each generating a separate flame. We therefore model the flame as consisting of nine azimuthal sectors. Each sector is associated with a reference velocity probe located 4 cm upstream the injection arm.

Each of these nine flames is too long to be considered compact, as clearly seen in Fig. 5.27. A practical modeling approach is to split each flame longitudinally, for instance into three segments. An example of such segmentation

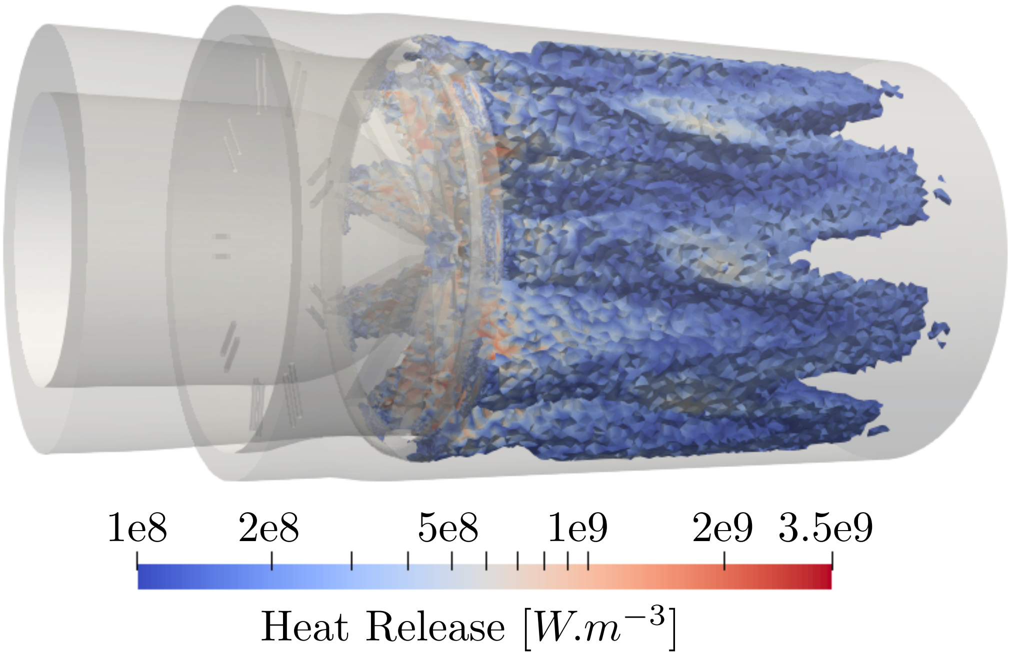


Figure 5.27: Flame shape extracted from LES simulation using a heat release threshold of 10^8 $W.m^{-3}$.

on a single sector is shown in Fig. 5.28.

In total, the flame system is composed of $9 \times 3 = 27$ flame segments. Because no experimental or LES-based FTF data is available for this configuration, we adopt a simplified modeling approach and perform a parametric study. We assume that all flames behave identically within each longitudinal segment. We also assume that these FTFs share a common gain N, while each has a distinct time delay τ_i, specific to the segment $i = 1, 2, 3$. The formulation for each FTF is:

$$\hat{q}_{i,k}(\omega) = \frac{\overline{Q}_i}{\overline{u}} N e^{-j\omega\tau_i} \hat{u}(x_{r,k}, \omega) \tag{5.7}$$

In this equation, $\hat{q}_{i,k}(\omega)$ represents the heat release rate fluctuation of the i-th flame segment in the k-th sector, which is modeled as proportional to the velocity fluctuation $\hat{u}(x_{r,k}, \omega)$ measured at a reference point $x_{r,k}$ just

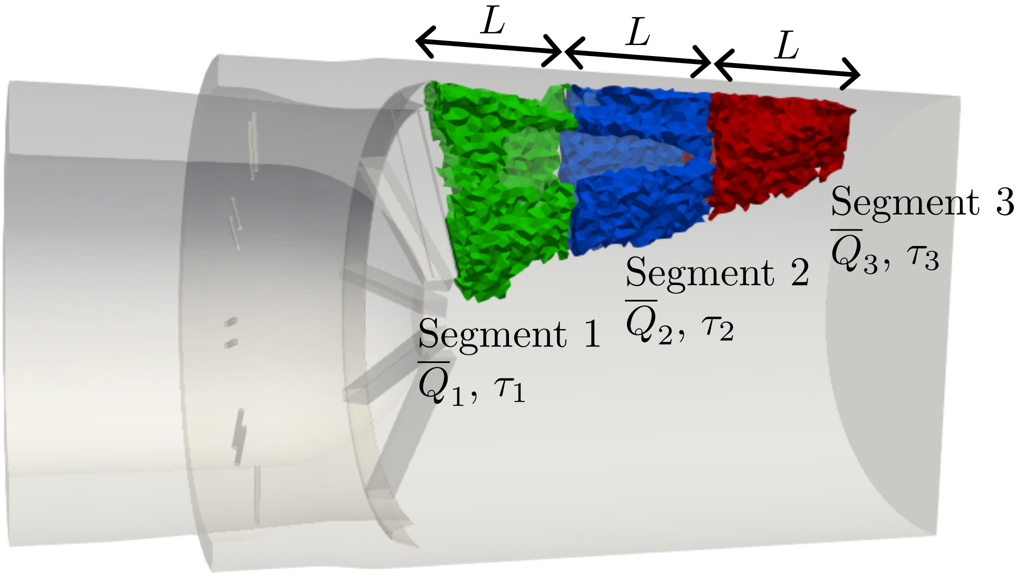


Figure 5.28: Example of the flame structure in one of the nine azimuthal sectors. The flame is segmented into three equal-length parts of $L = 0.22$ m for modeling purposes, with distinct colors used for visual clarity: green (first segment), blue (second), and red (third). Each segment is associated with a mean flame power $\overline{Q}_i$ and a time delay τ_i.

upstream of the k-th injector arm. $\overline{u} = 185$ m/s is the mean velocity, assumed identical for all flame segments and extracted from the LES. The mean heat release $\overline{Q}_i$ is segment-dependent: as we move downstream, each segment contributes less heat due to flame dilution. The heat release values, measured from LES integration, are: $\overline{Q}_1 = 1.94 \times 10^6$ W, $\overline{Q}_2 = 1.06 \times 10^6$ W, and $\overline{Q}_3 = 0.72 \times 10^6$ W. The time delays τ_i are linked to account for convective transport of perturbations along the flame. Considering each segment of the same length (here $L = 0.22$ m), they are defined as:

$$\tau_2 = \tau_1 + \frac{L}{\overline{u}} \text{ and } \tau_3 = \tau_2 + \frac{L}{\overline{u}} \tag{5.8}$$

The parametric study therefore focuses on two independent FTF parameters: the gain N and the initial delay τ_1. All segments share the same gain N, and the time delays are deduced from τ_1. The parameter space is explored by varying N from 0 to 3 in steps of 0.2, and τ_1 from 0.1 ms to 1 ms in steps of 0.1 ms, for a total of $15 \times 10 = 150$ thermoacoustic simulations.

As explained in Chapter 3, the STORM method requires several rational fits of each FTF to identify spurious modes. In this configuration, three distinct FTFs are used—one per flame segment stage—each associated with its own time delay. For each of these three FTFs, three different fits are performed, yielding a total of $3 \times 3 = 9$ rational approximations. The fit orders are adapted to the different time delays of the stages:

- $N_{pbf,1} = 18$, 19, 20 for the first-stage FTF
- $N_{pbf,2} = 26$, 27, 28 for the second-stage FTF
- $N_{pbf,3} = 34$, 35, 36 for the third-stage FTF.

These values are selected by identifying the minimum order at which the fit accurately captures the FTF dynamics, and then increasing it by one to allow the removal of non-physical modes, as detailed in Chapter 3. For each parameter pair (N, τ_1), 27 simulations are performed, resulting in a total of $27 \times 150 = 4050$ STORM runs to generate the full parametric plot (Fig. 5.29).

From a computational standpoint, the process is highly efficient: constructing the modal basis, common to all cases, requires about 30 minutes on 36 processors, while the full set of 4050 simulations is completed in roughly 1 hour on a single processor. This efficiency highlights a key advantage over classical finite element solvers, which typically require about one hour per mode and parameter point, making such studies practically infeasible with conventional methods.

The results of this parametric study, plotted in the frequency/growth rate plane, are presented in Fig. 5.29.

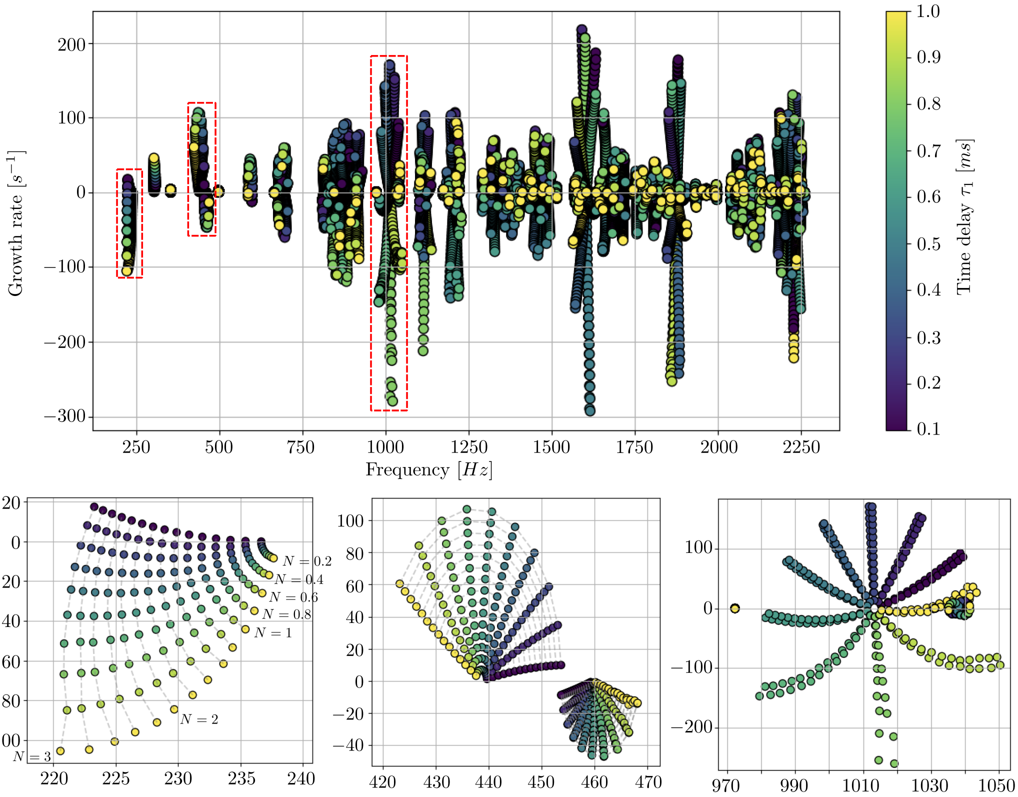


Figure 5.29: Results of the parametric study showing the computed thermoacoustic modes in the frequency–growth rate plane. Each point corresponds to a mode obtained from a STORM simulation using a specific combination of gain N and time delay τ_1, with $\tau_2 = \tau_1 + L/\overline{u}$ and $\tau_3 = \tau_2 + L/\overline{u}$ (with $L = 0.22$ m and $\overline{u} = 185$ m/s). All flame segments share the same gain N, varied from 0 to 3 by steps of 0.2, and the base time delay τ_1 is varied from 0.1 ms to 1 ms by steps of 0.1 ms. The three subplots below show zoomed-in views on selected regions (red dashed lines). These zooms provide better visibility of how growth rates evolve locally with varying FTF parameters. Gray dashed lines in the zoomed-in plots indicate levels of constant gain N.

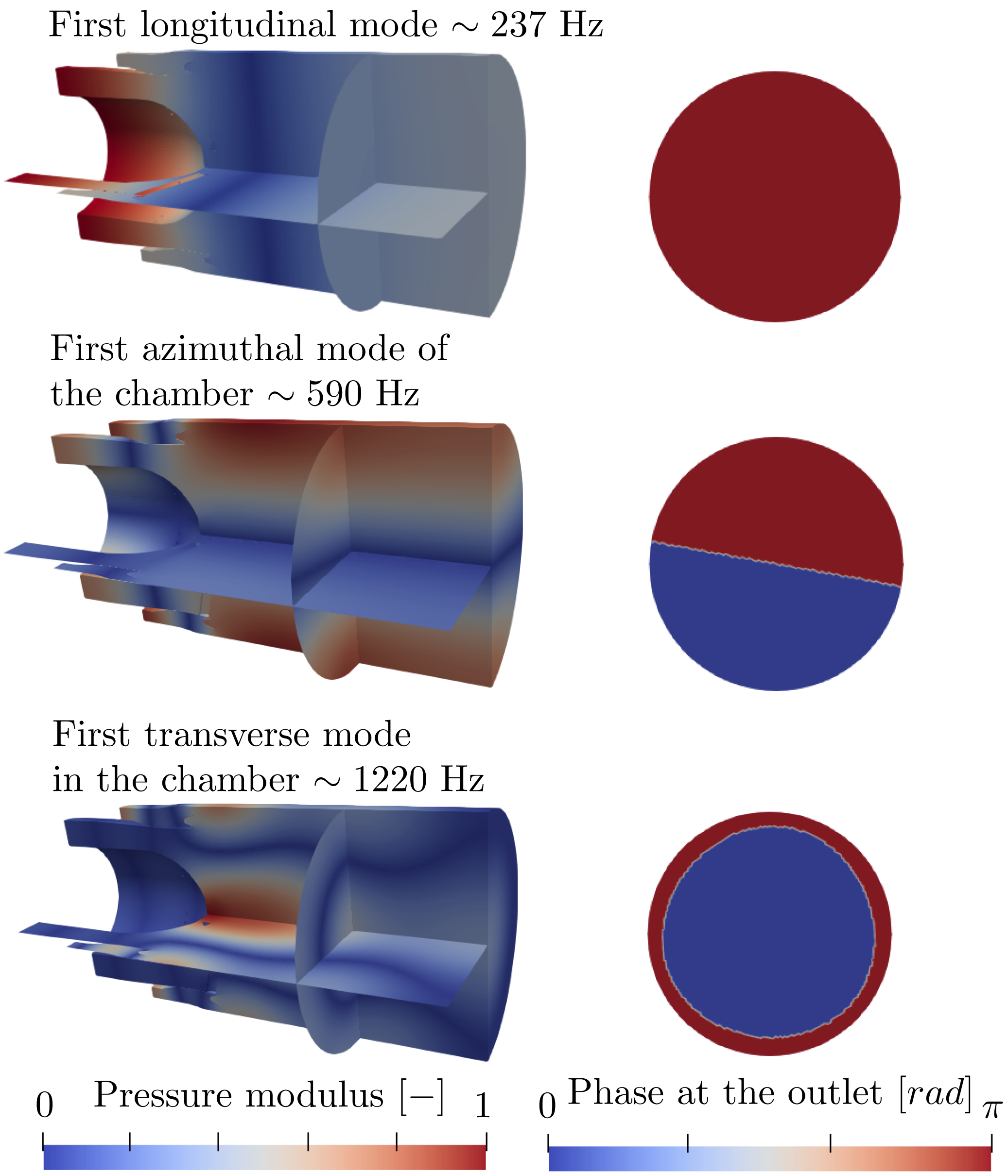


Figure 5.30: Representative acoustic modes of the chamber: the first longitudinal mode at ~ 237 Hz, the first azimuthal mode at ~ 590 Hz, and the first transverse mode at ~ 1220 Hz (i.e., the lowest frequency mode of each family). The left side shows the pressure magnitude distribution for each mode, while the right side displays their phase at the outlet of the combustion chamber.

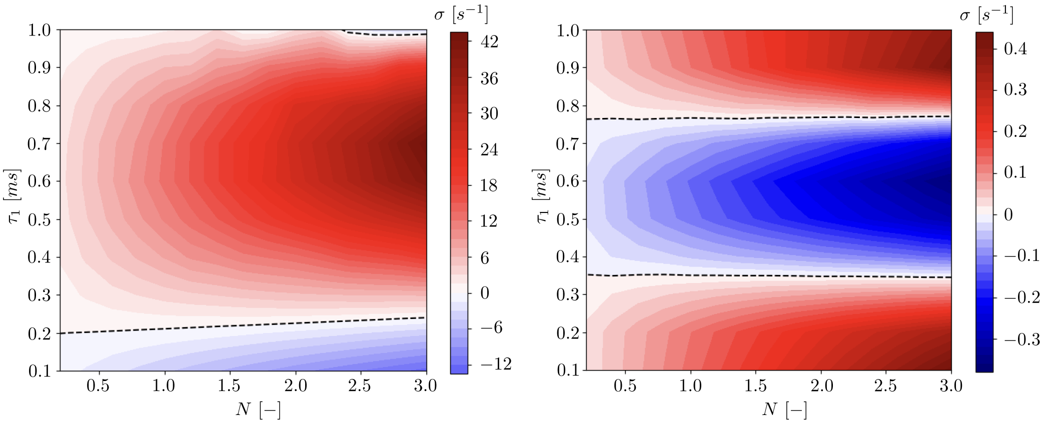
The resulting plot in Fig. 5.29 displays the typical trajectories of thermoacoustic modes as a function of FTF parameters. As expected, when the gain N varies, the eigenvalue trajectories follow straight lines converging at the frequency of the pure acoustic mode. When the time delay τ_1 varies, the trajectories form circular arcs around the same frequency. These trends are consistent with the theoretical behavior of $N-\tau$ constant FTF models [183].

Interestingly, even when using purely longitudinal FTFs based on the axial reference velocity, unstable transverse modes still appear in the simulations. For instance, Fig. 5.30 shows the first azimuthal mode of the chamber and the first transverse mode. The transverse unstable modes, commonly referred to as screech modes [184–186], are characteristic of post-combustion chambers and are known to be particularly damaging to chamber components. Capturing such modes is therefore of critical importance, and this study highlights the ability of STORM to detect them even when using simplified flame models.

Furthermore, the structure of the parametric plot in Fig. 5.29 reveals the presence of mode degeneracy due to the symmetry of the configuration. This is clearly visible in the zoom around 1015 Hz, where pairs of eigenvalues share nearly identical frequencies and growth rates. This degeneracy reflects the azimuthal symmetry of the geometry and is well captured by the model.

This parametric study also enables the generation of stability maps, which depict the stability of individual modes in the (N, τ_1) plane. Although such maps can be constructed for all identified modes, we focus here on two modes: the first azimuthal mode at approximately 590 Hz and the first transverse mode at approximately 1220 Hz (see Fig. 5.31).

Beyond these physical insights, this study also demonstrates the robustness of the mode identification procedure presented in Chap. 3. Across all



(a) Stability map for the first azimuthal mode at $\sim$ **590 Hz.**

(b) Stability map for the first transverse mode at $\sim$ **1220 Hz.**

Figure 5.31: Stability maps in the (N, τ_1) **plane for two representative modes. Red regions indicate instability with a positive growth rate** $(\sigma > 0)$**, while blue regions indicate stable conditions** $(\sigma < 0)$**.**

4050 simulations, no spurious modes were misidentified, which validates the effectiveness of the fitting and filtering methodology used.

Finally, while the FTF model adopted here is simplified due to the lack of experimental or LES-based FTF data, the resulting predictions already offer valuable design information. In practical terms, such stability maps can help engineers rapidly identify safe operating zones or detect parameter ranges prone to instability. Future studies may further refine this approach by incorporating nonlinear effects or cross-coupling between segments, but the present results already underscore the power and efficiency of the STORM solver for parametric thermoacoustic analysis.

5.5 Conclusion

This chapter presented the application of the STORM framework to four different experimental configurations, each introducing increasing levels of physical and numerical complexity.

The first part focused on the MIRADAS setup, where STORM yielded results in good agreement with experiments by using measured FTFs and boundary conditions. This configuration serves as a solid validation case, demonstrating that STORM can reliably handle 3D geometries with realistic flames and complex boundary conditions.

