GABRIEL SABINO DE OLIVEIRA

Implementation of numerical-computational tools for fluid-structure interaction analysis in industrial cases



FEDERAL UNIVERSITY OF UBERLÂNDIA
SCHOOL OF MECHANICAL ENGINEERING
GRADUATE PROGRAM IN MECHANICAL ENGINEERING

2025

GABRIEL SABINO DE OLIVEIRA

Implementation of numerical-computational tools for fluid-structure interaction analysis in industrial cases

Dissertation submitted to the Graduate Program in Mechanical Engineering at the Federal University of Uberlândia, as part of the requirements for the degree of Master in Mechanical Engineering.

Area of Concentration: Heat Transfer and Fluid Mechanics
Research Line: Computational Fluid Dynamics

Prof. Dr. João Marcelo Vedovotto,
Advisor

Prof. Dra. Polliana Cândida Oliveira Martins,
Co-advisor

Prof. Dr. Gilmar Guimarães
Coordinator of the Graduate Program in Mechanical Engineering

Uberlândia, Minas Gerais
2025

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU
com dados informados pelo(a) próprio(a) autor(a).

O48 2025

Oliveira, Gabriel Sabino de, 2000-
Implementation of numerical-computational tools for fluid-structure interaction analysis in industrial cases [recurso eletrônico] / Gabriel Sabino de Oliveira. - 2025.

Orientador: João Marcelo Vedovotto.
Coorientadora: Polliana Cândida Oliveira Martins.
Dissertação (Mestrado) - Universidade Federal de Uberlândia, Pós-graduação em Engenharia Mecânica.
Modo de acesso: Internet.
Disponível em: http://doi.org/10.14393/ufu.di.2025.130
Inclui bibliografia.
Inclui ilustrações.

1. Engenharia mecânica. I. Vedovotto, João Marcelo , 1981-, (Orient.). II. Martins, Polliana Cândida Oliveira ,1989-, (Coorient.). III. Universidade Federal de Uberlândia. Pós-graduação em Engenharia Mecânica. IV. Título.

CDU: 621

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:

Gizele Cristine Nunes do Couto - CRB6/2091
Nelson Marcos Ferreira - CRB6/3074

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
Coordenação do Programa de Pós-Graduação em Engenharia Mecânica
Av. João Naves de Ávila, nº 2121, Bloco 1M, Sala 212 - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902
Telefone: (34) 3239-4282 - www.posmecanicaufu.com.br - secposmec@mecanica.ufu.br

ATA DE DEFESA - PÓS-GRADUAÇÃO

Programa de Pós-Graduação em:	Engenharia Mecânica				
Defesa de:	Dissertação de Mestrado Acadêmico, número 656, PPGEM				
Data:	25/02/2025	Hora de início:	10:00	Hora de encerramento:	12:00
Matrícula do Discente:	12412EMC008				
Nome do Discente:	Gabriel Sabino de Oliveira				
Título do Trabalho:	Implementation of numerical-computational tools for fluid-structure interaction analysis in industrial cases				
Área de concentração:	Transferência de Calor e Mecânica dos Fluidos				
Linha de pesquisa:	Dinâmica dos Fluidos e Transferência de Calor				
Projeto de Pesquisa de vinculação:	Análise de interação fluido-estrutura em banco de tubos de caldeiras- MFLab, Petrobras				

Reuniu-se por videoconferência, a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Engenharia Mecânica, assim composta: Prof. Dr. Aldemir Aparecido Cavallini Junior - UFU FEMEC; Prof. Dr. Adriano Todorovic Fabro - UNB, Polliana Cândida Oliveira Martins - UNB, co-orientadora e convidada e Prof. Dr. João Marcelo Vedovotto - UFU FEMEC, orientador do candidato.

Iniciando os trabalhos o(a) presidente da mesa, Dr(a). João Marcelo Vedovotto, apresentou a Comissão Examinadora e o candidato(a), agradeceu a presença do público, e concedeu ao Discente a palavra para a exposição do seu trabalho. A duração da apresentação do Discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir o senhor(a) presidente concedeu a palavra, pela ordem sucessivamente, aos(às) examinadores(as), que passaram a arguir o(a) candidato(a). Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o(a) candidato(a):

Aprovado.

Esta defesa faz parte dos requisitos necessários à obtenção do título de Mestre.

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.

Documento assinado eletronicamente por **João Marcelo Vedovotto**, **Professor(a) do Magistério Superior**, em 25/02/2025, às 11:59, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Aldemir Aparecido Cavalini Junior**, **Professor(a) do Magistério Superior**, em 25/02/2025, às 12:00, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Adriano Todorovic Fabro**, **Usuário Externo**, em 25/02/2025, às 12:04, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **6003697** e o código CRC **283884F5**.

Referência: Processo nº 23117.000681/2025-61 SEI nº 6003697

GABRIEL SABINO DE OLIVEIRA

Implementation of numerical-computational tools for fluid-structure interaction analysis in industrial cases

Expected Dissertation by the Graduate Program in Mechanical Engineering at the Federal University of Uberlândia.

Area of Concentration: Heat Transfer and Fluid Mechanics
Research Line: Computational Fluid Dynamics

Uberlândia, february 25, 2025.