In the second part, the HYLON configuration is studied. It features a non-premixed flame fed by two distinct injection lines. A two-input FTF model was implemented in STORM and shown to be numerically consistent. However, the results did not match the experimental observations as closely. This discrepancy may arise from several factors, an example is the limitations of the two-input FTF model itself, which approximates the total flame response as the sum of its individual responses to each line, neglecting interaction effects.

The third part examined the MICCA setup, a 360° configuration with multiple flames. This case highlighted the ability of STORM to handle large and complex systems. It was shown that accounting for the injectors impedance is essential to accurately recover the experimental results. The implementation of transfer matrices was first validated on a 1D test case and then successfully applied to MICCA. This entire study was conducted mainly by Y. Gentil, a post-doctoral researcher at CERFACS. It was part of a preliminary effort to study the potential of STORM when used by an external scientist who was not involved in developing the solver.

Finally, the last configuration aimed to push the limits of the STORM methodology by applying it to a physically challenging geometry representative of a real afterburner. This test case introduces numerous difficulties, both physical and numerical, such as the absence of FTF or boundary condition data. However, the parametric study highlights the key advantage and novelty of the STORM framework: its ability to perform large-scale thermoacoustic analyses in a computationally affordable way. Covering a 2D space of 150 FTF parameter sets and performing 4050 full thermoacoustic simulations was achieved in under one hour on a single processor, after an initial modal basis computation that took only 30 minutes on 36 cores. Such an exploration would be practically impossible with conventional finite element solvers, which typically require an hour per point in the parameter space. Despite using simplified flame models, the study successfully captures important physical phenomena such as mode trajectories with respect to gain and time delay, the triggering of transverse screech modes, and modal degeneracy due to geometric symmetry. These results demonstrate not only the accuracy and flexibility of STORM, but above all its ability to unlock parametric design studies that were previously out of reach due to computational cost.

Overall, this chapter clarifies the current status of the STORM framework, what it can do, and where its limits lie. There are still important questions to be addressed, particularly regarding non-physical surface modes observed in the post-combustion simulation, which appear only when a liner is included in the geometry. In Section 4.5, a new approach was proposed to treat impedance jump conditions as source terms in the governing equations. This strategy has already been validated in 1D. Once fully implemented and validated in 3D, it could be applied to the post-combustion configuration to assess whether these non-physical modes disappear, paving the way for more accurate and robust simulations.

Chapter 6

General conclusion and perspectives

This thesis presents a comprehensive contribution to the development, validation and application of the STORM framework for the prediction of thermoacoustic instabilities in complex combustor systems. The work is structured around four main chapters, each addressing a critical component of the modeling chain.

Chapter 2 reviews the fundamental mechanisms of combustion instabilities and provides an overview of existing numerical methods for thermoacoustic stability analysis. Within this context, the STORM framework is introduced, highlighting its key strengths: a modal frame-based expansion strategy and a network-based formulation that enable efficient and flexible modeling of arbitrarily complex systems.

Chapter 3 focuses on the core numerical methodology of STORM: the frame modal expansion. While this approach offers computational efficiency, it also introduces numerical conditioning issues. A systematic method based on Singular Value Decomposition (SVD) was proposed to clean the modal basis, thus improving numerical robustness. This procedure is now an integral part of STORM. Additionally, the chapter details the treatment of flame dynamics using Flame Transfer Functions (FTF), including the automatic fitting process via Vector Fitting. Since such fitting operations introduces non-physical modes in the output, a dedicated method is developed to identify and remove them. This identification method is generic and is applied to all frequency-dependent models, including impedances, liner conductivity, Acoustic Transfer Matrices (ATM), etc. The chapter also justified the use of the FEniCS library for finite element assembly and provided code examples throughout.

Chapter 4 details how boundary and jump conditions in thermoacoustic systems are treated under the STORM framework. These conditions are essential, either to close the Helmholtz problem (boundary conditions) or to model physical effects not directly captured by the inviscid Helmholtz formulation (jump conditions), such as hydrodynamic-acoustic interactions, or liners. In 3D, these conditions are imposed using spectral connections, a technique originally introduced in STORM by C. Laurent [94] in 2021. Several numerical issues associated with these connections are identified, and practical solutions are implemented. Toward the end of the chapter, a new perspective is proposed: instead of relying on interface-based modeling via spectral connections, we introduce a volumic source-term formulation for jump conditions. This approach simplifies the domain decomposition, avoids the use of non-orthogonal frames, and provides a promising alternative for future developments. It is validated in 1D configurations, and further investigations are needed to extend it to realistic 3D setups.

Chapter 5 tests the current limits of STORM by applying it to a series of experimental and industrial configurations. This serves as a benchmark of the method's maturity and reveals new directions for improvement. While very good agreements are found with experimental results for the MICCA and MIRADAS setups, challenges remain for more complex cases such as HYLON and the afterburner configuration. The latter, in particular, reveals the presence of non-physical modes, despite prior validations of the spectral connections. Based on the structure of these modes, we hypothesize they originate from numerical conditioning issues in the spectral connection formalism. Two promising research directions are therefore identified: (i) improving the numerical robustness of the spectral connections, or (ii) transitioning to the newly introduced source-term-based jump condition approach to eliminate these spurious components. In this same chapter, a large-scale parametric study is conducted on the afterburner configuration, which demonstrates STORM's unique ability to explore a wide range of Flame Transfer Function parameters efficiently. Such a study would be practically impossible using a traditional finite element solver due to its prohibitive computational cost. This highlights one of STORM's key advantages: its ability to enable high-throughput stability analyses of realistic configurations at a fraction of the computational expense.

All the developments presented in this thesis are integrated into the STORM Python package, maintained at CERFACS and released under the CeCILL-B license. The package includes functional and integration tests to ensure the reliability of continuous development. It is fully documented on the CERFACS intranet, and provides a serial of 1D, 3D, and multi-domain tutorials, complete with schematics and automated scripts to facilitate user adoption. STORM is containerized for deployment on various HPC environments, using Docker or Singularity, which is also detailed in the documentation. It is important to note that this thesis is the third one which focuses on STORM, following those of C. Laurent and A. Badhe. Together, these contributions have significantly advanced the method's capabilities, robustness, and usability. Today, STORM is used by Safran Aircraft Engines to study thermoacoustic phenomena in their combustion chambers.

Although many axes for improvement remain, the current version of STORM provides an efficient and accurate tool for high-level parametric studies of thermoacoustic instabilities. It bridges the gap between low-order modeling and practical industrial application, offering a unique combination of flexibility, speed, and predictive power.

Appendix A

State-spaces realizations

Contents

A.1 Eliminating the feedthrough matrix in the State-Space formalism.

Let's assume that a complex-valued, time-dependent input variable $u(t)$ is in the input vector $\boldsymbol{U}_i$ of a given subdomain Ω_i, and must be used for the output equation $\boldsymbol{Y_i(t)} = \boldsymbol{C_i X_i(t)} + \boldsymbol{D_i U_i(t)}$. The idea is to incorporate this variable $u(t)$ in the state-vector $\boldsymbol{X}_i$ to simplify the output equation and make the Redheffer star product easier. As usual with the formalism adapted for the Helmholtz equation, variables of interest are decomposed into a static modulus and a temporal phase: $u(t) = |u|e^{j\omega t}$. Then, a new variable $x(t)$ is introduced, which satisfies the following differential equation:

$$\dot{x}(t) + \Omega_0 x(t) = \Omega_0 \underbrace{|u|e^{j\omega t}}_{u(t)} \tag{A.1}$$

where Ω_0 is a constant that will be well chosen. The solution of this equation is:

$$x(t) = Ae^{-\Omega_0 t} + \frac{\Omega_0}{j\omega + \Omega_0} u(t) \tag{A.2}$$

where A is an arbitrary constant (from the homogeneous solution). The variable $x(t)$ tends to $u(t)$ as Ω_0 tends to infinity: $\lim\limits_{\Omega_0 \to \infty} x(t) = u(t)$. Finally, the modified state-space can be written, by adding $x(t)$ to the original state-vector:

$$\frac{d}{dt}\begin{bmatrix} \boldsymbol{X}_i \\ x(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_i & \boldsymbol{0} \\ \boldsymbol{0} & -\Omega_0 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_i \\ x(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_i \\ \Omega_0 \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_i \\ u(t) \end{bmatrix} \tag{A.3}$$

The last line of the state-space represents the Eq. A.1. Therefore, any variable from $\boldsymbol{U}_i$ can be incorporated in $\boldsymbol{X}_i$ and the modified output equation is written:

$$\begin{bmatrix} \boldsymbol{Y}_i \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{C}_i & \boldsymbol{D}_i \end{bmatrix}}_{\widetilde{\boldsymbol{C}_i}} \underbrace{\begin{bmatrix} \boldsymbol{X}_i \\ \boldsymbol{U}_i \end{bmatrix}}_{\widetilde{\boldsymbol{X}_i}} \tag{A.4}$$

A.2 Geometrical block, truncated-SVD of the frame

A.2.1 1D case

Consider a 1D geometrical subdomain where the frame size is N. The truncated SVD is performed on this frame, which reduces the frame size to N_t. The modal expansion of the pressure field is written:

$$p'(x,t) = \sum_{n=1}^{N_t} \dot{\Upsilon}_n(t) U_n(x) \tag{A.5}$$

Recall that the U_n functions result from the orthonormalization of the original frame $(\phi_n)_{n\leq N}$. For simplicity, we consider a single one-dimensional connection (a section jump, for example) located at x_0, which introduces a normal velocity forcing $u_n^f(x_0,t)$ and an acoustic potential forcing $\varphi^f(x_0,t)$. Similarly, we assume that only one flame is present in the domain, positioned by a function shape $H(x)$ which generates a heat release denoted by $Q'(t)$. The formalism is generalizable to any number of flames or connections simply by adding each contribution.

Under these considerations, the temporal evolution of the modal amplitudes Υ_n is written:

$$\left\{ \begin{array}{lcl} \ddot{\Upsilon}_n(t) & = & -\sum_{k=1}^{N_t} [\boldsymbol{\Sigma V}^T \boldsymbol{\Omega}^2 \boldsymbol{V \Sigma}^{-1}]_{n,k} \Upsilon_k(t) \\ \\ & - & \rho_0 c_0^2 S U_n(x_0) u_n^f(x_0,t) \\ \\ & + & \rho_0 c_0^2 S \nabla_n U_n(x_0) \varphi^f(x_0,t) \\ \\ & + & (\gamma-1)\langle H|U_n\rangle Q'(t) \end{array} \right. \tag{A.6}$$

where ∇_n is the gradient operator projected onto the outward normal vector of the surface. Even in 1D cases, the cross-sectional area is taken into account, typically to enforce the continuity of mass flux. Here, this area is denoted by S. The dynamic equation A.6 is recast as a state-space formulation as follows:

$$\begin{cases} \dot{\boldsymbol{X}}_{\boldsymbol{1D}} &= \boldsymbol{A}_{\boldsymbol{1D}}\boldsymbol{X}_{\boldsymbol{1D}} + \boldsymbol{B}_{\boldsymbol{1D}}\boldsymbol{U}_{\boldsymbol{1D}} \\ \boldsymbol{Y}_{\boldsymbol{1D}} &= \boldsymbol{C}_{\boldsymbol{1D}}\boldsymbol{X}_{\boldsymbol{1D}} \end{cases} \tag{A.7}$$

where

$$\boldsymbol{X}_{\boldsymbol{1D}} = \begin{bmatrix} \vdots \\ \Upsilon_n(t) \\ \dot{\Upsilon}_n(t) \\ \vdots \end{bmatrix}, \boldsymbol{U}_{\boldsymbol{1D}} = \begin{bmatrix} u_n^f(x_0,t) \\ \varphi^f(x_0,t) \\ (\gamma-1)Q'(t) \end{bmatrix}, \boldsymbol{Y}_{\boldsymbol{1D}} = \begin{bmatrix} u_n'(x_0,t) \\ p'(x_0,t) \end{bmatrix} \tag{A.8}$$

$$\boldsymbol{A}_{\boldsymbol{1D}} = \begin{bmatrix} \vdots & \vdots & & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 1 \\ -[\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{\Omega}^2\boldsymbol{V}\boldsymbol{\Sigma}^{-1}]_{n,1} & 0 & \cdots & -[\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{\Omega}^2\boldsymbol{V}\boldsymbol{\Sigma}^{-1}]_{n,N_t} & 0 \\ \vdots & \vdots & & \vdots & \vdots \end{bmatrix} \tag{A.9}$$

$$\boldsymbol{B}_{\boldsymbol{1D}} = \rho_0 c_0^2 S \begin{bmatrix} \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ -U_n(x_0) & \nabla_n U_n(x_0) & \langle H|U_n\rangle \\ \vdots & \vdots & \vdots \end{bmatrix} \tag{A.10}$$

$$\boldsymbol{C}_{\boldsymbol{1D}} = \begin{bmatrix} \cdots & -\frac{1}{\rho_0}\nabla_n U_n(x_0) & 0 & \cdots \\ \cdots & 0 & U_n(x_0) & \cdots \end{bmatrix} \tag{A.11}$$

A.2.2 3D case

When considering a 3D geometrical domain, the previously established 1D state-space formulation must be adapted by incorporating the spectral connection formalism. The source term representing the flame is not modified and is therefore omitted here for simplicity. The necessary modifications appear only in the boundary forcing terms, specifically the acoustic potential forcing $\varphi^f(\boldsymbol{x_s}, t)$ and the normal velocity forcing $u_n^f(\boldsymbol{x_s}, t)$. These quantities are now approximated using a 2D modal expansion over the interface surface modes $(\mathcal{K}_k)_{k \leq N_K}$, as follows:

$$\left\{ \begin{array}{lcl} \varphi^f(\boldsymbol{x_s}, t) & = & \sum_{k=1}^{N_K} \nu_k(t) \mathcal{K}_k(\boldsymbol{x_s}) \\ \\ u_n^f(\boldsymbol{x_s}, t) & = & \sum_{k=1}^{N_K} \mu_k(t) \mathcal{K}_k(\boldsymbol{x_s}) \end{array} \right. \tag{A.12}$$

where $\boldsymbol{x_s}$ denotes the coordinate along the surface.

The temporal evolution of the modal amplitudes Υ_n is written:

$$\left\{ \begin{array}{lcl} \ddot{\Upsilon}_n(t) & = & -\sum_{k=1}^{N_t} [\boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{\Omega}^{\boldsymbol{2}} \boldsymbol{V} \boldsymbol{\Sigma}^{-1}]_{n,k} \Upsilon_k(t) \\ \\ & - & \rho_0 c_0^2 \sum_{k=1}^{N_K} \langle \mathcal{K}_k | U_n \rangle \mu_k(t) \\ \\ & + & \rho_0 c_0^2 \sum_{k=1}^{N_K} \langle \nabla_n U_n | \mathcal{K}_k \rangle \nu(t) \end{array} \right. \tag{A.13}$$

where $\langle \cdot | \cdot \rangle$ denotes the 2-dimensional inner product over the connection interface.

The state-equation is written:

$$\dot{\boldsymbol{X}}_{\boldsymbol{3D}} = \dot{\boldsymbol{A}}_{\boldsymbol{3D}}\boldsymbol{X}_{\boldsymbol{3D}} + \underbrace{\begin{bmatrix} \boldsymbol{B}_{1,1} & \cdots & \boldsymbol{B}_{1,N_K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{B}_{N_t} & \cdots & \boldsymbol{B}_{N_t,N_K} \end{bmatrix}}_{\boldsymbol{B_{3D}}} \underbrace{\begin{bmatrix} \boldsymbol{U}_1 \\ \vdots \\ \boldsymbol{U}_{N_K} \end{bmatrix}}_{\boldsymbol{U_{3D}}} \tag{A.14}$$

where

$$\boldsymbol{B_{n,k}} = \rho_0 c_0^2 \begin{bmatrix} 0 & 0 \\ -s_i\langle U_n|\mathcal{K}_k\rangle & \langle \nabla_n U_n|\mathcal{K}_k\rangle \end{bmatrix}, \boldsymbol{U_k} = \begin{bmatrix} \mu_k(t) \\ \nu_k(t) \end{bmatrix} \tag{A.15}$$

The state matrix/vector are unchanged compared to the 1D case: $\boldsymbol{A_{3D}} = \boldsymbol{A_{1D}}$ and $\boldsymbol{X_{3D}} = \boldsymbol{X_{1D}}$.

The output equation is written:

$$\underbrace{\begin{bmatrix} \boldsymbol{Y_1} \\ \vdots \\ \boldsymbol{Y_{N_K}} \end{bmatrix}}_{\dot{\boldsymbol{Y}}_{\boldsymbol{3D}}} = \underbrace{\begin{bmatrix} \boldsymbol{C}_{1,1} & \cdots & \boldsymbol{C}_{1,N_t} \\ \vdots & \ddots & \vdots \\ \boldsymbol{C}_{N_K} & \cdots & \boldsymbol{C}_{N_K,N_t} \end{bmatrix}}_{\boldsymbol{C_{3D}}} \boldsymbol{X_{3D}} \tag{A.16}$$

where

$$\boldsymbol{C_{k,n}} = s_i \begin{bmatrix} -\left\langle \dfrac{1}{\rho_0}\nabla_n U_n|\mathcal{K}_k \right\rangle & 0 \\ 0 & \langle U_n|\mathcal{K}_k\rangle \end{bmatrix}, \boldsymbol{Y_k} = s_i \begin{bmatrix} \langle u_n|\mathcal{K}_k\rangle \\ \langle p|\mathcal{K}_k\rangle \end{bmatrix} \tag{A.17}$$

In this equation, the quantity $s_i \in \{+1, -1\}$ is introduced to define the orientation of the geometrical domain connected to the spectral connection. This sign is particularly relevant when two 3D domains are connected through a shared interface. The value of s_i is automatically determined when the network is constructed: for each spectral connection, the user specifies the two connected geometrical domains. The first domain listed by the user is assigned $s_i = +1$, while the second is assigned $s_i = -1$.

A.3 3D3D spectral connections

The temporal evolution of modal amplitudes ν_k and μ_k associated with the 3D3D connection is written:

$$\left\{\begin{array}{rcl} \ddot{\nu}_k^i + \omega_k^2 \nu_k^i & = & -\dfrac{\overline{c_0}^2}{\lambda_k L_t}\left(s_i \langle u_n^i | \mathcal{K}_k^i \rangle + \sum_{l=1}^{N_K} \langle \mathcal{K}_l^j | \mathcal{K}_k^i \rangle s_j \langle u_n^j | \mathcal{K}_l^j \rangle\right) \\ \\ \dot{\mu}_k^i & = & \dfrac{1}{\lambda_k L_t \overline{\rho_0}}\left(s_i \langle p^i | \mathcal{K}_k^i \rangle + \sum_{l=1}^{N_K} \langle \mathcal{K}_l^j | \mathcal{K}_k^i \rangle s_j \langle p^j | \mathcal{K}_k^j \rangle\right) \\ \\ \nu_k^j & = & \sum_{l=1}^{N_K} \langle \mathcal{K}_l^i | \mathcal{K}_k^j \rangle \nu_k^i \\ \\ \mu_k^j & = & \sum_{l=1}^{N_K} \langle \mathcal{K}_l^i | \mathcal{K}_k^j \rangle \mu_k^i \end{array}\right. \tag{A.18}$$

The two last lines are the link between modal amplitudes associated with each boundary mesh $\mathcal{M}_i^b$ and $\mathcal{M}_j^b$ that defines the 3D3D connection.