Examination Committee:

Prof. Dr. João Marcelo Vedovotto (UFU)
Advisor

Prof. Dr. Aldemir Aparecido Cavallini Junior (UFU)
Internal member of the Examining Committee

Prof. Dr. Adriano Todorovic Fabro (UnB)
External member of the Examining Committee

To all those who dream with me,
who share the same hope,
who believe in the journey even without seeing the destination.
May our dreams cross paths, strengthen, and come true.
To you, my eternal gratitude.
Me.

Acknowledgements

I would like to express my gratitude to the professors who have made their profession a tool for transforming lives. In particular, I extend my sincere appreciation to Dr. João Marcelo Vedovotto for his invaluable teachings, support, and guidance. To Dr. Polliana Cândida Oliveira Martins, whose advice, mentorship, and lessons I will carry with me throughout my life. To Dr. Aldemir Aparecido Cavallini Junior, whose guidance and support played a fundamental role in the development of this work. His advice and insights have been invaluable to my growth, both academically and personally.

I would like to express my gratitude to Dr. Pedro Henrique Augusto Mota, who taught me that research goes beyond awakening to the world — it is a way of life, a journey of learning and sharing. Thank you for the opportunity to learn from you and for your companionship. I would also like to thank Eng. Dr. Ricardo Serfaty for his invaluable contributions in the industrial field, as well as for the lessons that I will carry with me in both my professional and personal life.

I am also grateful to the professors who kindly accepted to be part of my Dissertation Examining Committee, contributing their valuable scientific expertise: Dr. Aldemir Aparecido Cavallini Junior and Dr. Adriano Todorovic Fabro.

I would like to acknowledge the exceptional technical team of the Fluid Mechanics Laboratory (MFLab), a group of people from whom I not only learned scientific knowledge but also life lessons. In particular, I am deeply grateful to those who played a special role in my journey and became true friends: Pedro, João, Luiz, Rafael, Danilo, Estevan, Mauro, and Iago. I will always cherish our afternoon coffee meetings.

A special thank you to those who have lived my dreams with me, providing unwavering support and inspiration. To my parents, Flávia and Geraldo, and my grandparents, Valdeci and Sebastião, who walked so I could run. To my brother, Fernando, who always support and believed in me. To my uncles, Fabrício and Fabiano, who taught me the true value of knowledge. To my friends, who, despite the physical distance, have never stopped supporting and believing in me. And to Muriel, my beloved, who shares life and dreams with me, always encouraging and believing in our journey together. My love and gratitude to you all.

I am deeply grateful to the entire technical support team at the Fluid Mechanics Laboratory (MFLab) for their dedication, which made this work possible. I also extend my sincere appreciation to Petróleo Brasileiro S.A. (Petrobras), the National Council for Scientific and Technological Development (CNPq), the Minas Gerais State Agency for Research and Development (FAPEMIG), and the Coordination for the Improvement of Higher Education Personnel (CAPES) for their invaluable financial and technical support.

"All models are wrong, but some are useful."
George E. P. Box

OLIVEIRA, G. S. **Implementation of numerical-computational tools for fluid-structure interaction analysis in industrial cases**. 2025. 134f. Master's Dissertation, Federal University of Uberlândia, Uberlândia/MG, Brazil.

Abstract

Fluid-Structure Interaction (FSI) is a multiphysics phenomenon that describes the interaction between a moving fluid and a deformable or movable structure. This work develops numerical-computational tools for FSI simulations, using the MFSim software in conjunction with external finite element models. The main goal was to create a methodology to couple structural models with fluid dynamics simulations, focusing on industrial applications such as the recovery boiler.

The adopted approach involves the development of a mesh coupling algorithm and interpolating functions, allowing the association of structural and Lagrangian meshes for use in the immersed boundary method within a parallel execution environment. Additionally, a communication interface was created between MFSim and finite element software, such as Code_Aster and Salome-Meca, promoting the coupling of structural models with fluid dynamics simulations.

The mesh association algorithm and interpolating functions were applied in a static structural analysis, where the immersed boundary followed the displacement field of the structure. A robustness test was also performed, where the structure underwent large-scale deformations, exceeding one meter. In both cases, the mesh association algorithm and interpolating functions were found to be robust. The numerical and computational tools were validated by comparing numerical results with experimental data available in the literature. The data coupling between MFSim and Salome-Meca was validated through the modal analysis of a pipe subjected to internal pressure and filled with oil.

The developed computational tools were also applied in the modal analysis and fluid-structure interaction of a rectangular plate exposed to high temperatures. In all cases, the results showed good agreement with the literature, demonstrating the applicability of the developed approach.

The results of this work indicate that the developed numerical and computational tools allow MFSim to utilize external finite element libraries, enabling the coupling of structural models with computational fluid dynamics tools for fluid-structure interaction analyses.

Keywords: Fluid-Structure Interaction, Computational Modeling, Immersed Boundary Method, Finite Elements, Numerical Simulation.

OLIVEIRA, G. S. **Implementação de ferramentas numérico-computacionais para análise de interação fluido-estrutura em casos industriais.**. 2025. 134f. Dissertação de Mestrado, Universidade Federal de Uberlândia, Uberlândia/MG, Brasil.

Resumo

A Interação Fluido-Estrutura (FSI) é um fenômeno multifísico que descreve a interação entre um fluido em movimento e uma estrutura deformável ou móvel. Este trabalho desenvolve ferramentas numérico-computacionais para simulações de FSI, utilizando o software MFSim em conjunto com modelos externos de elementos finitos. O objetivo principal foi criar uma metodologia para acoplar modelos estruturais a simulações fluidodinâmicas, com foco em aplicações industriais, como a caldeira recuperadora.

A abordagem adotada envolve o desenvolvimento de um algoritmo de acoplamento de malhas e funções interpoladoras, permitindo a associação de malhas estruturais e Lagrangianas para uso no método da fronteira imersa, em um ambiente de execução paralelo. Além disso, foi criada uma interface de comunicação entre o MFSim e softwares de elementos finitos, como Code_Aster e Salome-Meca, promovendo o acoplamento dos modelos estruturais com as simulações fluidodinâmicas.

O algoritmo de associação de malhas e as funções interpoladoras foram aplicados em uma análise estrutural estática, na qual a fronteira imersa acompanhava o campo de deslocamento da estrutura. Também foi realizado um teste de robustez, no qual a estrutura sofreu deformações de grande escala, superiores a um metro. Em ambos os casos, tanto o algoritmo de associação de malha quanto as funções interpoladoras se mostraram robustos. As ferramentas numéricas e computacionais foram validadas por meio da comparação de resultados numéricos com dados experimentais disponíveis na literatura. O acoplamento de dados entre o MFSim e o Salome-Meca foi validado por meio da análise modal de um tubo sujeito a pressão interna e preenchido com óleo.

As ferramentas computacionais desenvolvidas também foram aplicadas na análise modal e na interação fluido-estrutura de uma placa retangular exposta a altas temperaturas. Em todos os casos, os resultados demonstraram boa concordância com a literatura, evidenciando a aplicabilidade do desenvolvimento realizado.

Os resultados deste trabalho indicam que as ferramentas numéricas e computacionais desenvolvidas permitem ao MFSim utilizar bibliotecas externas de elementos finitos, viabilizando o acoplamento de modelos estruturais com ferramentas de dinâmica dos fluidos computacional para análises de interação fluido-estrutura.

Palavras-chave: Interação Fluido-Estrutura, Modelagem Computacional, Método de Fronteira Imersa, Elementos Finitos, Simulação Numérica.

List of Figures

Figure 1 – Scheme of fluid–structure interaction (FSI), highlighting the three main domains (Fluid Mechanics, Structural Mechanics, and Dynamics) and their interactions. 22
Figure 2 – Computer-Aided Design (CAD) model of the recovery boiler developed by MFLab in partnership with Petrobras. 41
Figure 3 – Workflow of the unidirectional coupling process between Salome-Meca and MFSim. 44
Figure 4 – Diagram illustrating the computational framework for fluid–structure interaction. 56
Figure 5 – Representation of a Lagrangian mesh and Lagrangian points. 57
Figure 6 – Representation of the structural mesh (a) and the Lagrangian mesh (b) for the pipeline of a heat exchanger system. 65
Figure 7 – Orthogonal projection of a Lagrangian vertex onto a 1D structural element. 66
Figure 8 – Orthogonal projection of a Lagrangian vertex onto a 2D structural element. 67
Figure 9 – Curvature visualization in the structural mesh. 68
Figure 10 – Curvature visualization in the structural mesh in a 2D view. 69
Figure 11 – Lagrangian vertex mesh visualization. 69
Figure 12 – Flowchart of the orthogonal projection verification and correction process. 71
Figure 13 – Coupling scheme between the structural and Lagrangian meshes. 72
Figure 14 – Interpolation scheme using orthogonal projection and the barycentric method adapted for the geometry of 1D and 2D structural elements. . . 76
Figure 15 – Flowchart of the structural vibration analysis process, including the calculation of natural frequencies, vibration modes, and the assignment of immersed boundary points in MFSim. 81
Figure 16 – Variation of Young's modulus as a function of temperature. Curve adapted from Seif et al. (2016). 92
Figure 17 – Diagram illustrating how the finite element model is created and exported from Salome-Meca. 95
Figure 18 – Diagram illustrating how the modal analysis is performed within the Salome-Meca software. 96
Figure 19 – Diagram illustrating how the external structural model is used in MFSim. 97
Figure 20 – Boundary conditions and local application point of the static load. . . . 98
Figure 21 – Nodal displacement field obtained from Salome-Meca (a) and MFSim (b). 99

Figure 22 – Structural displacement obtained in MFSim (a) and the deformed immersed boundary mesh (b). 100
Figure 23 – Overlay of the deformed Lagrangian mesh on the deformed structural mesh. 101
Figure 24 – Representation of the deformed structural mesh (a) and the deformed Lagrangian mesh (b) under a large fictitious load. 102
Figure 25 – Bending (a) and torsion (b) vibration modes obtained using Salome-Meca. 104
Figure 26 – Pipe structural mesh. 107
Figure 27 – Curve of the elastic modulus of stainless steel (1Cr18Ni9Ti) as a function of temperature. 109
Figure 28 – Finite element mesh of the rectangular plate model. 110
Figure 29 – Eulerian domain created for fluid-structure interaction simulation. . . . 113
Figure 30 – Location of the applied excitation force and measurement points of the structural response. 114
Figure 31 – Comparison between the first three vibration modes of the plate obtained in Salome-Meca (a) and MFSim (b). 115
Figure 32 – Comparison of displacement over time at different temperatures for measurement point E. 117
Figure 33 – Structural response in the frequency domain for the cases of 200°C and 600°C, measurement point E. 118