The state-space realization is written:

$$\frac{d}{dt}\underbrace{\begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_{N_K} \end{bmatrix}}_{\boldsymbol{X}} = \underbrace{\begin{bmatrix} \boldsymbol{A}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \boldsymbol{A}_{N_K} \end{bmatrix}}_{\boldsymbol{A}} \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_{N_K} \end{bmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{B}_{1,1} & \cdots & \boldsymbol{B}_{1,N_K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{B}_{N_K} & \cdots & \boldsymbol{B}_{N_K,N_K} \end{bmatrix}}_{\boldsymbol{B}} \underbrace{\begin{bmatrix} \boldsymbol{U}_1 \\ \vdots \\ \boldsymbol{U}_{N_K} \end{bmatrix}}_{\boldsymbol{U}} \tag{A.19}$$

where the blocks $\boldsymbol{X}_k$, $\boldsymbol{A}_k$, $\boldsymbol{U}_k$ and $\boldsymbol{B}_{k,l}$ is written:

$$\boldsymbol{X}_k = \begin{bmatrix} \nu_k^i \\ \dot{\nu}_k^i \\ \mu_k^i \end{bmatrix}, \ \boldsymbol{A}_k = \begin{bmatrix} 0 & 1 & 0 \\ -\omega_k^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ \boldsymbol{U}_k = \begin{bmatrix} s_i \langle u_n^i | \mathcal{K}_k \rangle \\ s_i \langle p^i | \mathcal{K}_k \rangle \\ s_j \langle u_n^j | \mathcal{K}_k \rangle \\ s_j \langle p^j | \mathcal{K}_k \rangle \end{bmatrix}$$

$$\boldsymbol{B}_{k,l} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\frac{\overline{c_0}}{\lambda_k L_t}\delta_{k,l} & 0 & -\frac{\overline{c_0}}{\lambda_k L_t}\langle \mathcal{K}_l^j | \mathcal{K}_k^i \rangle & 0 \\ 0 & \frac{1}{\overline{\rho_0}\lambda_k L_t}\delta_{k,l} & 0 & \frac{1}{\overline{\rho_0}\lambda_k L_t}\langle \mathcal{K}_l^j | \mathcal{K}_k^i \rangle \end{bmatrix} \tag{A.20}$$

And the output equation is written:

$$\underbrace{\begin{bmatrix} \boldsymbol{Y}_1 \\ \vdots \\ \boldsymbol{Y}_{N_K} \end{bmatrix}}_{\boldsymbol{Y}} = \underbrace{\begin{bmatrix} \boldsymbol{C}_{1,1} & \cdots & \boldsymbol{C}_{1,N_K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{C}_{N_K,1} & \cdots & \boldsymbol{C}_{N_K,N_K} \end{bmatrix}}_{\boldsymbol{C}} \underbrace{\begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_{N_K} \end{bmatrix}}_{\boldsymbol{X}} \tag{A.21}$$

where the blocks $\boldsymbol{Y}_k$ and $\boldsymbol{C}_{k,l}$ is written:

$$\boldsymbol{Y}_k = \begin{bmatrix} \mu_k^i \\ \nu_k^i \\ \mu_k^j \\ \nu_k^j \end{bmatrix}, \quad \boldsymbol{C}_{k,l} = \begin{bmatrix} 0 & 0 & \delta_{l,k} \\ \delta_{l,k} & 0 & 0 \\ 0 & 0 & \langle \mathcal{K}_l^i | \mathcal{K}_k^j \rangle \\ \langle \mathcal{K}_l^i | \mathcal{K}_k^j \rangle & 0 & 0 \end{bmatrix}$$

A.4 Spectral connections - multiperforated liner

When modeling a pressure jump caused by a multiperforated liner, the temporal evolution of the spectral amplitudes ν_k and μ_k is modified. The formulation provided here corresponds to the case where the jump coefficient $K_R(\omega)$ is replaced by the second order polynomial expansion given by:

$$K_R(\omega) = -K_R^A j\omega + K_R^B \omega^2 \text{ , with } K_R^A = -\frac{\pi a^2}{2U},\ K_R^B = \frac{2a^3}{3U^2} + \frac{\pi a^2 h}{4U^2} \tag{A.22}$$

If $K_R(\omega)$ is not of this form, for example if it is measured experimentally, the function $K_R(\omega)$ can be approximated using the Vector Fitting (VF) algorithm. This algorithm constructs a rational approximation of the function required for state-space implementation. Once fitted, the resulting model can be incorporated into the state-space framework in the same way as for flame models or impedance boundary conditions.

$$\left\{\begin{array}{rcl} \ddot{\nu}_k^i + \omega_k^2 \nu_k^i & = & -\dfrac{\overline{c_0}^2}{\lambda_k L_t} \left(s_i \langle u_n^i | \mathcal{K}_k^i \rangle + \sum_{l=1}^{N_K} \langle \mathcal{K}_l^j | \mathcal{K}_k^i \rangle s_j \langle u_n^j | \mathcal{K}_l^j \rangle \right) \\ \\ \dot{\mu}_k^i & = & \dfrac{1}{\lambda_k L_t \overline{\rho_0}} \left(s_i \langle p^i | \mathcal{K}_k^i \rangle + \sum_{l=1}^{N_K} \langle \mathcal{K}_l^j | \mathcal{K}_k^i \rangle s_j \langle p^j | \mathcal{K}_k^j \rangle - s_i \dfrac{\rho_u d^2}{K_R} \langle \dot{u}_n^i | \mathcal{K}_k^i \rangle \right) \\ \\ \nu_k^j & = & \sum_{l=1}^{N_K} \langle \mathcal{K}_l^i | \mathcal{K}_k^j \rangle \nu_k^i \\ \\ \mu_k^j & = & \sum_{l=1}^{N_K} \langle \mathcal{K}_l^i | \mathcal{K}_k^j \rangle \mu_k^i \end{array}\right. \tag{A.23}$$

The state-space realization is written:

$$\frac{d}{dt} \underbrace{\begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_{N_K} \end{bmatrix}}_{\boldsymbol{X}} = \underbrace{\begin{bmatrix} \boldsymbol{A}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \boldsymbol{A}_{N_K} \end{bmatrix}}_{\boldsymbol{A}} \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_{N_K} \end{bmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{B}_{1,1} & \cdots & \boldsymbol{B}_{1,N_K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{B}_{N_K} & \cdots & \boldsymbol{B}_{N_K,N_K} \end{bmatrix}}_{\boldsymbol{B}} \underbrace{\begin{bmatrix} \boldsymbol{U}_1 \\ \vdots \\ \boldsymbol{U}_{N_K} \end{bmatrix}}_{\boldsymbol{U}} \tag{A.24}$$

where the blocks $\boldsymbol{X}_k$, $\boldsymbol{A}_k$, $\boldsymbol{U}_k$ and $\boldsymbol{B}_{k,l}$ are written:

$$\boldsymbol{X}_k = \begin{bmatrix} \nu_k^i \\ \dot{\nu}_k^i \\ \mu_k^i \\ z_k \end{bmatrix}, \ \boldsymbol{A}_k = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega_k^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\dfrac{\rho_u d^2}{\lambda_k L_t \overline{\rho_0}} \\ 0 & 0 & 0 & -K_R^A / K_R^B \end{bmatrix}, \ \boldsymbol{U}_k = \begin{bmatrix} s_i \langle u_n^i | \mathcal{K}_k \rangle \\ s_i \langle p^i | \mathcal{K}_k \rangle \\ s_j \langle u_n^j | \mathcal{K}_k \rangle \\ s_j \langle p^j | \mathcal{K}_k \rangle \end{bmatrix}$$

$$\boldsymbol{B}_{k,l} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\dfrac{\overline{c_0}}{\lambda_k L_t}\delta_{k,l} & 0 & -\dfrac{\overline{c_0}}{\lambda_k L_t}\langle\mathcal{K}_l^j|\mathcal{K}_k^i\rangle & 0 \\ 0 & \dfrac{1}{\overline{\rho_0}\lambda_k L_t}\delta_{k,l} & 0 & \dfrac{1}{\overline{\rho_0}\lambda_k L_t}\langle\mathcal{K}_l^j|\mathcal{K}_k^i\rangle \\ -\dfrac{1}{K_R^B}\delta_{k,l} & 0 & 0 & 0 \end{bmatrix} \tag{A.25}$$

And the output equation is written:

$$\underbrace{\begin{bmatrix} \boldsymbol{Y}_1 \\ \vdots \\ \boldsymbol{Y}_{N_K} \end{bmatrix}}_{\boldsymbol{Y}} = \underbrace{\begin{bmatrix} \boldsymbol{C}_{1,1} & \cdots & \boldsymbol{C}_{1,N_K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{C}_{N_K,1} & \cdots & \boldsymbol{C}_{N_K,N_K} \end{bmatrix}}_{\boldsymbol{C}} \underbrace{\begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_{N_K} \end{bmatrix}}_{\boldsymbol{X}} \tag{A.26}$$

where the blocks $\boldsymbol{Y}_k$ and $\boldsymbol{C}_{k,l}$ are written:

$$\boldsymbol{Y}_k = \begin{bmatrix} \mu_k^i \\ \nu_k^i \\ \mu_k^j \\ \nu_k^j \end{bmatrix}, \quad \boldsymbol{C}_{k,l} = \begin{bmatrix} 0 & 0 & \delta_{l,k} & 0 \\ \delta_{l,k} & 0 & 0 & 0 \\ 0 & 0 & \langle\mathcal{K}_l^i|\mathcal{K}_k^j\rangle & 0 \\ \langle\mathcal{K}_l^i|\mathcal{K}_k^j\rangle & 0 & 0 & 0 \end{bmatrix}$$

A.5 Flame block: single entry FTF

Under the Vector Fitting formalism, the fluctuating heat release rate $Q'(t)$ is written:

$$Q'(t) = \frac{\overline{Q}}{\overline{u}} \left(R_0 + \sum_{k=1}^{n_r} \frac{R_k}{j\omega - p_k} + \sum_{k=n_r+1}^{n_r+n_c} \left[\frac{R_k}{j\omega - p_k} + \frac{R_k^*}{j\omega - p_k^*} \right] \right) u_n'(\boldsymbol{x_r}, t) \tag{A.27}$$

Even though it is possible to work with matrices containing complex numbers, the model is currently coded to work only with real matrices. In this way, Eq. A.27 is recast:

$$Q'(t) = \frac{\bar{Q}}{\bar{u}} \left(R_0 + \sum_{k=1}^{n_r} \frac{R_k}{j\omega - p_k} + \sum_{k=n_r+1}^{n_r+n_c} \frac{j\omega 2\Re(R_k) - 2\Re(p_k R_k^*)}{|p_k|^2 - 2j\omega\Re(p_k) - \omega^2} \right) u_n'(\boldsymbol{x_r}, t) \tag{A.28}$$

Then we introduce a new variable z_k that satisfies:

$$\forall k \leq n_r \begin{cases} z_k &= \dfrac{1}{j\omega - p_k} u_n' \\ \\ \dot{z}_k &= p_k z_k + u_n' \end{cases}$$

$$\forall k > n_r \begin{cases} z_k &= \dfrac{1}{-\omega^2 - j\omega 2\Re(p_k) + |p_k|^2} u_n' \\ \\ \ddot{z}_k &= 2\Re(p_k)\dot{z}_k - |p_k|^2 z_k + u_n' \end{cases} \tag{A.29}$$

Equation A.28 can be rewritten:

$$Q'(t) = \frac{\bar{Q}}{\bar{u}} \left(R_0 u_n'(\boldsymbol{x_r}, t) + \sum_{k=1}^{n_r} R_k z_k + \sum_{k=n_r+1}^{n_r+n_c} 2\Re(R_k)\dot{z}_k - 2\Re(p_k R_k^*) z_k \right) \tag{A.30}$$

which is recast as a state-space formulation as follows:

$$\begin{cases} \dot{\boldsymbol{X}}_{\boldsymbol{f}} &= \boldsymbol{A_f X_f} + \boldsymbol{B_f U_f} \\ \boldsymbol{Y_f} &= \boldsymbol{C_f X_f} \end{cases} \tag{A.31}$$

where

$$\boldsymbol{X_f} = \left[\begin{array}{c} \vdots \\ z_{k\leq n_r} \\ \vdots \\ \hline \vdots \\ z_{k>n_r} \\ \dot{z}_{k>n_r} \\ \vdots \\ \hline x(t) \end{array}\right], \boldsymbol{U_f} = \left[\begin{array}{c} u_n'(\boldsymbol{x_r}) \\ p'(\boldsymbol{x_r}) \end{array}\right], \boldsymbol{Y_f} = \left[\begin{array}{c} (\gamma-1)Q' \end{array}\right] \tag{A.32}$$

$$\boldsymbol{A_f} = \left[\begin{array}{ccc|ccc|c} \ddots & & & & & & \\ & p_{k\leq n_r} & & & \boldsymbol{0} & & \boldsymbol{0} \\ & & \ddots & & & & \\ \hline & & & \ddots & & & \\ & & & & 0 & 1 & \\ & \boldsymbol{0} & & & -|p_{k>n_r}|^2 & 2\Re(p_{k>n_r}) & \boldsymbol{0} \\ & & & & & \ddots & \\ \hline & \boldsymbol{0} & & & \boldsymbol{0} & & -\Omega_0 \end{array}\right] \tag{A.33}$$

$$\boldsymbol{B_f} = \left[\begin{array}{cc} \vdots & \vdots \\ 1 & 0 \\ \vdots & \vdots \\ \hline \vdots & \vdots \\ 0 & 0 \\ 1 & 0 \\ \vdots & \vdots \\ \hline \Omega_0 & 0 \end{array}\right], \boldsymbol{C_f^T} = \frac{(\gamma-1)\overline{Q}}{\overline{u}} \left[\begin{array}{c} \vdots \\ R_k \\ \vdots \\ \hline \vdots \\ -2\Re(p_k R_k^*) \\ 2\Re(R_k) \\ \vdots \\ \hline R_0 \end{array}\right] \tag{A.34}$$

Ω_0 is set to a high value to built a time-derivator in the last line, forcing the variable $x(t)$ to exponentially converge to $u'_n(\boldsymbol{x_r}, t)$ (detailed in Appendix A.1).

A.6 Frequency dependent impedance $Z(\omega)$

The dynamic equation of the averaged velocity in the control volume is written:

$$j\omega\overline{\hat{u}_n} = \frac{1}{\rho_0 L_t}\hat{p}(\boldsymbol{x_s}) - \frac{c_0 Z(\omega)}{L_t}\hat{u}_n(\boldsymbol{x_s}) \tag{A.35}$$

Under the state-space framework, the impedance $Z = Z(\omega)$ is approximated by the pole residue formulation, which is written:

$$Z(\omega) \simeq R_0 + \sum_{k=1}^{n_r} \frac{R_k}{j\omega - p_k} + \sum_{k=n_r+1}^{n_r+n_c} \frac{j\omega 2\Re(R_k) - 2\Re(p_k R_k^*)}{|p_k|^2 - 2j\omega\Re(p_k) - \omega^2} \tag{A.36}$$

As for flame transfer functions, we introduce the variable z_k that satisfies:

$$\begin{aligned}
&\forall k \leq n_r \left\{ \begin{array}{lcl} z_k &=& \dfrac{1}{j\omega - p_k} u'_n(\boldsymbol{x_s}) \\ \\ \dot{z}_k &=& p_k z_k + u'_n(\boldsymbol{x_s}) \end{array} \right. \\
&\forall k > n_r \left\{ \begin{array}{lcl} z_k &=& \dfrac{1}{-\omega^2 - j\omega 2\Re(p_k) + |p_k|^2} u'_n(\boldsymbol{x_s}) \\ \\ \ddot{z}_k &=& 2\Re(p_k)\dot{z}_k - |p_k|^2 z_k + u'_n(\boldsymbol{x_s}) \end{array} \right.
\end{aligned} \tag{A.37}$$

To build the temporal state-space associated with the impedance subdomain, the dynamic equation A.35 is expressed in the temporal domain and the z_k variables are introduced:

$$\begin{aligned}
\frac{\partial \overline{u_n}}{\partial t} &= \frac{1}{\rho_0 L_t} p(\boldsymbol{x_s}) \\
&- \frac{c_0}{L_t}\left(R_0 u_n'(\boldsymbol{x_s}) + \textstyle\sum_{k=1}^{n_r} R_k z_k\right) \\
&- \frac{c_0}{L_t} \textstyle\sum_{k=n_r+1}^{n_r+n_c} \left(2\Re(R_k)\dot{z}_k - 2\Re(p_k R_k^*) z_k\right)
\end{aligned} \tag{A.38}$$

which is recast as a state-space formulation as follows:

$$\left\{\begin{array}{rcl}
\dot{\boldsymbol{X}}_{\boldsymbol{i}} &=& \boldsymbol{A_i X_i} + \boldsymbol{B_i U_i} \\
\boldsymbol{Y_i} &=& \boldsymbol{C_i X_i}
\end{array}\right. \tag{A.39}$$

where

$$\boldsymbol{X_i} = \left[\begin{array}{c}
\vdots \\ z_{k\le n_r} \\ \vdots \\ \hline \vdots \\ z_{k>n_r} \\ \dot{z}_{k>n_r} \\ \vdots \\ \hline \overline{u_n}
\end{array}\right], \ \boldsymbol{U_i} = \left[\begin{array}{c} u_n'(\boldsymbol{x_s}) \\ p'(\boldsymbol{x_s}) \end{array}\right], \ \boldsymbol{Y_i} = \left[\ \overline{u_n'}\ \right] \tag{A.40}$$

$$\boldsymbol{A_i} = \left[\begin{array}{ccc|cccc|c}
\ddots & & & & & & & \\
& p_{k\le n_r} & & & \mathbf{0} & & & \mathbf{0} \\
& & \ddots & & & & & \\ \hline
& & & \ddots & & & & \\
& & & & 0 & 1 & & \\
& \mathbf{0} & & & -|p_{k>n_r}|^2 & 2\Re(p_{k>n_r}) & & \mathbf{0} \\
& & & & & & \ddots & \\ \hline
\cdots & -\frac{c_0}{L_t} R_{k\le n_r} & \cdots & \cdots & \frac{c_0}{L_t} 2\Re(p_k R_k^*) & -\frac{c_0}{L_t} 2\Re(R_k) & \cdots & 0
\end{array}\right] \tag{A.41}$$

$$\boldsymbol{B_i} = \left[\begin{array}{cc} \vdots & \vdots \\ 1 & 0 \\ \vdots & \vdots \\ \hline \vdots & \vdots \\ 0 & 0 \\ 1 & 0 \\ \vdots & \vdots \\ \hline -\dfrac{c_0 R_0}{L_t} & \dfrac{1}{\rho_0 L_t} \end{array}\right], \ \boldsymbol{C_i^T} = \left[\begin{array}{c} \vdots \\ 0 \\ \vdots \\ \hline \vdots \\ 0 \\ 0 \\ \vdots \\ \hline 1 \end{array}\right] \tag{A.42}$$

A.7 Acoustic Transfer Matrix (ATM)

Acoustic transfer matrices are used to model wave propagation in linear acoustic systems. They provide a compact way to relate acoustic quantities such as pressure and velocity (or alternatively, Riemann invariants) between two locations in a system, such as the inlet and outlet of a duct or across components like injectors, heat sources, or nozzles. By representing each element of the acoustic path with a matrix, complex systems can be constructed by chaining these matrices together, making this approach particularly suitable for network-based modeling and frequency-domain analysis. Under the network framework presented in this thesis, two subdomains Ω_1 and Ω_2 are connected through a connection subdomain C that defines the acoustic transfer. A schematic representation is given in Fig. A.1.

In their most common form, transfer matrices link pressure and velocity at two cross-sections of a component. These matrices are frequency-dependent and can capture effects such as reflections, transmissions, resonance, dissipation, and phase lag.

The form of the transfer matrices implemented in the STORM tool is the following:

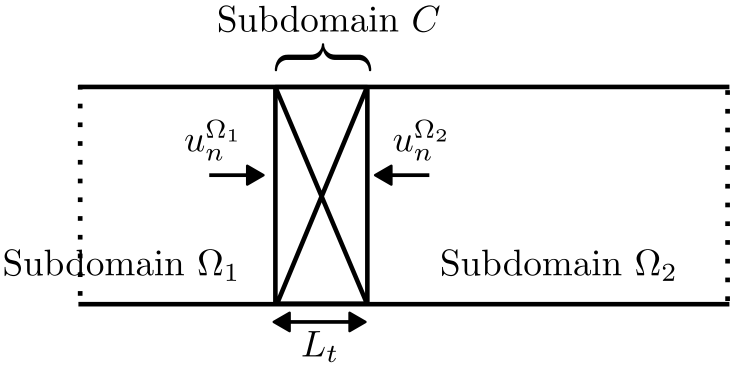


Figure A.1: Schematic representation of two geometrical subdomains Ω_1 and Ω_2 that are connected through a connection element C, which models the acoustic transfer.

$$\begin{bmatrix} p^{\Omega_2} \\ \rho_0 c_0 u_n^{\Omega_2} \end{bmatrix} = \begin{bmatrix} P(\omega) & R(\omega) \\ S(\omega) & T(\omega) \end{bmatrix} \begin{bmatrix} p^{\Omega_1} \\ \rho_0 c_0 u_n^{\Omega_1} \end{bmatrix} \tag{A.43}$$

The subscripts 1 and 2 correspond to upstream and downstream quantities, respectively (see Fig. A.1). ρ_0 and c_0 correspond to mean density and sound speed between the two connected domains. In the same way as for the section jump subdomain, a small volume is defined to enclose the component modeled by the transfer matrix. The mean pressure and velocity within this small volume of length L_t are then introduced and satisfy the volume-averaged Euler equations as follows:

$$\begin{cases} \dfrac{\partial \overline{u}}{\partial t} & = & \dfrac{1}{\rho_0} L_t \left(P(\omega) p^{\Omega_1} + R(\omega) \rho_0 c_0 u_n^{\Omega_1} - p^{\Omega_2} \right) \\ \dfrac{\partial \overline{p}}{\partial t} & = & \dfrac{\rho_0 c_0^2}{L_t} \left(\dfrac{1}{\rho_0 c_0} S(\omega) p^{\Omega_1} + T(\omega) u_n^{\Omega_1} + u_n^{\Omega_2} \right) \end{cases} \tag{A.44}$$

Note that in the case of a uniform fields, and by setting $P = 1$, $R = 0$, $S = 0$ and $T = A_1/A_2$, this state-space becomes exactly the same as that of the section jump between two tubes with cross-section areas A_1 and A_2. The coefficients of the transfer matrix are *a priori* depending on the pulsation.

They are therefore fitted using the Vector fitting algorithm, which is written on the generic form:

$$\begin{cases} P(\omega) = R_{0,p} + \sum_{k=1}^{N_p} \dfrac{R_{k,p}}{j\omega - p_{k,p}} \\ R(\omega) = R_{0,r} + \sum_{k=1}^{N_r} \dfrac{R_{k,r}}{j\omega - p_{k,r}} \\ S(\omega) = R_{0,s} + \sum_{k=1}^{N_s} \dfrac{R_{k,s}}{j\omega - p_{k,s}} \\ T(\omega) = R_{0,t} + \sum_{k=1}^{N_t} \dfrac{R_{k,t}}{j\omega - p_{k,t}} \end{cases} \tag{A.45}$$

The state-space associated with the ATM component must satisfy the convention for input/output vectors of connection elements (defined in Tab. 3.1).

$$\boldsymbol{A_{tm}} = \left[\begin{array}{cccc|ccc} \boldsymbol{A_P} & & & & & & \\ & \boldsymbol{A_R} & & & & \boldsymbol{0} & \\ & & \boldsymbol{A_S} & & & & \\ & & & \boldsymbol{A_T} & & & \\ \hline \dfrac{1}{\rho_0 L_t}\boldsymbol{C_P} & \dfrac{1}{\rho_0 L_t}\boldsymbol{C_R} & \boldsymbol{0} & \boldsymbol{0} & 0 & 0 & 0 \\ \boldsymbol{0} & \boldsymbol{0} & \dfrac{c_0}{L_t}\boldsymbol{C_S} & \dfrac{c_0}{L_t}\boldsymbol{C_T} & 0 & 0 & 0 \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & 0 & -\dfrac{1}{\rho_0} & 0 \end{array}\right], \boldsymbol{X_{tm}} = \left[\begin{array}{c} \boldsymbol{X_P} \\ \boldsymbol{X_R} \\ \boldsymbol{X_S} \\ \boldsymbol{X_T} \\ \hline \overline{u} \\ \overline{p} \\ \overline{\varphi} \end{array}\right] \tag{A.46}$$

$$\boldsymbol{B_{tm}} = \left[\begin{array}{cccc} \mathbf{0} & \boldsymbol{B_P} & \mathbf{0} & \mathbf{0} \\ \rho_0 c_0 \boldsymbol{B_R} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B_S} & \mathbf{0} & \mathbf{0} \\ \rho_0 c_0 \boldsymbol{B_T} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \frac{c_0}{L_t} R_{0,r} & \frac{1}{\rho_0 L_t} R_{0,p} & 0 & -\frac{1}{\rho_0 L_t} \\ \frac{\rho_0 c_0^2}{L_t} R_{0,t} & \frac{c_0}{L_t} R_{0,s} & \frac{\rho_0 c_0^2}{L_t} & 0 \\ 0 & 0 & 0 & 0 \end{array}\right], \boldsymbol{U_{tm}} = \begin{bmatrix} u_n^{\Omega_1} \\ p^{\Omega_1} \\ u_n^{\Omega_2} \\ p^{\Omega_2} \end{bmatrix} \tag{A.47}$$

$$\boldsymbol{Y_{tm}} = \begin{bmatrix} u_n(0) \\ \varphi(0) \\ u_n(L_t) \\ \varphi(L_t) \end{bmatrix}, \boldsymbol{C_{tm}} = \left[\begin{array}{cccc|ccc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & 0 & 0 & 1 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -1 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & 0 & 0 & 1 \end{array}\right] \tag{A.48}$$

Where individual matrices $\boldsymbol{A_{\{P,R,S,T\}}}$, $\boldsymbol{B_{\{P,R,S,T\}}}$, $\boldsymbol{C_{\{P,R,S,T\}}}$ and $\boldsymbol{X_{\{P,R,S,T\}}}$ are associated with the corresponding Vector Fitting formulations provided in Eq. A.45. The subscripts p, r, s, and t previously written on the poles p_k, the residues R_k, and the variables z_k are omitted here for clarity.

$$\boldsymbol{A_{\{P,R,S,T\}}} = \left[\begin{array}{c|c} \begin{matrix} \ddots & & \\ & p_{k\leq n_r} & \\ & & \ddots \end{matrix} & \mathbf{0} \\ \hline \mathbf{0} & \begin{matrix} \ddots & & \\ & \begin{matrix} 0 & 1 \\ -|p_{k>n_r}|^2 & 2\Re(p_{k>n_r}) \end{matrix} & \\ & & \ddots \end{matrix} \end{array}\right] \tag{A.49}$$

$$\boldsymbol{C}_{\{\boldsymbol{P},\boldsymbol{R},\boldsymbol{S},\boldsymbol{T}\}} = \left[\begin{array}{ccc|ccccc} \cdots & R_{k\leq n_r} & \cdots & \cdots & -2\Re(p_k R_k^*) & 2\Re(R_k) & \cdots \end{array} \right] \tag{A.50}$$

$$\boldsymbol{B}^{\boldsymbol{T}}_{\{\boldsymbol{P},\boldsymbol{R},\boldsymbol{S},\boldsymbol{T}\}} = \left[\begin{array}{ccc|cccc} \cdots & 1 & \cdots & \cdots & 0 & 1 & \cdots \end{array} \right] \tag{A.51}$$

$$\boldsymbol{X}^{\boldsymbol{T}}_{\{\boldsymbol{P},\boldsymbol{R},\boldsymbol{S},\boldsymbol{T}\}} = \left[\begin{array}{ccc|cccc} \cdots & z_{k\leq n_r} & \cdots & \cdots & z_{k>n_r} & \dot{z}_{k>n_r} & \cdots \end{array} \right] \tag{A.52}$$

A.8 Velocity jump as a source term: $[u] = \alpha u'(x_0)$

When considering only a source term of the form $[u] = \alpha u_n'(x_0)$, the temporal evolution of modal amplitudes Υ_n in the geometrical subdomain is written:

$$\left\{ \begin{array}{rcl} p'(\boldsymbol{x},t) & = & \sum_n^N \dot{\Upsilon}_n(t) U_n(x) \\ \\ \ddot{\Upsilon}_n(t) & = & -\sum_k^N [\boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{\Omega}^2 \boldsymbol{V} \boldsymbol{\Sigma}^{-1}]_{n,k} \Upsilon_k(t) \\ \\ & + & \langle H | U_n \rangle \rho_0 c_0^2 \alpha u_n'(x_0,t) \end{array} \right. \tag{A.53}$$

where $H(\boldsymbol{x})$ of the unit $[m^{-1}]$ is the Gaussian shape in the normal direction to the surface. The state matrix $\boldsymbol{A}$ of the geometrical subdomain is not modified (i.e Eq. A.9), only the input/output components change, such as:

$$\boldsymbol{X} = \begin{bmatrix} \vdots \\ \Upsilon_n \\ \dot{\Upsilon}_n \\ \vdots \end{bmatrix}, \boldsymbol{U} = \begin{bmatrix} \rho_0 c_0^2 \alpha u_n'(x_0,t) \end{bmatrix}, \boldsymbol{Y} = \begin{bmatrix} u_n'(x_0,t) \\ p'(x_0,t) \end{bmatrix} \tag{A.54}$$

$$\boldsymbol{B} = \begin{bmatrix} \vdots \\ 0 \\ \langle H|U_n\rangle \\ \vdots \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} \cdots & -\frac{1}{\rho_0}\nabla_n U_n(x_0) & 0 & \cdots \\ \cdots & 0 & U_n(x_0) & \cdots \end{bmatrix} \tag{A.55}$$

The state-space realization associated with the velocity jump of the form $[u] = \alpha u'_n(x_0)$ is written:

$$\begin{bmatrix} \dot{x}(t) \end{bmatrix} = \begin{bmatrix} -\Omega_0 \end{bmatrix} \begin{bmatrix} x(t) \end{bmatrix} + \begin{bmatrix} \Omega_0 & 0 \end{bmatrix} \begin{bmatrix} u'_n(x_0, t) \\ p'(x_0, t) \end{bmatrix} \tag{A.56}$$

$$\begin{bmatrix} y(t) \end{bmatrix} = \begin{bmatrix} \rho_0 c_0^2 \alpha \end{bmatrix} \begin{bmatrix} x(t) \end{bmatrix} \tag{A.57}$$

The value of Ω_0 is set high enough, typically $\Omega_0 = 2\pi \times 10^9$ so that $x(t)$ converges exponentially to $u'_n(x_0, t)$. This allows to return a source term y in the output equation that exponentially converges such that $y(t) \xrightarrow{t} \rho_0 c_0^2 \alpha u'_n(x_0, t)$.

A.9 Pressure jump as a source term: $[p] = \alpha p(x_0)$

Assume that at the position x_0, a pressure jump of the following form is applied:

$$[p] = \alpha p(x_0) \tag{A.58}$$

When considering a pressure jump, we need to encounter the velocity field correction.

Velocity field correction

As explained in section 4.5, the modal expansion of velocity fluctuations is corrected as follows:

$$\frac{\partial u_n'(x,t)}{\partial t} = -\frac{1}{\rho_0}\nabla_n p'(x,t) + \frac{1}{\rho_0}[p]H(x) \tag{A.59}$$

Integrating this equation and replacing the terms by their modal expansion leads to:

$$\begin{aligned} \boldsymbol{u'}(\boldsymbol{x},t) &= -\frac{1}{\rho_0}\textstyle\sum_{n=1}^{N} \Upsilon_n(t)\boldsymbol{\nabla U_n(x)} \\ &+ \frac{1}{\rho_0}\alpha\textstyle\sum_{n=1}^{N} \Upsilon_n(t)U_n(\boldsymbol{x_0})\widetilde{H}(\boldsymbol{x})\boldsymbol{n} \end{aligned} \tag{A.60}$$

where $\boldsymbol{n}$ is the unitary vector normal to the surface where the jump condition is applied.

State-space realization

When considering only a pressure jump of the form $[p] = \alpha p'(x_0)$, the temporal evolution of modal amplitudes $\Upsilon_n(t)$ in the geometrical subdomain is written:

$$\begin{cases} p'(\boldsymbol{x},t) &= \sum_n^N \dot{\Upsilon}_n(t)U_n(x) \\ \ddot{\Upsilon}_n(t) &= -\sum_k^N[\boldsymbol{\Sigma V^T\Omega^2 V\Sigma^{-1}}]_{n,k}\Upsilon_k(t) \\ &- \langle H'|U_n\rangle c_0^2\alpha\int^t p'(x_0,t')\mathrm{d}t' \end{cases} \tag{A.61}$$

Here H' is the derivative of the Gaussian that approximates the Dirac function. The integration of pressure fluctuation is obtained through the modal expansion by:

$$\left\{ \begin{array}{rcl} p'(x_0,t) & = & \sum_n^N \dot{\Upsilon}_n(t)U_n(x_0) \\ \\ \int^t p'(x_0,t')\mathrm{d}t' & = & \sum_n^N \Upsilon(t)U_n(x_0) \end{array} \right. \tag{A.62}$$

The state matrix $\boldsymbol{A}$ of the geometrical subdomain is not modified, only the input/output components change, such as:

$$\boldsymbol{X} = \begin{bmatrix} \vdots \\ \Upsilon_n \\ \dot{\Upsilon}_n \\ \vdots \end{bmatrix}, \boldsymbol{U} = \begin{bmatrix} -c_0\alpha \int^t p'(x_0,t')\mathrm{d}t' \end{bmatrix}, \boldsymbol{Y} = \begin{bmatrix} \int^t p'(x_0,t')\mathrm{d}t' \end{bmatrix} \tag{A.63}$$

$$\boldsymbol{B} = \begin{bmatrix} \vdots \\ 0 \\ \langle H'|U_n\rangle \\ \vdots \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} \cdots & U_n(x_0) & 0 & \cdots \end{bmatrix} \tag{A.64}$$

The state-space realization associated with the pressure jump of the form $[p] = \alpha p'(x_0)$ is written:

$$\begin{bmatrix} \dot{x}(t) \end{bmatrix} = \begin{bmatrix} -\Omega_0 \end{bmatrix} \begin{bmatrix} x(t) \end{bmatrix} + \begin{bmatrix} 0 & \Omega_0 \end{bmatrix} \begin{bmatrix} u'_n(x_0,t) \\ \int^t p'(x_0,t')\mathrm{d}t' \end{bmatrix} \tag{A.65}$$

$$\begin{bmatrix} y(t) \end{bmatrix} = \begin{bmatrix} -c_0^2\alpha \end{bmatrix} \begin{bmatrix} x(t) \end{bmatrix} \tag{A.66}$$

The value of Ω_0 is set high enough, typically $\Omega_0 = 2\pi \times 10^9$ so that $x(t)$ converges exponentially to $\int^t p'(x_0,t')\mathrm{d}t'$. This allows to return a source term y in the output equation that exponentially converges such that $y(t) \xrightarrow{t} -c_0^2\alpha \int^t p'(x_0, t')\mathrm{d}t'$.

A.10 Multiperforated liner as a source term

The Howe's model for multiperforated plates relates the pressure jump to the velocity perturbation upstream of the liner so that:

$$[p] = -\frac{1}{K_R}\rho_0 \frac{\partial u_n'(x_0, t)}{\partial t} \tag{A.67}$$

where K_R is the Rayleigh conductivity of the liner in $[m^{-1}]$. When considering a pressure jump, we need to encounter the velocity field correction.

Velocity field correction

As explained in section 4.5, the modal expansion of velocity fluctuations is corrected as follows:

$$\frac{\partial u_n'(x, t)}{\partial t} = -\frac{1}{\rho_0}\nabla_n p'(x, t) + \frac{1}{\rho_0}[p]H(x) \tag{A.68}$$

Including Eq. A.67 it comes:

$$\frac{\partial u_n'(x, t)}{\partial t} = -\frac{1}{\rho_0}\nabla_n p'(x, t) - \frac{1}{K_R}\frac{\partial u_n'(x_0, t)}{\partial t}H(x) \tag{A.69}$$

Evaluating this equation at $x = x_0$ allows to determine the value of the velocity field u_n' at x_0:

$$\frac{\partial u_n'(x_0, t)}{\partial t} = \frac{1}{1 + H(x_0)/K_R}\left(-\frac{1}{\rho_0}\nabla_n p'(x_0, t)\right) \tag{A.70}$$

By integrating this equation, and approximating all the terms by their modal expansion, one can express the modal expansion of velocity fluctuations, at the position x_0:

$$u'_n(x_0,t) = \frac{1}{1+\widetilde{H}(x_0)/K_R}\left(-\frac{1}{\rho_0}\sum_{n=1}^{N}\Upsilon_n(t)\nabla_n U_n(x_0)\right) \tag{A.71}$$

where $\widetilde{H}$ is the projection of the Gaussian shape H on the modal basis:

$$\widetilde{H}(x_0) = \sum_{n=1}^{N}\langle H|U_n\rangle U_n(x_0) \tag{A.72}$$

Integrating Eq. A.69 leads to the expression of the modal expansion of velocity fluctuations:

$$\left\{\begin{array}{rcl} \boldsymbol{u'}(\boldsymbol{x},t) & = & -\dfrac{1}{\rho_0}\sum_{n=1}^{N}\Upsilon_n(t)\boldsymbol{\nabla U_n(x)} \\ & + & \dfrac{\widetilde{H}(\boldsymbol{x})\boldsymbol{n}}{\rho_0\left(K_R+\widetilde{H}(x_0)\right)}\sum_{n=1}^{N}\Upsilon_n(t)\nabla_n U_n(\boldsymbol{x_0}) \end{array}\right. \tag{A.73}$$

where $\boldsymbol{n}$ is the unitary vector normal to the surface where the jump condition is applied.

State-space realization

The temporal evolution of modal amplitudes $\Upsilon_n(t)$ in the geometrical subdomain is written:

$$\left\{\begin{array}{rcl} p'(\boldsymbol{x},t) & = & \sum_n^N \dot{\Upsilon}_n(t)U_n(x) \\ \ddot{\Upsilon}_n(t) & = & -\sum_k^N[\boldsymbol{\Sigma V}^T\boldsymbol{\Omega^2 V\Sigma}^{-1}]_{n,k}\Upsilon_k(t) \\ & + & \langle H'|U_n\rangle c_0^2\rho_0\dfrac{1}{K_R}u'_n(x_0,t) \end{array}\right. \tag{A.74}$$

The state matrix $\boldsymbol{A}$ of the geometrical subdomain is not modified (see Eq. A.9), only the input/output components change, such as:

$$\boldsymbol{X} = \begin{bmatrix} \vdots \\ \Upsilon_n \\ \dot{\Upsilon}_n \\ \vdots \end{bmatrix}, \boldsymbol{U} = \begin{bmatrix} \dfrac{c_0^2\rho_0}{K_R} u_n'(x_0, t) \end{bmatrix}, \boldsymbol{Y} = \begin{bmatrix} u_n'(x_0, t) \\ p'(x_0, t) \end{bmatrix} \tag{A.75}$$

$$\boldsymbol{B} = \begin{bmatrix} \vdots \\ 0 \\ \langle H' | U_n \rangle \\ \vdots \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} \cdots & -\dfrac{\nabla_n U_n(x_0)}{\rho_0(1 + \widetilde{H}(x_0)/K_R)} & 0 & \cdots \\ \cdots & 0 & U_n(x_0) & \cdots \end{bmatrix} \tag{A.76}$$

The state-space realization of the liner-type pressure jump condition therefore is written:

$$\begin{bmatrix} x(t) \end{bmatrix} = \begin{bmatrix} -\Omega_0 \end{bmatrix} \begin{bmatrix} x(t) \end{bmatrix} + \begin{bmatrix} \Omega_0 & 0 \end{bmatrix} \begin{bmatrix} u_n'(x_0, t) \\ p'(x_0, t) \end{bmatrix} \tag{A.77}$$

$$\begin{bmatrix} y(t) \end{bmatrix} = \begin{bmatrix} \dfrac{c_0^2\rho_0}{K_R} \end{bmatrix} \begin{bmatrix} x(t) \end{bmatrix} \tag{A.78}$$

The value of Ω_0 is set high enough, typically $\Omega_0 = 2\pi \times 10^9$ so that $x(t)$ converges exponentially to $u_n'(x_0, t)$. This allows to return a source term y in the output equation that exponentially converges such that $y(t) \xrightarrow{t} c_0^2\rho_0 u_n'(x_0, t)/K_R$.