List of Tables

Table 1 – MFSim Code Applications and References (MOTA, 2023; VASCONCELLOS, 2024). 43
Table 2 – Comparison of natural frequencies for the first 10 vibration modes. . . . 105
Table 3 – Material properties of the pipe and oil density. 106
Table 4 – Comparison between experimental natural frequencies from Massa et al. (2009) and computational results. 107
Table 5 – Dimensions of the plate used in the experiment (WU et al., 2016). . . . 109
Table 6 – Comparison of natural frequencies obtained experimentally by Wu et al. (2016) and in the present study for the first vibration mode of the plate. 110
Table 7 – Comparison of natural frequencies obtained experimentally by Wu et al. (2016) and in the present study for the second vibration mode of the plate.111
Table 8 – Comparison of natural frequencies obtained experimentally by Wu et al. (2016) and in the present study for the third vibration mode of the plate.111
Table 9 – Comparison of natural frequencies obtained numerically by Wu et al. (2016) and in the present study for the first vibration mode of the plate. 112
Table 10 – Comparison of natural frequencies obtained numerically by Wu et al. (2016) and in the present study for the second vibration mode of the plate.112
Table 11 – Comparison of natural frequencies obtained numerically by Wu et al. (2016) and in the present study for the third vibration mode of the plate.112
Table 12 – Comparison between the natural frequencies obtained experimentally by Wu et al. (2016) and those computed through fluid-structure interaction analysis in this study. 117
Table 13 – Summary of Code_Aster, Salome, and Salome-Meca softwares functionalities. 134

List of abbreviations and acronyms

Institutions and Organizations

UFU	Federal University of Uberlândia
MFLab	Fluid Mechanics Laboratory
Petrobras	Petróleo Brasileiro S.A.
CNPq	National Council for Scientific and Technological Development
FAPEMIG	Minas Gerais State Agency for Research and Development
CAPES	Coordination for the Improvement of Higher Education Personnel
EDF	Électricité de France
CEA	French Alternative Energies and Atomic Energy Commission

Units and Measurements

m	meter (length)
kg	kilogram (mass)
s	second (time)
N	newton (force)
Pa	pascal (pressure)
°C	degree Celsius (temperature)

Mathematical Operators

$\frac{\partial}{\partial t}$	Partial derivative with respect to time
∇	Gradient operator
$\nabla\cdot$	Divergence of a vector field
∇^2	Laplacian operator

$\cdot$	Dot product
$\times$	Cross product

List of symbols

Primary Symbols

ρ	Density of the fluid or solid
μ	Dynamic viscosity
ν	Poisson's ratio
σ	Stress
ε	Strain
E	Young's modulus (elastic modulus)
T	Temperature
k	Thermal conductivity
$\mathbf{u}$	Fluid velocity vector (u, v, w)
p	Pressure
Re	Reynolds number ($Re = \frac{\rho U D}{\mu}$)
St	Strouhal number ($St = \frac{fD}{U}$)
ω	Angular frequency
D	Characteristic diameter
x	x-direction
y	y-direction
z	z-direction
1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional

Subscripts

f	Refers to fluid (e.g., ρ_f)

s	Refers to structure (e.g., ρ_s)
i, j, k	Coordinate or summation indices (used in tensors and matrices)
n	Normal direction or time step index n
∞	Upstream flow condition or reference value at "infinity" (e.g., u_∞)

Contents

Acknowledgements **8**

List of abbreviations and acronyms **15**

List of symbols **17**

Contents **19**

1 INTRODUCTION **22**
1.1 Contextualization and Background **24**
1.2 Objectives **25**
1.3 Master thesis structure **25**

2 LITERATURE REVIEW **27**
2.1 Fluid-structure Iteration **27**
2.1.1 Areas of Application 27
2.1.2 Approaches to Modeling FSI 28
2.1.2.1 Monolithic Approach 28
2.1.2.2 Partitioned Approach 28
2.1.2.3 One-Way and Two-Way Coupling 29
2.1.3 Dynamic Analysis of FSI 30
2.1.4 Finite Element Method 31
2.1.5 Research on FSI at MFLab 32
2.2 Dynamics and Mechanisms of Fluid-Structure Interaction **33**
2.2.1 Vortex Shedding 33
2.2.2 Wake Induced Vibration 34
2.2.3 Turbulent buffeting 36
2.2.4 Fluid-Elastic Instability 37

3 METHODOLOGY **40**
3.1 Description of the Industrial Case **40**
3.2 Approach to the case study **41**
3.2.1 MFSim: a multiphysics simulation software 42
3.2.2 Salome-Meca 42
3.2.3 Coupling MFSim with Salome-Meca 43
3.3 Mathematical Modeling **44**
3.3.1 Fluid Domain 44

3.3.1.1 Mass Balance . . . 44
3.3.1.2 Linear Momentum Balance . . . 44
3.3.1.3 Thermal Energy Balance . . . 45
3.3.2 Structure Domain . . . 45
3.3.2.1 Stress-Strain Relationship . . . 45
3.3.2.2 Equation of Motion . . . 48
3.3.2.3 Modal Analysis . . . 51
3.4 Fluid-Structure Interaction Coupling . . . 52
3.4.1 Weak Coupling . . . 52
3.4.2 Strong Coupling . . . 53
3.4.3 Numerical Integration . . . 53