Appendix B

Pseudo-analytical solutions

B.1 Jump conditions in a duct

The configuration in Fig. B.1 considered is a 1-dimensional duct where a jump condition either on the pressure or on the velocity fluctuation is imposed at the position $x = x_0$.

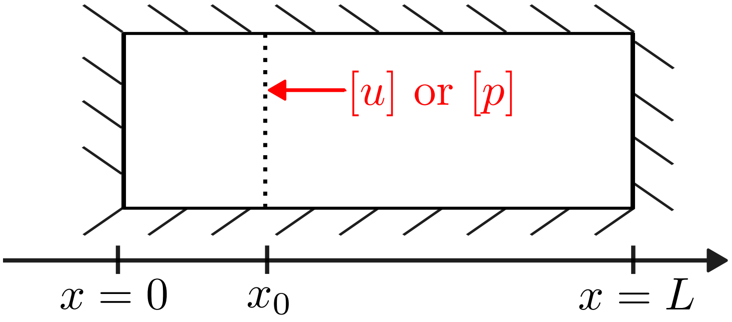


Figure B.1: Scheme of the academic configuration of a single 1-dimensional duct of length $L = 1\ m$, where either a pressure jump $[p]$ or velocity jump $[u]$ is applied at the position $x_0 = 0.32\ m$.

B.1.1 Velocity jump in a duct: $[u] = \alpha u(x_0)$

The pseudo-analytical solutions are given by applying the conservation of the pressure and the acoustic flux at the position $x = x_0$. To derive the pseudo-analytical solutions, the pressure and velocity fluctuation are decomposed in each side of the jump position x_0 as a couple of forward/backward waves:

$$\left\{\begin{array}{rcl} p_i & = & A_i^+ e^{jk_i(x_i - x_{i0})} + A_i^- e^{-jk_i(x_i - x_{i0})} \\ \rho_i c_i u_i & = & A_i^+ e^{jk_i(x_i - x_{i0})} - A_i^- e^{-jk_i(x_i - x_{i0})} \end{array}\right. \tag{B.1}$$

where ρ_i and c_i are the density and sound speed. The subscript i denotes either the left part of the duct $i = 1$ (i.e $x < x_0$) or the right side $i = 2$ (i.e $x > x_0$). By construction, the values of the x-axis origin shift are $x_{10} = 0$ and $x_{20} = x_0$. The jump relations are then written:

$$\left\{\begin{array}{rcl} u_2(x_0) - (1+\alpha)u_1(x_0) & = & 0 \\ p_2(x_0) - p_1(x_0) & = & 0 \end{array}\right. \tag{B.2}$$

Boundary conditions must also be imposed to close the system. We chose rigid-wall BCs on both sides that are written:

$$\left\{\begin{array}{rcl} u_1(0) & = & 0 \\ u_2(L) & = & 0 \end{array}\right. \tag{B.3}$$

Combining Eqs. B.1 B.2 and B.3 lead to the following relations for the coefficients A_i^+ and A_i^-:

$$\left\{\begin{array}{rcl} A_1^+ - A_1^- & = & 0 \\ A_2^+ e^{jk_2(L-x_0)} - A_2^- e^{-jk_2(L-x_0)} & = & 0 \\ \dfrac{1}{\rho_2 c_2}(A_2^+ - A_2^-) - \dfrac{1+\alpha}{\rho_1 c_1}(A_1^+ e^{jk_1 x_0} - A_1^- e^{-jk_1 x_0}) & = & 0 \\ A_1^+ e^{jk_1 x_0} + A_1^- e^{-jk_1 x_0} - A_2^+ - A_2^- & = & 0 \end{array}\right. \tag{B.4}$$

This system is rewritten in the form of $\boldsymbol{A}(\omega)\boldsymbol{X} = 0$ where the $\boldsymbol{A}$ matrix and the $\boldsymbol{X}$ vector are given by:

$$\boldsymbol{A} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & e^{jk_2(L-x_0)} & -e^{-jk_2(L-x_0)} \\ -\dfrac{1+\alpha}{\rho_1 c_1} e^{jk_1 x_0} & \dfrac{1+\alpha}{\rho_1 c_1} e^{-jk_1 x_0} & \dfrac{1}{\rho_2 c_2} & -\dfrac{1}{\rho_2 c_2} \\ e^{jk_1 x_0} & e^{-jk_1 x_0} & -1 & -1 \end{bmatrix} \tag{B.5}$$

$$\boldsymbol{X^T} = \begin{bmatrix} A_1^+ & A_1^- & A_2^+ & A_2^- \end{bmatrix} \tag{B.6}$$

The dependency of matrix $\boldsymbol{A}$ on ω is not explicit here, but the pulsation is related to the wavenumbers such as: $\omega = k_i c_i$. The pseudo-analytical method consists in deriving analytically this system and then solving $\det \boldsymbol{A} = 0$. This is performed with the Python library Scipy.

B.1.2 Pressure jump in a duct: $[p] = \alpha p(x_0)$

Applying only pressure jump condition of the form $[p] = \alpha p(x_0)$ at the position x_0 leads to the dispersion relation $\det \boldsymbol{A} = 0$, where $\boldsymbol{A}$ is defined as:

$$\boldsymbol{A} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & e^{jk_2(L-x_0)} & -e^{-jk_2(L-x_0)} \\ -\dfrac{1}{\rho_1 c_1} e^{jk_1 x_0} & \dfrac{1}{\rho_1 c_1} e^{-jk_1 x_0} & \dfrac{1}{\rho_2 c_2} & -\dfrac{1}{\rho_2 c_2} \\ (1+\alpha) e^{jk_1 x_0} & (1+\alpha) e^{-jk_1 x_0} & -1 & -1 \end{bmatrix} \tag{B.7}$$

B.1.3 MLPF jump type in a duct

The Howe's model for multiperforated plates relates the pressure jump to the velocity perturbation upstream of the liner so that:

$$[p] = -\frac{1}{K_R}\rho_0 \frac{\partial u_n'(x_0,t)}{\partial t} \tag{B.8}$$

Pseudo-analytical solution are given by solving the dispersion relation $\det \boldsymbol{A} = 0$ where $\boldsymbol{A}$ is written:

$$\boldsymbol{A} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & e^{jk_2(L-x_0)} & -e^{-jk_2(L-x_0)} \\ -\frac{1}{\rho_1 c_1} e^{jk_1 x_0} & \frac{1}{\rho_1 c_1} e^{-jk_1 x_0} & \frac{1}{\rho_2 c_2} & -\frac{1}{\rho_2 c_2} \\ \left(-\frac{j\omega}{K_R c_2} - 1\right) e^{jk_1 x_0} & \left(\frac{j\omega}{K_R c_2} - 1\right) e^{-jk_1 x_0} & 1 & 1 \end{bmatrix} \tag{B.9}$$

B.2 Double entry FTF

The double input / single output FTF model that has been implemented in the STORM framework (Eq. B.10) takes the following form.

$$Q' = \overline{Q}\left(N_{air} e^{j\omega\tau_{air}} \frac{u'(x_{air})}{\overline{u}_{air}} + N_{H_2} e^{j\omega\tau_{H_2}} \frac{u'(x_{H_2})}{\overline{u}_{H_2}} \right) \tag{B.10}$$

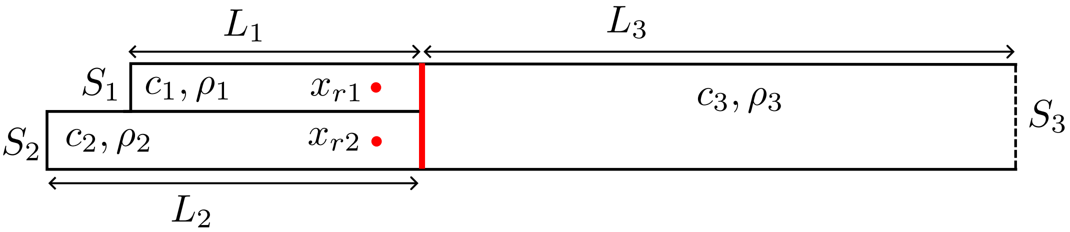


Figure B.2: Schematic representation of the 1.5D test case used to validate the numerical implementation of the double entry FTF.

In each duct i the spatial evolution of pressure/velocity fluctuations is expressed as a couple of forward/backward waves as follows. The origin

$x_i = 0$ is defined in each duct, as the left side of the duct. The indices $i = 1$, $i = 2$ and $i = 3$ can be respectively associated with the H2 inlet, the Air inlet and the combustion chamber. The single flame is positioned at the outlet of the two inlet lines, and it is represented by the red line in Fig. B.2. x_{r1} and x_{r2} are the two reference points for velocity perturbations in the double entry FTF. L_i and S_i are respectively the length and surface of each ducts. c_i and ρ_i are the sound speed and density. The value of all these parameters are summarized in Tab. 5.3. The boundary conditions are set as walls at the inlet of the two fuel lines, and open atmosphere and the outlet of the combustion chamber.

In each duct i, the pressure and velocity fluctuation are decomposed as follows:

$$\left\{\begin{array}{lll} p_i'(x_i,t) & = & \left(A_i^+ e^{jk_i x_i} + A_i^- e^{-ik_i x_i}\right) e^{-j\omega t} \\ \\ u_i'(x_i,t) & = & \dfrac{1}{\rho_i c_i}\left(A_i^+ e^{jk_i x_i} - A_i^- e^{-jk_i x_i}\right) e^{-j\omega t} \end{array}\right. \tag{B.11}$$

Note that the sign convention used to derive the analytical solution ($e^{-j\omega t}$) is the opposite from the one used in STORM, but this has no impact on the final results. The 6 unknowns A_i^+ and A_i^- are determined by applying the boundary/jump conditions. The 3 boundary conditions are the followings:

$$\left\{\begin{array}{lll} u_1'(x_1=0) & = & 0 \Rightarrow A_1^+ - A_1^- = 0 \\ \\ u_2'(x_2=0) & = & 0 \Rightarrow A_2^+ - A_2^- = 0 \\ \\ p_3'(x_3=L_3) & = & 0 \Rightarrow A_3^+ e^{jk_3 L_3} + A_3^- e^{-jk_3 L_3} = 0 \end{array}\right. \tag{B.12}$$

The continuity of pressure leads to the following relations:

$$\left\{\begin{array}{lcl} p_1'(x_1 = L_1) & = & p_3'(x_3 = 0) \;\Rightarrow A_1^+ e^{jk_1L_1} + A_1^- e^{-jk_1L_1} = A_3^+ + A_3^- \\ \\ p_2'(x_2 = L_2) & = & p_3'(x_3 = 0) \;\Rightarrow A_2^+ e^{jk_2L_2} + A_2^- e^{-jk_2L_2} = A_3^+ + A_3^- \end{array}\right. \tag{B.13}$$

Under the forcing induced by the unsteady heat release rate Q' of the flame, the flux conservation is expressed as follows:

$$S_1 u_1'(x_1 = L_1) + S_2 u_2'(x_2 = L_2) + \frac{\gamma - 1}{\gamma p_0} Q' = S_3 u_3'(x_3 = 0) \tag{B.14}$$

Combining Eqs. B.12 B.13 and B.14 leads to the following dispersion relation:

$$\begin{aligned} & A_1^+ \underbrace{\frac{1}{\rho_1 c_1}\left(S_1 e^{jk_1L_1} + \frac{\gamma-1}{\gamma p_0}\frac{\overline{Q}}{\overline{u}_1} N_1 e^{j\omega\tau_1} e^{jk_1x_{r1}}\right)}_{\Gamma_1^+} \\ + \quad & A_1^- \underbrace{\frac{1}{\rho_1 c_1}\left(-S_1 e^{-jk_1L_1} - \frac{\gamma-1}{\gamma p_0}\frac{\overline{Q}}{\overline{u}_1} N_1 e^{j\omega\tau_1} e^{-jk_1x_{r1}}\right)}_{\Gamma_1^-} \\ + \quad & A_2^+ \underbrace{\frac{1}{\rho_2 c_2}\left(S_2 e^{jk_2L_2} + \frac{\gamma-1}{\gamma p_0}\frac{\overline{Q}}{\overline{u}_2} N_2 e^{j\omega\tau_2} e^{jk_2x_{r2}}\right)}_{\Gamma_2^+} \\ + \quad & A_2^- \underbrace{\frac{1}{\rho_2 c_2}\left(-S_2 e^{-jk_2L_2} - \frac{\gamma-1}{\gamma p_0}\frac{\overline{Q}}{\overline{u}_2} N_2 e^{j\omega\tau_2} e^{-jk_2x_{r2}}\right)}_{\Gamma_2^-} \\ - \quad & \frac{S_3}{\rho_3 c_3}\left(A_3^+ - A_3^-\right) = 0 \end{aligned} \tag{B.15}$$

The unknowns modal amplitudes are then found by solving the linear system $\boldsymbol{AX} = \mathbf{0}$ where

$$\boldsymbol{X} = \begin{bmatrix} A_1^+ \\ A_1^- \\ A_2^+ \\ A_2^- \\ A_3^+ \\ A_3^- \end{bmatrix} \text{ and } \boldsymbol{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{jk_3L_3} & e^{-jk_3L_3} \\ e^{jk_1L_1} & e^{-jk_1L_1} & 0 & 0 & -1 & -1 \\ 0 & 0 & e^{jk_2L_2} & e^{-jk_2L_2} & -1 & -1 \\ \Gamma_1^+ & \Gamma_1^- & \Gamma_2^+ & \Gamma_2^- & -\dfrac{S_3}{\rho_3 c_3} & \dfrac{S_3}{\rho_3 c_3} \end{bmatrix} \tag{B.16}$$

This linear system has non-trivial solutions only if $\det(A) = 0$, leading to the dispersion relation. To solve this equation, we use the Python function minimize from Scipy library, by minimizing the quantity $1/\kappa(A)$, where $\kappa(A)$ is the condition number of A, defined as the ratio between the largest and the smallest singular values.

Appendix C

Singular Value decomposition

C.1 Metric adaptation

The initial frame $\boldsymbol{\phi}(\boldsymbol{x})$ and the orthonormalized basis $\boldsymbol{U}(\boldsymbol{x})$ are row-vectors defined as follows: $\boldsymbol{\phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}) \cdots \phi_N(\boldsymbol{x})]$ and $\boldsymbol{U}(\boldsymbol{x}) = [U_1(\boldsymbol{x}) \cdots U_N(\boldsymbol{x})]$. In this notation, each component $\phi_i(\boldsymbol{x})$ or $U_i(\boldsymbol{x})$ is an analytical function, and we note by $\underline{\boldsymbol{\phi_i}}$ or $\underline{\boldsymbol{U_i}}$ the numerical array containing the approximation of the function at each vertex of the mesh.

The SVD orthonormalization process must be done with respect to the inner product: $\langle\phi_n|\phi_m\rangle = \int_V \phi_n(\boldsymbol{x})\phi_m(\boldsymbol{x})\mathbf{d}\boldsymbol{x}$, involving the volume integral of the component-wise product of the two vectors ϕ_n and ϕ_m. From a numerical point of view, this inner product must be weighted by the volume of each vertices of the mesh as follows:

$$\langle\phi_n|\phi_m\rangle = \underline{\boldsymbol{\phi_n}}^T \, \underline{\underline{\boldsymbol{M}}} \, \underline{\boldsymbol{\phi_m}} \tag{C.1}$$

where $\boldsymbol{M}$ is the mass matrix associated with the mesh. This quantity is inherent to each mesh, and is calculated as the inner product between each basis function spaces of the mesh. If the SVD is directly done on $\boldsymbol{\phi}(\boldsymbol{x})$ without any modification, the algorithm will orthonormalize the frame with respect to the numerical inner product $\underline{\boldsymbol{\phi_n}}^T \, \underline{\boldsymbol{\phi_m}}$ without considering the volume of each vertex $\mathbf{d}\boldsymbol{x_i}$. Thus, a metric adaptation is applied on $\boldsymbol{\phi}$ before computing the SVD. The metric-adapted frame $\widetilde{\boldsymbol{\phi}}$ is:

$$\widetilde{\boldsymbol{\phi}} = \underline{\underline{\boldsymbol{C}}}\boldsymbol{\phi} \tag{C.2}$$

where $\underline{\underline{\boldsymbol{C}}}$ is the lower triangulate matrix from the Cholesky decomposition of the mass matrix such as: $\underline{\underline{\boldsymbol{M}}} = \underline{\underline{\boldsymbol{C}}}^T\underline{\underline{\boldsymbol{C}}}$. The SVD is then done on this metric-corrected frame $\tilde{\boldsymbol{\phi}}$ to orthonormalize the frame with respect to the correct numerical inner product defined in Eq. C.1. It is written:

$$\widetilde{\boldsymbol{\phi}} = \widetilde{\boldsymbol{U}}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{C.3}$$

Finally, the $\boldsymbol{U}$ basis required is obtained with $\boldsymbol{U} = \underline{\underline{\boldsymbol{C}}}^{-1}\widetilde{\boldsymbol{U}}$.

C.2 Short note on the threshold selection for the eigensolver and truncated SVD

Let $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ be a symmetric matrix.

C.2.1 Residual versus forward error on eigenvectors

The purpose of this section is to describe a bound on the norm of the difference between an exact unitary eigenvector $\boldsymbol{\phi}$ of $\boldsymbol{A}$ and the one computed numerically $\hat{\boldsymbol{\phi}}$. This bound includes the 2-norm of the residual $\boldsymbol{r} = (\boldsymbol{A} - \hat{\lambda}\boldsymbol{I})\hat{\boldsymbol{\phi}}$ associated with the corresponding eigenpair $(\hat{\lambda} = \hat{\boldsymbol{\phi}}^T\boldsymbol{A}\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\phi}})$ and the distance with the closest eigenvalue of $\hat{\lambda}$.

Theorem 1. *(See e.g., [187, Theorem 3.9, p. 63]).*
Let $\hat{\boldsymbol{\phi}}$ be an approximate eigenvector of unit norm of $\boldsymbol{A}$, $\hat{\lambda} = \hat{\boldsymbol{\phi}}^T\boldsymbol{A}\hat{\boldsymbol{\phi}}$ its associated approximate eigenvalue and $\boldsymbol{r} = (\boldsymbol{A} - \hat{\lambda}\boldsymbol{I})\hat{\boldsymbol{\phi}}$. Let λ be the eigenvalue of $\boldsymbol{A}$ closest to $\hat{\lambda}$ and δ the distance from $\hat{\lambda}$ to the rest of the spectrum, i.e.,

$\delta = \min_i\{|\lambda_i - \hat{\lambda}|, \lambda_i \neq \lambda\}$. *The eigenvalue for which the minimum distance* δ *is reached is noted* λ_j*, i.e* $\delta = |\tilde{\lambda} - \lambda_j|$. *If* $\boldsymbol{\phi}$ *is the eigenvector of* $\boldsymbol{A}$ *associated with* λ *we have*

$$\sin\theta(\hat{\boldsymbol{\phi}}, \boldsymbol{\phi}) \leq \frac{\|\boldsymbol{r}\|_2}{\delta}. \tag{C.4}$$

Let us write $\hat{\boldsymbol{\phi}} = \cos\theta\boldsymbol{\phi} + \sin\theta\boldsymbol{z}$ where $\boldsymbol{z}$ is a unit vector orthogonal to $\boldsymbol{\phi}$. If $\theta \ll 1$, so that $\cos\theta \approx 1$ and $\sin\theta \approx \theta$ then $\hat{\boldsymbol{\phi}} - \boldsymbol{\phi} \approx \theta\boldsymbol{z}$ so that

$$\|\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}\|_2 \leq \frac{\|\boldsymbol{r}\|_2}{\delta}.$$

Notice that the assumption $\theta \ll 1$ can be discarded to get an exact expression of

$$\|\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}\|_2 = \tan\theta(1 + \sin^2\theta)^{\frac{1}{2}} \tag{C.5}$$

that is not much useful as we shall try to compute the eigenpairs accurately enough using a stopping criterion based on the residual (possibly scaled).

In practice the closest eigenvalue to $\hat{\lambda}$ is not known so that δ cannot be computed, but a lower bound can be derived that only involves the computed eigenvalues and the residual norm [187, p. 64]:

$$\delta = |\hat{\lambda} - \lambda_j| \geq |\hat{\lambda} - \hat{\lambda}_j| - \|\boldsymbol{r_j}\|_2.$$

This result indicates that the computed eigenvectors are at a distance that depends on the associated residual norm and how close are the computed eigenvalues of $\boldsymbol{A}$.

For a set of computed eigenvectors (frame) $\hat{\Phi} = [\hat{\boldsymbol{\phi}}_{\boldsymbol{1}}, ..., \hat{\boldsymbol{\phi}}_{\boldsymbol{N}}]$ we will give in the next section the norm of the difference between $\hat{\boldsymbol{\phi}}_{\boldsymbol{i}}$ and its approximation extracted from its truncated SVD.

C.2.2 Error associated with a computed eigenvector $\hat{\phi}$ versus truncated SVD of $\hat{\Phi}$

Let $\hat{\Phi} = \boldsymbol{U\Sigma V}^T$ be the SVD of $\hat{\Phi}$, where $\boldsymbol{U} = [\boldsymbol{u_1}, ..\boldsymbol{u_r}] \in \mathbb{R}^{n\times r}$ are the left singular vectors that form an orthonormal basis of $range(\hat{\Phi})$, $\boldsymbol{\Sigma}$ is a diagonal matrix where the diagonal entries are the non-zero singular values σ_i is a decreasing order and $\boldsymbol{V} = [\boldsymbol{v_1}, ..., \boldsymbol{v_r}] \in \mathbb{R}^{m\times r}$ the right singular vectors.

For $k \leq m$, we define the best rank k approximation of $\hat{\Phi}$ as $\hat{\Phi}_k = \boldsymbol{U_k \Sigma_k V_k^T}$ with $\boldsymbol{U_k} = [\boldsymbol{u_1}, ..., \boldsymbol{u_k}]$, ...

Theorem 2. *Eckart-Young Theorem (See e.g., [98, Theorem 2.4.8, p. 79]).*

$$\min_{rank(\Psi)=k} \|\hat{\Phi} - \Psi\|_2 = \|\hat{\Phi} - \hat{\Phi}_k\|_2 = \sigma_{k+1}$$

Coming back to the definition of the 2-norm of a matrix that is

$$\|\hat{\Phi}\|_2 = \max_{\|\boldsymbol{x}\|_2=1} \|\hat{\Phi}\boldsymbol{x}\|_2$$

and considering the canonical basis vectors we have

$$\|(\hat{\Phi} - \hat{\Phi}_k)e_i\|_2 = \|\boldsymbol{\hat{\phi}_i} - \hat{\Phi}_k e_i\|_2 \leq \sigma_{k+1}.$$

This shows that the best approximation of any vector $\hat{\phi}$ in the space spanned by U_k is at most at a distance (in 2-norm) of σ_{k+1}, that is

$$\min_{\tilde{\boldsymbol{\phi}} \in span(\boldsymbol{U_k})} \|\boldsymbol{\hat{\phi}_i} - \tilde{\boldsymbol{\phi}}\| \leq \sigma_{k+1} \tag{C.6}$$

The bounds (C.5) and (C.6) indicate that the stopping criterion threshold for the eigensolvers used to compute the $\hat{\phi}$ and the threshold used to define the truncated SVD should be chosen consistently. In practice, the truncation threshold is chosen to be ten times higher than the solver tolerance.

Appendix D

Code development and package management

The Python module developed in the framework of this thesis, named STORM, is organized following the common conventions for Python packaging and code readability (as recommended by PEP8 and community guidelines). The goal of this structure is to clearly separate the core functionalities (solvers and numerical routines), input/output utilities, and testing framework, while ensuring the code remains modular and easy to extend.

D.1 Code organization

The global directory tree of the module is presented below, highlighting the organization of its main components.

```
.
storm/
├── README.md
├── CHANGELOG.md
├── CONTRIBUTING.md
├── LICENSE
├── setup.py
├── setup.cfg
├── Dockerfile
├── docs/
├── tests/
│   ├── unit/
│   └── func/
└── src/storm/
    ├── smio.py
    ├── preproc/
    │   ├── VF/
    │   ├── create_shapes.py
    │   ├── fenics2storm_mod.py
    │   ├── fenics2storm_connection.py
    │   ├── fits2storm.py
    │   ├── flame2storm.py
    │   └── helmholtz_solvers.py
    ├── postproc/
    │   ├── plot_modes.py
    │   ├── results.py
    │   └── spurious_id.py
    └── state_spaces/
        ├── state_space.py
        ├── assembler.py
        ├── geometrical.py
        ├── connection.py
        └── flame.py
```

D.1.1 General workflow

In addition to the internal structure of the STORM module, it is important to clarify how a user should organize a project folder in order to run a simulation efficiently. While STORM is designed as a standalone Python package, each study (or simulation case) requires a specific directory layout to properly handle input data (such as geometry, boundary conditions, and transfer functions), configuration files, and the results generated by the solver.

The recommended folder organization for a typical simulation case is illustrated in Fig. D.1. This structure, along with the role of each file and subdirectory, allows users to quickly set up, run, and post-process their computations using STORM.

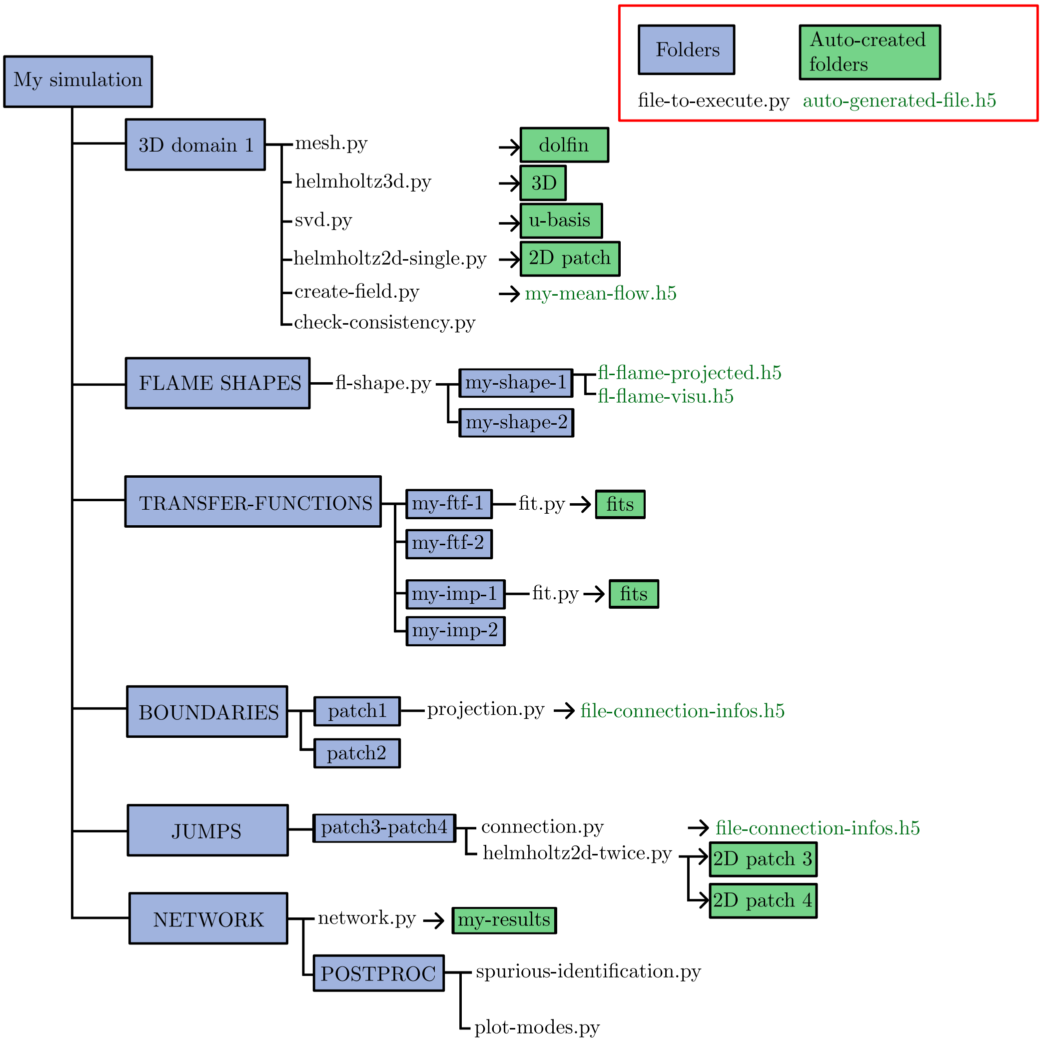


Figure D.1: Blue boxes represent the main directories that the user must create depending on the studied case (e.g., geometry, boundary conditions, and transfer functions). Green boxes correspond to directories automatically generated by STORM during the simulation, which are later used for post-processing or visualization. The .py files shown in the structure are Python scripts that the user must execute to set up, run, or analyze the simulation. Templates for these scripts are provided in the STORM documentation to guide new users.

The folder structure presented in Figure D.1 provides the physical organization of a STORM simulation case. Complementary to this, Figure D.2 illustrates the logical workflow of a typical simulation, detailing the order in which the different Python scripts are executed and how the results produced at each step are stored within the folder hierarchy. This diagram guides the user through the successive stages of a simulation — from mesh and data preparation, through pre-processing and core computations, to post-processing — while indicating decision points based on data availability (e.g., AVBP fields, experimental transfer functions, or non-trivial boundary conditions).

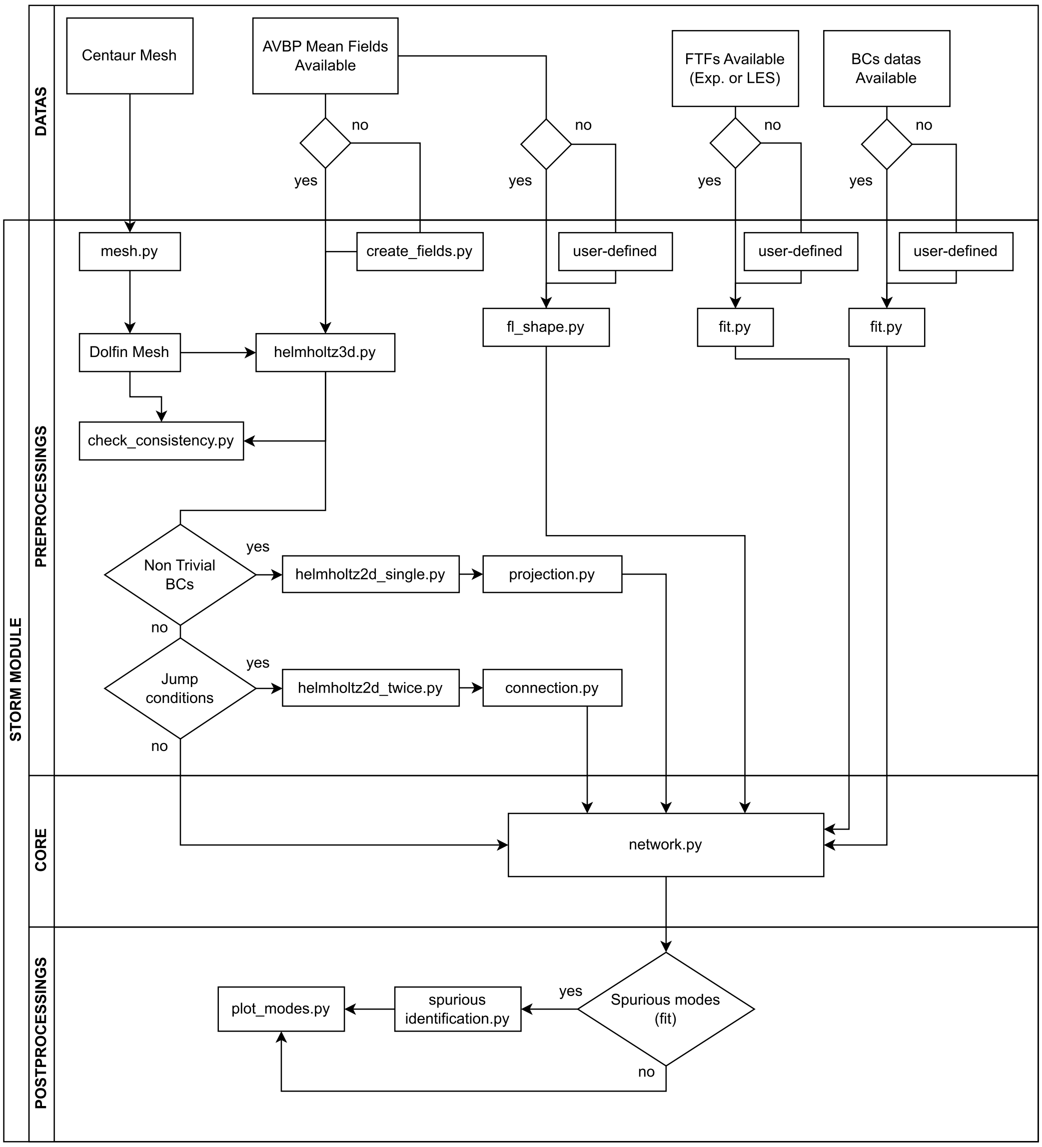


Figure D.2: Logical workflow of a STORM simulation, showing the sequence of Python scripts to execute and decision points based on data availability. Outputs from each step are organized into the folders introduced in Figure D.1.

D.1.2 Tutorials and documentation website

Building upon the folder organization and the simulation workflow, the STORM documentation provides a series of tutorials that illustrate typical use cases of the solver. These tutorials not only guide new users through practical examples but also strictly adhere to the directory structure (Fig. D.1) and workflow (Fig. D.2) previously described, ensuring consistency between user practices and the internal logic of the code.

From a development perspective, the tutorials serve as part of STORM's continuous integration system: they are executed automatically every time new code is pushed to the repository, ensuring that core functionalities remain operational. According to the coverage metric reported by pytest, these functional tests cover approximately 70% of the STORM codebase. The test suite includes a wide range of cases, such as a simple tube with a flame, a multiperforated plate on an annular geometry, the MICCA combustor with experimentally measured transfer matrices, etc. This variety of test cases ensures that the solver remains robust and consistent across different physical and geometrical configurations.

The documentation also details how to install STORM. For Linux users, a Docker-based installation is provided. For high-performance computing (HPC) environments, a Singularity container is recommended, ensuring portability across clusters. Both the Dockerfile and Singularity files are distributed within the STORM source code, allowing users to build and customize their own containers if needed.

D.2 FEniCS scripts

This section provides examples of FEniCS scripts, illustrating how the finite element library is used within STORM. These examples aim to serve as practical references for developers who wish to customize or extend the solver by leveraging FEniCS functionalities for mesh handling, field manipulation, or numerical solvers.

Throughout the work conducted in this thesis, the FEniCS Project Discourse forum is an invaluable resource. This community-driven platform contains a wealth of examples, ranging from basic syntax demonstrations to advanced applications, which significantly helped the development of the STORM's source code.

D.2.1 Extrapolating values

The following lines of code allow to compute the value of the inner product between mode 6 of the mesh $\mathcal{M}_1^b$ and mode 7 of the mesh $\mathcal{M}_2^b$ i.e $(M_m)_{6,7} = \langle \mathcal{K}_6^{(1)} | \mathcal{K}_7^{(2)} \rangle$.

Listing D.1: Example to compute inner products between modes that are not defined on the same mesh (extrapolation) using FEniCS

```
from fenics import *
from ufl import dx, inner
import numpy as np

# Assume that mesh1 and mesh 2 are already defined
# Define the function space of elements
# of order 1 on the mesh 1
fs1 = FunctionSpace(mesh1, "Lagrange", 1)

# Define the function space of elements
# of order 1 on the mesh 2
fs2 = FunctionSpace(mesh2, "Lagrange", 1)
```

```
# Define the useful functions
mode_1 = Function(fs1)  # Surface mode on mesh 1
mode_2 = Function(fs2)  # Surface mode on mesh 2

# Surface mode on mesh 1
# extrapolated from mesh 2
mode_2_on_1 = Function(fs1)

# Allow the extrapolation of the surface
# mode defined on the mesh 2 on the mesh 1
mode_2.set_allow_extrapolation(True)

# Example to compute inner products between
# the mode 6 on the mesh 1 and the mode 7
# on the mesh 2

# Get the value of the surface
# mode 6 on mesh 1 (stored in kappa_1)
mode_1.vector()[:] = kappa_1[:, 5]

# Get the value of the surface
# mode 7 on mesh 2 (stored in kappa_2)
mode_2.vector()[:] = kappa_2[:, 6]

# Extrapolation of mode 7, originally
# defined on mesh 2, by evaluating it
# at the nodal coordinates of mesh 1.
mode_2_on_1.vector()[:] =\
np.array([mode_2(coord) for
          coord in
          mesh1.coordinates()])

# Compute the inner product and store it
# in the matching matrix
matching_matrix[5, 6] =\
assemble(inner(mode_1, mode_2_on_1)*dx)
```

Articles and conference

- M. Cances, L. Giraud, M. Bauerheim, L. Gicquel and F. Nicoud. «Robustness and reliability of state-space, frame-based modeling for thermoacoustics.» *Journal of Computational Physics.* Vol. 520, p. 113472, jan. 2025. doi: 10.1016/j.jcp.2024.113472.
- This paper was presented at the 2023 Symposium on Thermoacoustics in Combustion (SoTiC), held in Zürich.
- A part of the work conducted in this thesis, in particular the results related to the post-combustion study, was presented at the INCA 2024 conference (Initiative Nationale en Combustion Avancée), held in Marseille and co-organized by the M2P2 laboratory and Safran Tech.

Bibliography

[1] Bobby Noble et al. “Advanced Gas Turbine Combustor Health Monitoring Using Combustion Dynamics Data”. en. In: (2016) (cit. on p. 4).

[2] Lord Rayleigh. “The theory of sound; with a historical introduction by Robert Bruce Lindsay”. In: (1945) (cit. on p. 4).

[3] Thomas Schmitt et al. “Large-eddy simulations of a sub-scale liquid rocket combustor: Influence of fuel injection temperature on thermoacoustic stability”. In: *7th European Conference for Aeronautics and Aerospace Sciences (EUCASS).* 2017 (cit. on p. 5).

[4] Praveen Kasthuri et al. “Coupled interaction between acoustics and unsteady flame dynamics during the transition to thermoacoustic instability in a multi-element rocket combustor”. In: *Combustion and Flame* 240 (2022), p. 112047 (cit. on p. 5).

[5] K-U Schildmacher et al. “Unsteady flame and flow field interaction of a premixed model gas turbine burner”. In: *Proceedings of the Combustion Institute* 31.2 (2007), pp. 3197–3205 (cit. on p. 5).

[6] Lei Shi, Zhibin Yu, and Artur J Jaworski. “Vortex shedding flow patterns and their transitions in oscillatory flows past parallel-plate thermoacoustic stacks”. In: *Experimental Thermal and Fluid Science* 34.7 (2010), pp. 954–965 (cit. on p. 5).

[7] Abhishek Sharma, Ashoke De, and Sunil Kumar. “Investigation of injector-coupled combustion dynamics in a methane–oxygen combustor using large eddy simulation and dynamic mode decomposition”. In: *Physics of Fluids* 36.6 (2024) (cit. on p. 5).

[8] P Merkli and H Thomann. “Thermoacoustic effects in a resonance tube”. In: *Journal of fluid mechanics* 70.1 (1975), pp. 161–177 (cit. on p. 5).

[9] Gregory W Swift. “Thermoacoustic engines”. In: *the Journal of the Acoustical Society of America* 84.4 (1988), pp. 1145–1180 (cit. on p. 5).

[10] FEC Culick. “Combustion instabilities and Rayleigh’s criterion”. In: *Modern Research Topics in Aerospace Propulsion: In Honor of Corrado Casci.* Springer, 1991, pp. 135–151 (cit. on p. 5).

[11] Ashwini Karmarkar et al. “Impact of precessing vortex core dynamics on the thermoacoustic instabilities in a swirl-stabilized combustor”. In: *Journal of Fluid Mechanics* 946 (2022), A36 (cit. on p. 5).