4 NUMERICAL AND COMPUTATIONAL MODELING . . . 54
4.1 Immersed Boundary Method . . . 54
4.1.1 Multi-Direct Forcing (MDF) Method . . . 55
4.1.2 Eulerian-Lagrangian Coupling . . . 55
4.2 IBM-FEM Coupling . . . 56
4.2.1 Lagrangian mesh association . . . 58
4.2.1.1 Search for Orthogonal Projection . . . 59
4.2.1.2 Graph-Based Structural Mesh Representation Method . . . 62
4.2.1.3 Challenges and Limitations of the Projection Method . . . 63
4.2.2 Interpolation Methods . . . 72
4.2.2.1 Barycentric Interpolation for 1D Structural Elements . . . 76
4.2.2.2 Barycentric Interpolation for 2D Structural Elements . . . 77
4.3 Finite Element Method . . . 80
4.3.1 Purpose of Using External Models . . . 80
4.3.2 Structural System Representation . . . 81
4.3.3 Types of Structural Elements . . . 82
4.3.3.1 Beam element theory . . . 83
4.3.3.2 Plate element theory . . . 84
4.3.3.3 Boundary Conditions with Discrete Elements . . . 87
4.3.3.4 Pre-stress Modeling and Geometric Stiffness Matrix . . . 89
4.3.4 Thermal Effects . . . 90
4.3.5 Coupling Between Elements of Different Theories . . . 92
4.4 Software Coupling and Data Transfer . . . 94

5 VALIDATIONS CASES . . . 98
5.1 Validation of the interpolation functions . . . 98
5.2 Modal Analysis with Mixed Structural Models . . . 103

5.3 Modal Analysis Considering Internal Pressure and Added Mass on a Pipe . . . 106
5.4 Fluid-structure interaction analysis in a plate under a high-temperature environment . . . 108

6 CONCLUSION . . . 120

BIBLIOGRAPHY . . . 122

A INTRODUCING CODE_ASTER, SALOME, AND SALOME-MECA 131
A.1 Code_Aster . . . 131
A.1.1 Main Features . . . 131
A.1.2 Modeling and Solution Types . . . 132
A.1.3 Computational Capabilities . . . 132
A.1.4 Usage Environment . . . 132
A.1.5 Study Reference . . . 132
A.2 Salome . . . 132
A.2.1 Main Functionalities . . . 133
A.3 Salome-Meca . . . 133
A.3.1 Main Functionalities . . . 133
A.4 Final Remarks . . . 134

1 Introduction

Fluid–structure interaction (FSI) is a physical phenomenon present in a wide variety of scientific and industrial contexts, characterized by the mutual influence between a fluid and a structure. This phenomenon occurs when the force exerted by the fluid on a structure alters its shape or movement, which in turn modifies the flow of the surrounding fluid. The complexity of this phenomenon arises from the dynamic coupling between the two domains, which requires sophisticated mathematical formulations for its description and analysis (BAZILEVS; TAKIZAWA; TEZDUYAR, 2013). Collar (1946) proposes a representation of the fluid–structure interaction related to aircraft in a diagram that connects the fluid and structural domains with the predominant forces of the phenomenon. An analogy of this diagram for the major fields of fluid–structure interaction can be seen in Fig. 1, where the phenomenon is divided into three main interdependent components: fluid mechanics, which deals with fluid forces and their dynamics; structural mechanics, which governs the behavior of deformable structures; and dynamic coupling, which captures the effects of the interaction between the two systems (DOWELL, 2021). As in Collar's Diagram, this representation allows one to visualize how different aspects of FSI interrelate and give rise to phenomena such as flow-induced vibration, flutter, fluid–elastic and structural instability caused by oscillatory forces.

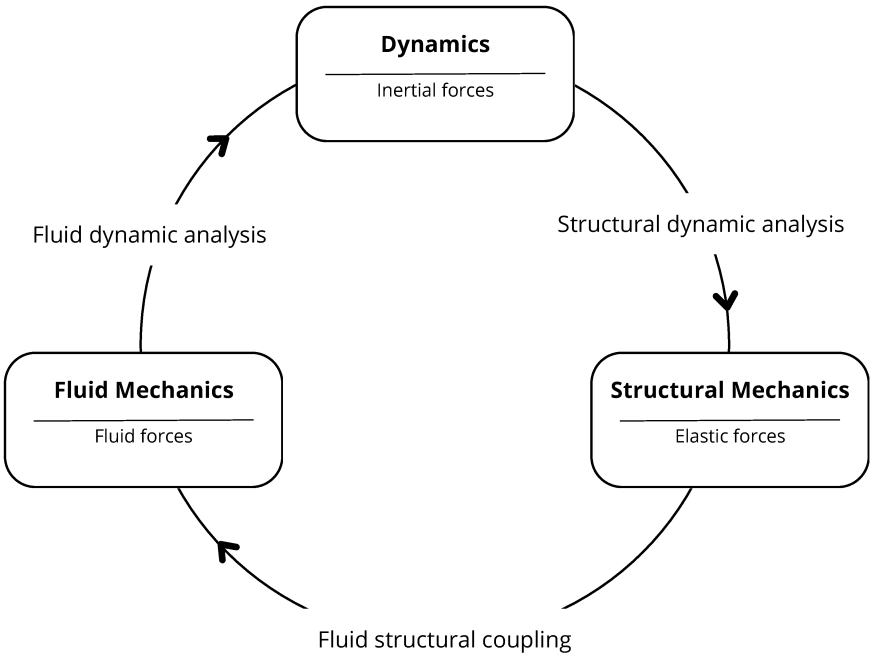


Figure 1 – Scheme of fluid–structure interaction (FSI), highlighting the three main domains (Fluid Mechanics, Structural Mechanics, and Dynamics) and their interactions.