[12] Geng Chen, Lihua Tang, and Brian R Mace. “Theoretical and experimental investigation of the dynamic behaviour of a standing-wave thermoacoustic engine with various boundary conditions”. In: *International Journal of Heat and Mass Transfer* 123 (2018), pp. 367–381 (cit. on p. 5).

[13] Matthias Haeringer and Wolfgang Polifke. “Time-domain Bloch boundary conditions for efficient simulation of thermoacoustic limit cycles in (can-) annular combustors”. In: *Journal of Engineering for Gas Turbines and Power* 141.12 (2019), p. 121005 (cit. on p. 5).

[14] B. Schuermans, V. Bellucci, and C. Paschereit. “Thermoacoustic Modeling and Control of Multiburner Combustion Systems”. In: *International Gas Turbine and Aeroengine Congress & Exposition, ASME Paper.* Vol. 2003-GT-38688. 2003 (cit. on pp. 5, 27).

[15] Nicolas Noiray et al. “A unified framework for nonlinear combustion instability analysis based on the describing function”. In: *Journal of Fluid Mechanics* 615 (Nov. 2008), pp. 139–167 (cit. on pp. 5, 57).

[16] Francisco Huhn and Luca Magri. “Stability, sensitivity and optimisation of chaotic acoustic oscillations”. In: *Journal of Fluid Mechanics* 882 (2020), A24 (cit. on p. 5).

[17] D Laera, G Campa, and SM Camporeale. “A finite element method for a weakly nonlinear dynamic analysis and bifurcation tracking of thermo-acoustic instability in longitudinal and annular combustors”. In: *Applied Energy* 187 (2017), pp. 216–227 (cit. on p. 5).

[18] RI Sujith, MP Juniper, and PJ Schmid. “Non-normality and nonlinearity in thermoacoustic instabilities”. In: *International Journal of Spray and Combustion Dynamics* 8.2 (2016), pp. 119–146 (cit. on p. 5).

[19] Geng Chen, Lihua Tang, and Brian R Mace. “Bistability and triggering in a thermoacoustic engine: A numerical study”. In: *International Journal of Heat and Mass Transfer* 157 (2020), p. 119951 (cit. on p. 5).

[20] Aurelien Genot, Stany Gallier, and Thierry Schuller. “Thermo-acoustic instabilities driven by fuel droplet lifetime oscillations”. In: *Proceedings of the Combustion Institute* 37.4 (2019), pp. 5359–5366 (cit. on p. 7).

[21] Nicolas Noiray and Bruno Schuermans. “Theoretical and experimental investigations on damper performance for suppression of thermoacoustic oscillations”. In: *Journal of sound and vibration* 331.12 (2012), pp. 2753–2763 (cit. on p. 7).

[22] Jean-François Bourgouin et al. “Characterization and modeling of a spinning thermoacoustic instability in an annular combustor equipped with multiple matrix injectors”. In: *Journal of Engineering for Gas Turbines and Power* 137.2 (2015), p. 021503 (cit. on p. 7).

[23] Christoph M Arndt et al. “Experimental analysis of thermo-acoustic instabilities in a generic gas turbine combustor by phase-correlated PIV, chemiluminescence, and laser Raman scattering measurements”. In: *Experiments in Fluids* 56 (2015), pp. 1–23 (cit. on p. 7).

[24] Bruno Schuermans et al. “Thermoacoustic modeling of a gas turbine using transfer functions measured under full engine pressure”. In: (2010) (cit. on p. 7).

[25] Jan Kopitz et al. “Thermoacoustic stability analysis of an annular combustion chamber with acoustic low order modeling and validation against experiment”. In: *Turbo expo: power for land, sea, and air.* Vol. 4725. 2005, pp. 583–593 (cit. on p. 7).

[26] G. Staffelbach et al. “Large Eddy Simulation of self-excited azimuthal modes in annular combustors”. In: 32 (2009), pp. 2909–2916 (cit. on pp. 7, 16).

[27] G. Staffelbach, L. Gicquel, and T. Poinsot. “Highly parallel Large Eddy Simulations of multiburner configurations in industrial gas turbines.” In: *Lecture Notes in Computational Science and Engineering - Complex effects in Large Eddy Simulation.* Ed. by Springer. Vol. 56. 2006, pp. 326–336 (cit. on p. 7).

[28] Tin-Hang Un and Salvador Navarro-Martinez. “Stochastic fields with adaptive mesh refinement for high-speed turbulent combustion”. In: *Combustion and Flame* 272 (2025), p. 113897 (cit. on p. 8).

[29] Cedric Mehl, Shuaishuai Liu, and Olivier Colin. “A strategy to couple thickened flame model and adaptive mesh refinement for the les of turbulent premixed combustion”. In: *Flow, Turbulence and Combustion* 107.4 (2021), pp. 1003–1034 (cit. on p. 8).

[30] T. Poinsot. “Prediction and control of combustion instabilities in real engines”. en. In: *Proceedings of the Combustion Institute* 36.1 (2017), pp. 1–28 (cit. on pp. 8, 14, 18).

[31] Luca Magri and Matthew P Juniper. “Adjoint-based linear analysis in reduced-order thermo-acoustic models”. In: *International Journal of Spray and Combustion Dynamics* 6.3 (2014), pp. 225–246 (cit. on p. 8).

[32] Alessandro Orchini and Matthew P Juniper. “Linear stability and adjoint sensitivity analysis of thermoacoustic networks with premixed flames”. In: *Combustion and Flame* 165 (2016), pp. 97–108 (cit. on p. 8).

[33] M. R. Bothien et al. “Time domain modelling and stability analysis of complex thermoacoustic systems”. In: *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy* 221.5 (Jan. 2007). Publisher: IMECHE, pp. 657–668 (cit. on pp. 8, 64).

[34] Iain Waugh, Matthias Geuß, and Matthew Juniper. “Triggering, bypass transition and the effect of noise on a linearly stable thermoacoustic system”. In: *Proceedings of the Combustion Institute* 33.2 (2011), pp. 2945–2952 (cit. on p. 8).

[35] Luca Magri. “Adjoint Methods as Design Tools in Thermoacoustics”. en. In: *Applied Mechanics Reviews* 71.2 (Mar. 2019), p. 020801 (cit. on p. 8).

[36] Wolfgang Polifke and C Schram. “Low-order analysis tools for aero- and thermo-acoustic instabilities”. In: *Advances in Aero-Acoustics and Thermo-Acoustics* 59 (2010) (cit. on p. 8).

[37] Michaël Bauerheim et al. “An analytical model for azimuthal thermoacoustic modes in an annular chamber fed by an annular plenum”. en. In: *Combustion and Flame* 161.5 (May 2014), pp. 1374–1389 (cit. on pp. 8, 22).

[38] Fabian Duvigneau et al. “Thermo-acoustic performance of full engine encapsulations–A numerical, experimental and psychoacoustic study”. In: *Applied Acoustics* 102 (2016), pp. 79–87 (cit. on p. 8).

[39] Gregory William Swift. “Analysis and performance of a large thermoacoustic engine”. In: *the Journal of the Acoustical Society of America* 92.3 (1992), pp. 1551–1563 (cit. on p. 8).

[40] MEH Tijani and S Spoelstra. “A high performance thermoacoustic engine”. In: *Journal of Applied Physics* 110.9 (2011) (cit. on p. 8).

[41] F Flemming, A Sadiki, and J Janicka. “Investigation of combustion noise using a LES/CAA hybrid approach”. In: *Proceedings of the Combustion Institute* 31.2 (2007), pp. 3189–3196 (cit. on p. 9).

[42] Timo Klenke et al. “Two-way hybrid LES/CAA approach including acoustic feedback loop for the prediction of thermoacoustic instabilities in technical combustors”. In: *Turbo Expo: Power for Land, Sea, and Air*. Vol. 50848. American Society of Mechanical Engineers. 2017, V04AT04A019 (cit. on p. 9).

[43] S Jaensch et al. “Hybrid CFD/low order modeling of thermoacoustic limit cycles”. In: *Sonderforschungsbereich/Transregio 40-Summer Program Report 2015* (2015) (cit. on p. 9).

[44] Matthias Haeringer and Wolfgang Polifke. “Hybrid CFD/low-order modeling of thermoacoustic limit cycle oscillations in can-annular configurations”. In: *International Journal of Spray and Combustion Dynamics* 14.1-2 (2022), pp. 143–152 (cit. on p. 9).

[45] Matthew Yoko and Matthew P Juniper. “Optimal experiment design with adjoint-accelerated Bayesian inference”. In: *Data-Centric Engineering* 5 (2024), e17 (cit. on p. 9).

[46] Renaud Gaudron and Aimee S Morgans. “Thermoacoustic stability prediction using classification algorithms”. In: *Data-Centric Engineering* 3 (2022), e17 (cit. on p. 9).

[47] Matthew P Juniper. “Machine learning for thermoacoustics”. In: *Machine Learning and its Application to Reacting Flows: ML and Combustion*. Springer International Publishing Cham, 2023, pp. 307–337 (cit. on p. 9).

[48] Mahmoud A Alamir. “An artificial neural network model for predicting the performance of thermoacoustic refrigerators”. In: *International Journal of heat and mass transfer* 164 (2021), p. 120551 (cit. on p. 9).

[49] C. Laurent et al. “A novel modal expansion method for low-order modeling of thermoacoustic instabilities in complex geometries”. en. In: *Combustion and Flame* 206 (Aug. 2019), pp. 334–348 (cit. on pp. 10, 28, 38–40, 42, 54).

[50] M. Meindl, A. Albayrak, and W. Polifke. “A state-space formulation of a discontinuous Galerkin method for thermoacoustic stability analysis”. en. In: *Journal of Sound and Vibration* 481 (Sept. 2020), p. 115431 (cit. on p. 10).

[51] TC Treurniet, FTM Nieuwstadt, and BJ Boersma. “Direct numerical simulation of homogeneous turbulence in combination with premixed combustion at low Mach number modelled by the-equation”. In: *Journal of Fluid Mechanics* 565 (2006), pp. 25–62 (cit. on pp. 13, 14).

[52] M Boger et al. “Direct numerical simulation analysis of flame surface density concept for large eddy simulation of turbulent premixed combustion”. In: *Symposium (International) on combustion*. Vol. 27. 1. Elsevier. 1998, pp. 917–925 (cit. on pp. 13, 14).

[53] P. Wolf et al. “Using LES to Study Reacting Flows and Instabilities in Annular Combustion Chambers”. In: 88 (1 2012), pp. 191–206 (cit. on pp. 13, 16).

[54] L. Y. M. Gicquel, G. Staffelbach, and T. Poinsot. “Large Eddy Simulations of gaseous flames in gas turbine combustion chambers”. In: 38.6 (2012), pp. 782–817 (cit. on pp. 13, 16).

[55] Evatt R Hawkes et al. “Direct numerical simulation of turbulent combustion: fundamental insights towards predictive models”. In: *Journal of Physics: Conference Series*. Vol. 16. 1. IOP Publishing. 2005, p. 65 (cit. on p. 15).

[56] T. Poinsot, S. Candel, and A. Trouvé. “Application of direct numerical simulation to premixed turbulent combustion”. In: 21 (1996), pp. 531–576 (cit. on p. 15).

[57] Parviz Moin and Sourabh Apte. “Large-Eddy Simulation of Realistic Gas Turbine Combustors”. In: *Aiaa Journal - AIAA J* 44 (Apr. 2006) (cit. on p. 16).

[58] K. Mahesh et al. “Large-eddy simulation of reacting turbulent flows in complex geometries”. In: *Journal of Applied Mechanics, Transactions ASME* 73.3 (2006), pp. 374–381 (cit. on p. 16).

[59] P. Wolf et al. “Massively parallel LES of azimuthal thermo-acoustic instabilities in annular gas turbines”. In: *Comptes Rendus - Mecanique* 337.6-7 (2009), pp. 385–394 (cit. on p. 16).

[60] Jochen Fröhlich and Dominic Von Terzi. “Hybrid LES/RANS methods for the simulation of turbulent flows”. en. In: *Progress in Aerospace Sciences* 44.5 (July 2008), pp. 349–377 (cit. on p. 16).

[61] Yang Zhang and Maarten Vanierschot. “Modeling capabilities of unsteady RANS for the simulation of turbulent swirling flow in an annular bluff-body combustor geometry”. In: *Applied Mathematical Modelling* 89 (Jan. 2021), pp. 1140–1154 (cit. on p. 16).

[62] L. Crocco and S. I. Cheng. *Theory of combustion instability in liquid propellant rocket motors.* Vol. Agardograph No 8. Butterworths Science, 1956 (cit. on pp. 16, 56).

[63] C. O. Paschereit et al. “Measurement of Transfer Matrices and Source Terms of Premixed Flames”. In: *Journal of Engineering for Gas Turbines and Power* 124.2 (Mar. 2002), pp. 239–247 (cit. on pp. 16, 56).

[64] Faizan H. Vance, Çetin Alanyalıoğlu, and Christian Hasse. “Analysis of Lewis number effects on dynamic response of laminar premixed flames”. en. In: *Combustion and Flame* 248 (Feb. 2023), p. 112508 (cit. on p. 16).

[65] Sébastien Ducruix, Daniel Durox, and Sébastien Candel. “Theoretical and experimental determinations of the transfer function of a laminar premixed flame”. en. In: *Proceedings of the Combustion Institute* 28.1 (Jan. 2000), pp. 765–773 (cit. on pp. 16, 57).

[66] Naman Purwar et al. “Flame response to transverse velocity excitation leading to frequency doubling and modal coupling”. en. In: *Combustion and Flame* 230 (Aug. 2021), p. 111412 (cit. on pp. 16, 17).

[67] S. Candel, C. Huynh., and T. Poinsot. “Some modeling methods of combustion instabilities”. In: *Unsteady combustion.* Nato ASI Series, Kluwer Academic Publishers, Dordrecht, 1996, pp. 83–112 (cit. on p. 16).

[68] T. Schuller, D. Durox, and S. Candel. “A unified model for the prediction of laminar flame transfer functions: comparisons between conical and V-flames dynamics”. In: 134 (2003), pp. 21–34 (cit. on p. 16).

[69] D. Durox et al. “Experimental analysis of nonlinear flame transfer functions for different flame geometries”. In: 32.1 (2009), pp. 1391–1398 (cit. on p. 16).

[70] Sébastien Ducruix, Daniel Durox, and Sébastien Candel. “Theoretical and experimental determinations of the transfer function of a laminar premixed flame”. In: 28.1 (2000), pp. 765–773 (cit. on p. 16).

[71] Xingsi Han, Jingxuan Li, and Aimee S. Morgans. “Prediction of combustion instability limit cycle oscillations by combining flame describing function simulations with a thermoacoustic network model”. en. In: *Combustion and Flame* 162.10 (Oct. 2015), pp. 3632–3647 (cit. on p. 17).

[72] P Palies et al. “Experimental Study on the Effect of Swirler Geometry and Swirl Number on Flame Describing Functions”. In: 183.7 (2011), pp. 704–717 (cit. on p. 17).

[73] T Poinsot and D Veynante. *Theoretical and numerical combustion.* R. T. Edwards, Inc., 2005 (cit. on pp. 19, 83).

[74] F. Nicoud et al. “Acoustic modes in combustors with complex impedances and multidimensional active flames”. In: 45 (2007), pp. 426–441 (cit. on pp. 20, 25, 80).

[75] K. Wieczorek et al. “Assessing non-normal effects in thermoacoustic systems with mean flow”. In: 23.10 (2011), p. 107103 (cit. on pp. 20, 80).

[76] E. Motheau, F. Nicoud, and T. Poinsot. “Using Boundary Conditions to Account for Mean Flow Effects in a Zero Mach Number Acoustic Solver”. In: 134.11, 111502 (2012), p. 111502 (cit. on p. 20).

[77] Emmanuel Motheau, Franck Nicoud, and Thierry Poinsot. “Mixed acoustic–entropy combustion instabilities in gas turbines”. en. In: *Journal of Fluid Mechanics* 749 (June 2014), pp. 542–576 (cit. on p. 20).

[78] Camilo Silva et al. “Boundary conditions for the computation of thermoacoustic modes in combustion chambers”. In: 52.6 (2014), pp. 1180–1193 (cit. on pp. 20, 80).

[79] M. S. Howe. “Influence of wall thickness on Rayleigh conductivity and flow-induced aperture tones”. In: 11.4 (1997), pp. 351–366 (cit. on p. 20).

[80] E. Gullaud et al. “Effect of multiperforated plates on the acoustic modes in combustors”. In: *Mécanique* 337.6-7 (2009), pp. 406–414 (cit. on pp. 21, 80).

[81] F. Ni et al. “Accounting for Acoustic Damping in a Helmholtz Solver”. In: 55.4 (2017), pp. 1205–1220 (cit. on pp. 21, 164).

[82] W. Polifke and C. O. Paschereit. “Determination of thermo-acoustic transfer matrices by experiment and computational fluid dynamics”. In: ERCOFTAC bulletin, 1998, p. 38 (cit. on pp. 21, 80).

[83] Giovanni Campa and Sergio Mario Camporeale. “Application of transfer matrix method in acoustics”. In: *European COMSOL Conference.* 2010 (cit. on pp. 21, 80).

[84] Matthias K Gobbert and Shiming Yang. “Numerical demonstration of finite element convergence for Lagrange elements in COMSOL Multiphysics”. In: *Proceedings of the COMSOL Conference.* Vol. 2008. 2008 (cit. on p. 24).

[85] Georg A. Mensah, Philip E. Buschmann, and Alessandro Orchini. “Iterative solvers for the thermoacoustic nonlinear eigenvalue problem and their convergence properties”. en. In: *International Journal of Spray and Combustion Dynamics* 14.1-2 (Mar. 2022). Publisher: SAGE Publications Ltd STM, pp. 30–41 (cit. on pp. 26, 145).

[86] Camilo F. Silva. “Intrinsic thermoacoustic instabilities”. en. In: *Progress in Energy and Combustion Science* 95 (Mar. 2023), p. 101065 (cit. on p. 26).

[87] Philip E. Buschmann et al. “Solution of Thermoacoustic Eigenvalue Problems With a Noniterative Method”. en. In: *Journal of Engineering for Gas Turbines and Power* 142.3 (Mar. 2020), p. 031022 (cit. on p. 26).

[88] Mark J Balas. “Modal control of certain flexible dynamic systems”. In: *SIAM Journal on Control and Optimization* 16.3 (1978), pp. 450–462 (cit. on p. 27).

[89] Robert M Kirby and Spencer J Sherwin. “Stabilisation of spectral/hp element methods through spectral vanishing viscosity: Application to fluid mechanics modelling”. In: *Computer methods in applied mechanics and engineering* 195.23-24 (2006), pp. 3128–3144 (cit. on p. 27).

[90] Martin J. Gander and Gerhard Wanner. “From Euler, Ritz, and Galerkin to Modern Computing”. en. In: *SIAM Review* 54.4 (Jan. 2012), pp. 627–666 (cit. on p. 28).

[91] Thomas Emmert et al. “Linear State Space Interconnect Modeling of Acoustic Systems”. en. In: *Acta Acustica united with Acustica* 102.5 (Sept. 2016), pp. 824–833 (cit. on p. 32).

[92] Bruno Schuermans, Valter Bellucci, and Christian Oliver Paschereit. “Thermoacoustic Modeling and Control of Multi Burner Combustion Systems”. In: *Volume 2: Turbo Expo 2003*. Atlanta, Georgia, USA: ASMEDC, Jan. 2003, pp. 509–519 (cit. on p. 33).

[93] P. M. Morse and K. U. Ingard. *Theoretical acoustics*. Vol. 332. Princeton University Press, 1968 (cit. on p. 35).

[94] C. Laurent, A. Badhe, and F. Nicoud. “Representing the geometrical complexity of liners and boundaries in low-order modeling for thermoacoustic instabilities”. en. In: *Journal of Computational Physics* 428 (Mar. 2021), p. 110077 (cit. on pp. 40, 89–92, 116, 190).

[95] João Carlos Alves Barata and Mahir Saleh Hussein. “The Moore–Penrose pseudoinverse: A tutorial review of the theory”. In: *Brazilian Journal of Physics* 42.1 (2012), pp. 146–165 (cit. on p. 42).

[96] Edward Anderson et al. *LAPACK users' guide*. SIAM, 1999 (cit. on p. 42).

[97] Fikri Öztürk and Fikri Akdeniz. "Ill-conditioning and multicollinearity". en. In: *Linear Algebra and its Applications* 321.1-3 (Dec. 2000), pp. 295–305 (cit. on p. 50).

[98] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 2013 (cit. on pp. 53, 230).

[99] K. Truffin and T. Poinsot. "Comparison and extension of methods for acoustic identification of burners". en. In: *Combustion and Flame* 142.4 (Sept. 2005), pp. 388–400 (cit. on p. 56).

[100] P. Palies et al. "Nonlinear combustion instability analysis based on the flame describing function applied to turbulent premixed swirling flames". en. In: *Combustion and Flame* 158.10 (Oct. 2011), pp. 1980–1991 (cit. on p. 57).

[101] Alexis Cuquel, Daniel Durox, and Thierry Schuller. "Impact of flame base dynamics on the non-linear frequency response of conical flames". en. In: *Comptes Rendus Mécanique*. Combustion, spray and flow dynamics for aerospace propulsion 341.1 (Jan. 2013), pp. 171–180 (cit. on p. 57).

[102] K. S. Kedia, H. M. Altay, and A. F. Ghoniem. "Impact of flame-wall interaction on premixed flame dynamics and transfer function characteristics". en. In: *Proceedings of the Combustion Institute* 33.1 (Jan. 2011), pp. 1113–1120 (cit. on p. 57).

[103] L. Y. M. Gicquel, G. Staffelbach, and T. Poinsot. "Large Eddy Simulations of gaseous flames in gas turbine combustion chambers". en. In: *Progress in Energy and Combustion Science* 38.6 (Dec. 2012), pp. 782–817 (cit. on p. 57).

[104] Christer Fureby. "LES of a Multi-burner Annular Gas Turbine Combustor". en. In: *Flow, Turbulence and Combustion* 84.3 (Apr. 2010), pp. 543–564 (cit. on p. 57).

[105] Piero Triverio. *Vector Fitting*. en. arXiv:1908.08977 [physics]. Aug. 2019 (cit. on pp. 58, 65, 141).

[106] Piero Triverio and Stefano Grivet-Talocia. "Robust Causality Characterization via Generalized Dispersion Relations". en. In: *IEEE Transactions on Advanced Packaging* 31.3 (Aug. 2008), pp. 579–593 (cit. on p. 58).

[107] P. Triverio et al. "Stability, Causality, and Passivity in Electrical Interconnect Models". en. In: *IEEE Transactions on Advanced Packaging* 30.4 (Nov. 2007), pp. 795–808 (cit. on pp. 58, 64).

[108] Piero Triverio. "Robust Causality Check for Sampled Scattering Parameters via a Filtered Fourier Transform". en. In: *IEEE Microwave and Wireless Components Letters* 24.2 (Feb. 2014), pp. 72–74 (cit. on p. 58).

[109] K Krishna and M Narasimha Murty. "Genetic K-means algorithm". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 29.3 (1999), pp. 433–439 (cit. on p. 61).

[110] Claude Brezinski and Jeannette Van Iseghem. "Padé approximations". In: *Handbook of Numerical Analysis* 3 (1994), pp. 47–222 (cit. on p. 62).

[111] Hamidreza Amindavar and James A Ritcey. "Padé approximations of probability density functions". In: *IEEE Transactions on Aerospace and Electronic Systems* 30.2 (1994), pp. 416–424 (cit. on p. 62).

[112] Mustafa Celik et al. "Pole-zero computation in microwave circuits using multipoint Padé approximation". In: *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 42.1 (1995), pp. 6–13 (cit. on p. 62).

[113] Mark Elzinga, Kathleen L Virga, and John L Prince. "Improved global rational approximation macromodeling algorithm for networks characterized by frequency-sampled data". In: *IEEE Transactions on microwave theory and techniques* 48.9 (2000), pp. 1461–1468 (cit. on p. 62).

[114] Q.Douasbin et al. "Delayed-time domain impedance boundary conditions (D-TDIBC)". In: *Journal of Computational Physics* (2018) (cit. on p. 63).

[115] B.Gustavsen and A.Semlyen. "Rational approximation of frequency domain responses by Vector Fitting". In: *IEEE Trans. Power Delivery* 14.3 (July 1999), pp. 1052–1061 (cit. on p. 63).

[116] Bruno Schuermans et al. "Thermoacoustic Modeling of a Gas Turbine Using Transfer Functions Measured Under Full Engine Pressure". en. In: *Journal of Engineering for Gas Turbines and Power* 132.11 (Nov. 2010), p. 111503 (cit. on p. 64).

[117] Priya Subramanian et al. "Distributed time lag response functions for the modelling of combustion dynamics". en. In: *Combustion Theory and Modelling* 19.2 (Mar. 2015), pp. 223–237 (cit. on p. 64).

[118] Alessandro Orchini, Georgios Rigas, and Matthew P. Juniper. “Weakly nonlinear analysis of thermoacoustic bifurcations in the Rijke tube”. en. In: *Journal of Fluid Mechanics* 805 (Oct. 2016), pp. 523–550 (cit. on p. 64).

[119] Georg A Mensah et al. “EFFECTS OF ASYMMETRY ON THERMOACOUSTIC MODES IN ANNULAR COMBUSTORS: A HIGHER-ORDER PERTURBATION STUDY”. en. In: (2018) (cit. on p. 64).

[120] S. Jaensch, C. Sovardi, and W. Polifke. “On the robust, flexible and consistent implementation of time domain impedance boundary conditions for compressible flow simulations”. en. In: *Journal of Computational Physics* 314 (June 2016), pp. 145–159 (cit. on p. 64).

[121] X. Y. Li, X. D. Li, and Christopher K. W. Tam. “Improved Multipole Broadband Time-Domain Impedance Boundary Condition”. In: *AIAA Journal* 50.4 (2012). Publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.J051361, pp. 980–984 (cit. on p. 64).

[122] Benjamin Cotté et al. “Time-Domain Impedance Boundary Conditions for Simulations of Outdoor Sound Propagation”. en. In: *AIAA Journal* 47.10 (Oct. 2009), pp. 2391–2403 (cit. on p. 64).

[123] C. Sanathanan and J. Koerner. “Transfer function synthesis as a ratio of two complex polynomials”. In: *IEEE Transactions on Automatic Control* 8.1 (1963), pp. 56–58 (cit. on p. 64).

[124] John Bechhoefer. “Kramers–Kronig, Bode, and the meaning of zero”. en. In: *American Journal of Physics* 79.10 (Oct. 2011), pp. 1053–1059 (cit. on p. 64).

[125] Stefano Grivet-Talocia and Bjorn Gustavsen. *Passive macromodeling: Theory and applications.* John Wiley & Sons, 2015 (cit. on pp. 64, 65).

[126] Q. Douasbin et al. “Delayed-time domain impedance boundary conditions (D-TDIBC)”. en. In: *Journal of Computational Physics* 371 (Oct. 2018), pp. 50–66 (cit. on p. 64).

[127] Avraham Hirschberg and SW Rienstra. *An introduction to acoustics.* 1992 (cit. on p. 64).

[128] B. Gustavsen and A. Semlyen. “Rational approximation of frequency domain responses by vector fitting”. In: *IEEE Transactions on Power Delivery* 14.3 (July 1999), pp. 1052–1061 (cit. on p. 65).

[129] Hans Petter Langtangen and Anders Logg. *Solving PDEs in Python*. en. Cham: Springer International Publishing, 2016 (cit. on pp. 66, 141).

[130] Marc Snir. *MPI–the Complete Reference: The MPI core*. en. Google-Books-ID: x79puJ2YkroC. MIT Press, 1998 (cit. on p. 66).

[131] Lisandro Dalcín, Rodrigo Paz, and Mario Storti. “MPI for Python”. In: *Journal of Parallel and Distributed Computing* 65.9 (Sept. 2005), pp. 1108–1115 (cit. on p. 66).

[132] Lisandro D. Dalcin et al. “Parallel distributed computing using Python”. en. In: *Advances in Water Resources* 34.9 (Sept. 2011), pp. 1124–1139 (cit. on p. 66).

[133] John Dolbow and Ted Belytschko. “Numerical integration of the Galerkin weak form in meshfree methods”. In: *Computational mechanics* 23.3 (1999), pp. 219–230 (cit. on p. 69).

[134] Martin Sandve Alnæs. “UFL: a finite element form language”. en. In: *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*. Ed. by Anders Logg, Kent-Andre Mardal, and Garth Wells. Berlin, Heidelberg: Springer, 2012, pp. 303–338 (cit. on p. 70).

[135] Anders Logg et al. “FFC: the FEniCS form compiler”. en. In: *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*. Ed. by Anders Logg, Kent-Andre Mardal, and Garth Wells. Berlin, Heidelberg: Springer, 2012, pp. 227–238 (cit. on p. 70).

[136] Satish Balay et al. “PETSc”. In: *See http://www. mcs. anl. gov/petsc* (2001) (cit. on p. 70).

[137] Vicente Hernandez, Jose E. Roman, and Vicente Vidal. “SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems”. en. In: *ACM Transactions on Mathematical Software* 31.3 (Sept. 2005), pp. 351–362 (cit. on p. 71).

[138] Lisandro Dalcin. “SLEPc for Python”. In: *Trans. Math. Softw* 31.3 (2005), pp. 351–362 (cit. on p. 71).

[139] Stewart. *Matrix Algorithms. Volume 2: Eigensystems*. Society for Industrial and Applied Mathematics, 2001 (cit. on p. 71).

[140] Henk A. van der Vorst. “Computational methods for large eigenvalue problems”. In: *Handbook of Numerical Analysis* 8 (2002), pp. 3–179 (cit. on p. 71).

[141] Jörg Liesen and Zdenek Strakos. *Krylov Subspace Methods: Principles and Analysis*. Oxford University Press, Oct. 2012 (cit. on p. 72).

[142] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003 (cit. on p. 72).

[143] Michael Eiermann and Oliver G Ernst. “Geometric aspects of the theory of Krylov subspace methods”. In: *Acta Numerica* 10 (2001), pp. 251–312 (cit. on p. 72).

[144] Richard B Lehoucq and Danny C Sorensen. “Deflation techniques for an implicitly restarted Arnoldi iteration”. In: *SIAM Journal on Matrix Analysis and Applications* 17.4 (1996), pp. 789–821 (cit. on p. 72).

[145] G. W. Stewart. “A Krylov–Schur Algorithm for Large Eigenproblems”. In: *SIAM Journal on Matrix Analysis and Applications* 23.3 (2002), pp. 601–614 (cit. on p. 72).

[146] Gerard LG Sleijpen and Henk A Van der Vorst. “A Jacobi–Davidson iteration method for linear eigenvalue problems”. In: *SIAM review* 42.2 (2000), pp. 267–293 (cit. on p. 73).

[147] Peter Sonneveld. “CGS, a fast Lanczos-type solver for nonsymmetric linear systems”. In: *SIAM journal on scientific and statistical computing* 10.1 (1989), pp. 36–52 (cit. on p. 73).

[148] Thomas Ericsson and Axel Ruhe. “The Spectral Transformation Lanczos Method for the Numerical Solution of Large Sparse Generalized Symmetric Eigenvalue Problems”. In: *Mathematics of Computation* 35.152 (1980), pp. 1251–1268 (cit. on p. 74).

[149] Roger G. Grimes, John G. Lewis, and Horst D. Simon. “A Shifted Block Lanczos Algorithm for Solving Sparse Symmetric Generalized Eigenproblems”. In: *SIAM Journal on Matrix Analysis and Applications* 15.1 (1994), pp. 228–272 (cit. on p. 74).

[150] Christoph J Mack and Peter J Schmid. “A preconditioned Krylov technique for global hydrodynamic stability analysis of large-scale compressible flows”. In: *Journal of Computational Physics* 229.3 (2010), pp. 541–560 (cit. on p. 74).

[151] H Voss. “A Jacobi–Davidson method for nonlinear and nonsymmetric eigenproblems”. In: *Computers & Structures* 85.17-18 (2007), pp. 1284–1292 (cit. on p. 74).

[152] Philippe Guillaume, Yousef Saad, and Masha Sosonkina. “Rational approximation preconditioners for sparse linear systems”. In: *Journal of computational and applied mathematics* 158.2 (2003), pp. 419–442 (cit. on p. 74).

[153] T. Reis and H.J. Wilson. “Rolie-Poly fluid flowing through constrictions: Two distinct instabilities”. en. In: *Journal of Non-Newtonian Fluid Mechanics* 195 (May 2013), pp. 77–87 (cit. on p. 77).

[154] M. S. Howe. “On the Theory of Unsteady High Reynolds Number Flow Through a Circular Aperture”. In: *, Mathematical and Physical Sciences* 366.1725 (1979), pp. 205–223 (cit. on pp. 80, 115, 123, 130).

[155] Eleuterio F Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction.* Springer Science & Business Media, 2013 (cit. on p. 82).

[156] Randall J LeVeque. *Finite volume methods for hyperbolic problems.* Vol. 31. Cambridge university press, 2002 (cit. on p. 82).

[157] MJJ Nijhof, YH Wijnant, and A De Boer. “An acoustic finite element including viscothermal effects”. In: *14th International Congress on Sound and Vibration, ICSV 2007.* International Institute of Acoustics and Vibration (IIAV). 2007 (cit. on p. 88).

[158] E Rodarte et al. “Sound attenuation in tubes due to visco-thermal effects”. In: *Journal of Sound and Vibration* 231.5 (2000), pp. 1221–1242 (cit. on p. 88).

[159] Miguel Molerón, Marc Serra-Garcia, and Chiara Daraio. “Visco-thermal effects in acoustic metamaterials: from total transmission to total reflection and high absorption”. In: *New Journal of Physics* 18.3 (2016), p. 033003 (cit. on p. 88).

[160] A Duclos, Denis Lafarge, and Vincent Pagneux. “Transmission of acoustic waves through 2D phononic crystal: visco-thermal and multiple scattering effects”. In: *The European Physical Journal Applied Physics* 45.1 (2009), p. 11302 (cit. on p. 88).

[161] M. S. Howe. *Acoustics of Fluid-Structure Interaction.* Cambridge University Press, 1998 (cit. on pp. 115, 123, 130).

[162] Gorkem Oztarlik et al. “Suppression of instabilities of swirled premixed flames with minimal secondary hydrogen injection”. en. In: *Combustion and Flame* 214 (Apr. 2020), pp. 266–276 (cit. on pp. 140, 141, 145).

[163] Gorkem Oztarlik. “Numerical and Experimental Investigations of Combustion Instabilities of Swirled Premixed Methane-Air Flames With Hydrogen Addition”. In: (2020) (cit. on p. 141).

[164] T. Schønfeld and M. Rudgyard. “Steady and Unsteady Flows Simulations Using the Hybrid Flow Solver AVBP”. In: 37.11 (1999), pp. 1378–1385 (cit. on p. 141).

[165] Igor A. Baratta et al. *DOLFINx: The next generation FEniCS problem solving environment.* eng. Dec. 2023 (cit. on p. 141).

[166] L. Benoit and F. Nicoud. “Numerical assessment of thermo-acoustic instabilities in gas turbines”. In: 47.8-9 (2005), pp. 849–855 (cit. on p. 142).

[167] Kah Joon Yong et al. *Categorization of Thermoacoustic Modes in an Ideal Resonator with Phasor Diagrams.* arXiv:2211.03842 [physics]. Nov. 2022 (cit. on p. 145).

[168] Hugo Paniez et al. “High-frequency thermo-acoustic instability in a dual swirl H2 burner”. In: *Proceedings of the Combustion Institute* 40.1-4 (2024), p. 105679 (cit. on p. 147).

[169] Sylvain Marragou et al. “Flame stabilization and pollutant emissions from a H2/air dual swirl coaxial injector at elevated pressure”. In: *International Journal of Hydrogen Energy* 100 (2025), pp. 163–172 (cit. on p. 147).

[170] Roberto Meloni et al. “Analysis of the NOx formation pathways in a partially premixed burner operated with pure hydrogen”. In: *Proceedings of the 46th Meeting of the Italian Section of The Combustion Institute.* 2024 (cit. on p. 147).

[171] A. Huber and W. Polifke. “Dynamics of Practical Premixed Flames, Part I: Model Structure and Identification”. en. In: *International Journal of Spray and Combustion Dynamics* 1.2 (June 2009), pp. 199–228 (cit. on p. 150).

[172] V. Latour et al. “Experiments on symmetry breaking of azimuthal combustion instabilities and their analysis combining acoustic energy balance and flame describing functions”. en. In: *Journal of Fluid Mechanics* 985 (Apr. 2024), A31 (cit. on pp. 158–160, 162, 163, 167, 170, 171).

[173] Mérouane Atig, Jean-Pierre Dalmont, and Joël Gilbert. “Termination impedance of open-ended cylindrical tubes at high sound pressure level”. In: *Comptes Rendus Mécanique* 332.4 (2004), pp. 299–304 (cit. on p. 160).

[174] D.W. Bechert. “Sound absorption caused by vorticity shedding, demonstrated with a jet flow”. In: *Journal of Sound and Vibration* 70.3 (1980), pp. 389–405 (cit. on p. 164).

[175] I.D.J. DUPÈRE and A.P. DOWLING. “THE ABSORPTION OF SOUND NEAR ABRUPT AXISYMMETRIC AREA EXPANSIONS”. In: *Journal of Sound and Vibration* 239.4 (2001), pp. 709–730 (cit. on p. 164).

[176] *A LES Based Sound Absorption Analysis of High-Amplitude Waves Through an Orifice With Bias Flow.* Vol. Volume 2: Combustion, Fuels and Emissions, Parts A and B. Turbo Expo: Power for Land, Sea, and Air. June 2011, pp. 613–622 (cit. on p. 164).

[177] Véranika Latour et al. “Assessing Transfer Matrix Models and Measurements Using Acoustic Energy Conservation Principles”. In: *Journal of Engineering for Gas Turbines and Power* 146.1 (2024) (cit. on p. 164).

[178] Davide Laera et al. “Modelling of Thermoacoustic Combustion Instabilities Phenomena: Application to an Experimental Test Rig”. In: *Energy Procedia* 45 (2014). ATI 2013 - 68th Conference of the Italian Thermal Machines Engineering Association, pp. 1392–1401 (cit. on p. 164).

[179] J Lovett et al. “Development needs for advanced afterburner designs”. In: *40th AIAA/ASME/SAE/ASEE joint propulsion conference and exhibit.* 2004, p. 4192 (cit. on p. 171).

[180] Abdulla Ghani et al. “LES of longitudinal and transverse self-excited combustion instabilities in a bluff-body stabilized turbulent premixed flame”. In: *Combustion and Flame* 162.11 (2015), pp. 4075–4083 (cit. on p. 171).

[181] S Manikandan and RI Sujith. “Rate dependent transition to thermoacoustic instability via intermittency in a turbulent afterburner”. In: *Experimental Thermal and Fluid Science* 114 (2020), p. 110046 (cit. on p. 171).

[182] Balázs Vincze et al. “Models for Large-Eddy Simulation of reheat combustion”. In: *Proceedings of the Combustion Institute* 40.1-4 (2024), p. 105216 (cit. on p. 172).

[183] Nejat Olgac et al. “Parametric investigation of thermoacoustic instability (TAI) in a rijke tube: a time-delay perspective”. In: *International Journal of Spray and Combustion Dynamics* 7.1 (2015), pp. 39–68 (cit. on p. 184).

[184] Carlo Filippo Quaglia. "A method for the numerical analysis of combustion instabilities with an application to afterburner screech". PhD thesis. 2015 (cit. on p. 184).

[185] Guillaume Jourdain and Lars-Erik Eriksson. "Time-domain Modeling Of Screech-damping Liners In Combustors And Afterburner". In: *ISABE-2011-1113*. 2011 (cit. on p. 184).

[186] Perry L Blackshear, Warren D Rayle, and Leonard K Tower. *Experimental determination of gas motion accompanying screeching combustion in a 6-inch simulated afterburner*. Tech. rep. 1953 (cit. on p. 184).

[187] Yousef Saad. *Numerical methods for large eigenvalue problems: revised edition*. SIAM, 2011 (cit. on pp. 228, 229).