This type of interaction is present in various applications, such as the aerodynamic

behavior of aircraft, the performance of wind turbines, the biomechanics of blood flow, and naval engineering. The complexity of this interaction is due to the fact that, in most cases, the coupling between the two domains is highly nonlinear and time-dependent, making the use of advanced methods for its analysis and prediction essential (BAZILEVS; TAKIZAWA; TEZDUYAR, 2013).

In view of this complexity, computational modeling has emerged as an indispensable tool for understanding and predicting the effects of fluid–structure interaction. The advancement of numerical simulation techniques has allowed for the partial or complete replacement of expensive physical experiments, as well as enabling detailed analyses and parameterizations of systems on an industrial scale. Among the main methods used for FSI modeling are the Arbitrary Lagrangian–Eulerian (ALE) method, stabilized finite element methods, and multi-scale simulations, which allow the capture of highly dynamic phenomena sensitive to parameter variations. These methods are widely employed to study the response of materials and structures under different flow regimes and enable the precise optimization of the performance of mechanical and biological systems (BAZILEVS; TAKIZAWA; TEZDUYAR, 2013).

The need for computational modeling of fluid–structure interaction problems is widely recognized in engineering and industry. Traditionally, the design of mechanical and structural systems relied heavily on physical testing and experimental prototypes, which limited innovation and increased technological development costs. However, with the increase in computational capacity and the refinement of numerical algorithms, computational simulation has become established as the third pillar of scientific knowledge, complementing traditional theory and experimentation. Oberkampf (2010) emphasize that computational modeling plays a fundamental role in decision-making, as it allows the testing of a wide range of operating conditions and the prediction of potential failures, thereby reducing costs and risks in engineering projects.

The impact of computational modeling in industry has become even more evident with the advent of Industry 4.0 and the digitization of production processes. The integration of computational models with sensors and real-time data analysis has enabled the development of technologies such as digital twins, which are highly accurate virtual representations of physical systems. These innovations not only allow for the simulation of processes even before a product is manufactured, but also enable the continuous optimization of industrial operations, increasing efficiency and reducing waste. Moreover, the use of predictive simulations in manufacturing, the aerospace sector, and renewable energy production has led to significant improvements in the reliability and safety of complex systems (LEE BEHNAM BAGHERI, 2018).

Despite these advances, computational modeling of fluid–structure interaction still faces significant challenges. The accuracy of the models depends on proper mathematical formulation, parameter calibration, and validation against experimental data. Numerical

errors and uncertainties inherent in the discretization process can compromise the reliability of simulations, making the application of rigorous verification and validation (V&V) methods essential. Oberkampf (2010) emphasize that without a careful assessment of model accuracy, there is a risk of generating imprecise predictions, which can compromise not only the effectiveness of projects but also the safety of critical systems such as aircraft and civil infrastructures.

Flow-induced vibration on cylinders is a phenomenon of great relevance in engineering, especially in the design of industrial equipment where fluid–structure interaction can directly impact safety and operational efficiency. Despite advances in understanding this phenomenon, its application in design is still partially limited. This is because, in real applications, different excitation mechanisms may act simultaneously, making it essential to separate and study each one in isolation to understand the structural response to these oscillatory forces. Thus, detailed investigations of the excitation mechanisms are fundamental for the improvement of analysis methodologies and computational modeling of this type of interaction (NEUMEISTER; BARBIERI, 2022).

Computational modeling has been a crucial tool for advancing the understanding and prediction of flow-induced vibrations on cylinders. By combining computational fluid dynamics (CFD) techniques with structural dynamics, it is possible to simulate and analyze different scenarios of fluid–structure interaction. However, challenges persist, particularly regarding the high computational cost of three-dimensional simulations and the need for rigorous experimental validation to ensure model accuracy. The synchronization of vortex shedding with the structural response represents one of the main challenges for numerical modeling, as it requires advanced techniques to capture the evolution of fluid forces in real time and their consequences on structural behavior (NEUMEISTER; BARBIERI, 2022).

1.1 Contextualization and Background

The present work is related to an ongoing scientific research project that focuses on the mathematical modeling and numerical simulation of turbulent flows with fluid-structure interaction. Developed by the Fluid Mechanics Laboratory (MFLab) and the “Prof. José Eduardo Tannus Reis” Structural Mechanics Laboratory (LMEst) of the School of Mechanical Engineering at the University of Uberlandia (UFU) in partnership with Petrobras, the project explores fluid-structure interaction in boiler tube bundles, focusing on developing a computational tool.

In summary, the study examines the interaction between the flow and the tube bundle of a recuperative boiler, aiming to identify the factors that cause vibrations and to propose strategies for mitigating vortex-induced vibrations. The industrial case is presented in the Chapter 3, Methodology — a decision made because the relevant fluid-structure interaction concepts are introduced in the Chapter 2, Literature Review. This structure ensures that,

by the time the reader is familiar with the industrial case, they already understand the underlying concepts as well as the impact and challenges of the computational study of fluid-structure interaction.

Due to concerns regarding intellectual property and industrial secrecy, the data related to the project will be presented in an approximate manner, using references proportional to the orders of magnitude involved. This approach aims to balance the preservation of confidential information with the need to convey the relevance and magnitude of the industrial case in question.

1.2 Objectives

The objective of this study is to develop the necessary numerical and computational tools to enable the multiphysics simulation software MFSim to perform fluid-structure interaction simulations of the heat recovery boiler. This will allow the study to examine the interaction between the flow and the tube bundle of a recuperative boiler using finite element models imported from external finite element software.

Based on this, the specific objectives are:

- Determine the coupling approach between MFSim and finite element software to allow the import of structural models;
- Implement an algorithm to associate the structural mesh with the mesh used in the immersed boundary method;
- Develop numerical and computational tools that enable fluid-structure interaction analysis using external finite element models;
- Validate the tools developed in this study.

1.3 Master thesis structure

This work is divided into six chapters and one appendix, each addressing specific aspects related to the overall objective of the study.

Chapter 2 presents the literature review on the phenomenon of fluid-structure interaction and the dynamics and mechanisms governing it.

Chapter 3 covers the mathematical and numerical foundations of this study, discussing the governing equations of fluid-structure interaction from a numerical and computational perspective.

Chapter 4 focuses on the numerical and computational modeling developed in this study. In this chapter, the reader will find the tools that were developed and implemented in the MFSim software.

Chapter 5 presents the results that validate the tools implemented in MFSim.
Chapter 6 provides the conclusions and final considerations of this study.
Appendix A details the Salome-Meca software and related applications.

2 Literature Review

The main objective of this chapter is to consolidate the general theoretical foundations that support the development of this work. To achieve this, it covers the concepts of fluid-structure interaction, the main theme of this study, as well as the finite element method and finite volume method, providing the basis for the applications that will be discussed in Chapter 3 and 4. Finally, the MFSim and Salome-Meca software are presented, detailing their functionalities and previous works conducted using these tools. This chapter is grounded in books, articles, and software documentation to ensure the theoretical basis.

2.1 Fluid-structure Iteration

Fluid-Structure Interaction (FSI) is a multi-physics problem that deals with the mutual interaction between a fluid flow and a deformable or movable structure. In FSI, the fluid forces cause the structure to deform or oscillate, and these structural changes, in turn, modify the fluid flow. The inherently nonlinear and time-dependent nature of FSI makes analytical solutions impractical in most cases, which has driven significant advances in computational methods. These methods are now capable of handling complex three-dimensional geometries and fully coupled simulations (BAZILEVS; TAKIZAWA; TEZDUYAR, 2013; PAïDOUSSIS, 2014).

2.1.1 Areas of Application

FSI is present in a wide range of engineering and scientific applications. Some key areas include:

- **Aerospace Engineering:** The analysis of aircraft wings and wind-turbine blades, where flow-induced deformations play a critical role in performance Bazilevs, Takizawa e Tezduyar (2013);
- **Biomechanics:** The study of blood flow through the heart and the operation of bioprosthetic aortic valves, where fluid forces and structural motion are tightly coupled (COSTA et al., 2025);
- **Nuclear Engineering:** The design and optimization of reactor components that must endure dynamic fluid forces, with attention to mitigating pipeline vibrations (PAïDOUSSIS, 2014);
- **Civil and Marine Engineering:** The analysis of bridges, offshore structures, pipelines, and heat exchangers, where FSI effects such as vortex-induced vibra-

tions (VIV) are critical for structural integrity (NEUMEISTER; BARBIERI, 2022; WILLIAMSON; GOVARDHAN, 2004).

A characteristic of FSI is its multidisciplinary nature, as evidenced by the works of several authors. For instance, Belytschko, Lin e Tsay (1982) define FSI as a coupled phenomenon in which the structural response is influenced by the surrounding fluid's acoustic modes, pressure variations, and inertial forces; they emphasize the importance of accounting for rapid pressure waves in scenarios such as waterhammer events. Similarly, Hamida (2025) describe how FSI plays a key role in enhancing heat transfer and inducing structural deformations in enclosures with flexible components, noting that the interplay between convective fluid flow and structural dynamics governs both the thermal performance and mechanical stresses of the system. Furthermore, Moschopoulos, Dimakopoulos e Tsamopoulos (2025) extend the FSI framework to cover multiphase interactions, where both fluid-fluid and fluid-structure couplings coexist, thereby necessitating numerical techniques that ensure stress and velocity continuity at dynamic interfaces. In marine applications, Kim e Shin (2025) investigate the impact of hydrodynamic forces on composite propellers, demonstrating that structural deformations can affect propulsion efficiency and vibrational characteristics. In the field of turbomachinery, Li et al. (2023) discuss flutter-induced instabilities in compressor blades caused by unsteady aerodynamic forces, which may lead to structural failure if not properly controlled. Finally, Seyed-Aghazadeh, Edraki e Modarres-Sadeghi (2019) analyze vortex-induced vibrations in submerged flexible structures, highlighting how these oscillatory motions can significantly impact fatigue life and overall stability.

2.1.2 Approaches to Modeling FSI

The computational modeling of FSI involves several strategies that account for the complex interactions between the fluid and structural domains. The main approaches include:

2.1.2.1 Monolithic Approach

In the monolithic approach, the governing equations for both the fluid and the structure are solved simultaneously as one coupled system. This method provides strong coupling and can accurately capture the interaction dynamics, but it generally demands high computational resources (BUNGARTZ MIRIAM MEHL, 2010).

2.1.2.2 Partitioned Approach

The partitioned approach treats the fluid and structural domains separately using specialized solvers for each. The interface conditions are enforced iteratively to ensure

compatibility between the two domains. While this method allows leveraging existing solvers, it may suffer from numerical instabilities when the coupling is strong (HOU JIN WANG, 2012).

2.1.2.3 One-Way and Two-Way Coupling

Fluid-Structure Interaction (FSI) problems can be classified based on the level of interaction between the fluid and the solid domains. This classification defines the extent to which the deformation of the structure influences the surrounding flow and vice versa.

- **One-Way Coupling:** In one-way coupling, the fluid forces act on the structure, but the structural response does not significantly alter the fluid flow field. This approach is computationally efficient and is widely used in problems where the structural deformations are negligible compared to the flow dynamics. For example, Blevins (1990) discusses cases where vortex-induced vibrations (VIV) in bluff bodies can be approximated using a one-way coupled model, assuming that the flow field remains largely unchanged by the motion of the structure. Similarly, one-way coupling has been extensively used in biomechanics for modeling blood flow through arteries, where arterial wall deformations are minimal compared to the fluid forces exerted by the pulsatile blood flow (SILVA et al., 2023);
- **Two-Way Coupling:** In contrast, two-way coupling considers the mutual interaction between the fluid and the structure. The deformation of the structure significantly influences the fluid flow, which in turn further affects the structure. This bidirectional feedback loop is crucial in scenarios where the structural motion alters aerodynamic forces, such as in aerospace applications where flutter in aircraft wings can lead to catastrophic failures (WILLIAMSON; GOVARDHAN, 2004).

A key advantage of the one-way FSI approach is its lower computational cost. Studies have shown that the computational time required for a one-way simulation can be significantly lower than that of a two-way simulation. For instance, in the context of tall buildings subjected to wind loads, Wijesooriya et al. (2021) demonstrated that an uncoupled one-way FSI method required approximately 74 simulation hours to obtain structural responses for 18 configurations, while a comparable two-way FSI simulation took over 4800 simulation hours to evaluate just six configurations.

However, despite its computational efficiency, one-way FSI has limitations. One significant drawback is its inability to capture aerodynamic damping and structural feedback mechanisms that can influence flow patterns. Wijesooriya et al. (2021) emphasize that two-way coupling is necessary for capturing transient fluid-structure interactions where the structural response significantly affects flow behavior, such as in the case of vortex-induced resonance (VIR) in super-tall buildings.

In biomedical applications, Silva et al. (2023) conducted a comparative study on aneurysm modeling and found that one-way FSI approaches tend to underestimate wall stress values throughout the cardiac cycle. This discrepancy arises because the arterial wall motion is not accounted for in the fluid solution, leading to an overestimation of wall shear stress when compared to two-way coupled simulations.

Moreover, in turbomachinery applications, Wijesooriya et al. (2021) highlight that one-way FSI can fail to predict fluidelastic instabilities accurately. When unsteady aerodynamic forces interact with flexible compressor blades, a two-way approach is required to fully capture the onset of flutter and self-excited oscillations that could lead to blade failure.

Thus, the selection between one-way and two-way FSI approaches depends on the nature of the problem. One-way FSI is suitable for scenarios where structural deformations have minimal impact on the flow field, making it computationally efficient for preliminary design and parametric studies. However, for cases involving significant fluid-structure coupling, where structural motions induce flow alterations that must be considered, two-way FSI is necessary despite its higher computational cost.

Additional computational strategies involve the choice of mesh handling. Some methods rely on conforming meshes that require re-meshing as the structure deforms, while others use non-conforming (immersed) techniques that embed the structure within a fixed fluid mesh and incorporate interaction forces accordingly (BUNGARTZ MIRIAM MEHL, 2010).

2.1.3 Dynamic Analysis of FSI

Dynamic phenomena are critical to understanding FSI behavior. Key dynamic effects include:

- **Vortex-Induced Vibrations (VIV):** Periodic shedding of vortices from a bluff body generates oscillatory forces that can lead to significant structural vibrations Williamson e Govardhan (2004), Seyed-Aghazadeh, Edraki e Modarres-Sadeghi (2019);
- **Flutter:** A dynamic instability seen in slender structures, such as aircraft wings, where aerodynamic forces induce self-excited oscillations Païdoussis (2014);
- **Fluidelastic Instability:** Occurs when the energy transferred from the fluid to the structure exceeds the damping capacity, resulting in sustained and potentially damaging oscillations Blevins (1990).

Studies on nonlinear behaviors, including limit cycles and chaotic oscillations, have deepened the understanding of these dynamics and contributed to solutions aimed at mitigating issues like pipeline vibrations and optimizing reactor component designs Païdoussis (2014).

2.1.4 Finite Element Method

The finite element method (FEM) is a powerful tool for solving fluid-structure interaction (FSI) problems, offering a significant advantage over traditional analytical models. Unlike analytical approaches, which often rely on simplifying assumptions regarding structural behavior, FEM allows for a more detailed and realistic representation of the structure. By discretizing complex geometries into smaller, interconnected elements, FEM makes it possible to accurately capture the interaction between fluids and structures, even in cases involving intricate shapes and heterogeneous materials. This approach not only enhances the accuracy of the structural response prediction but also enables the consideration of nonlinear effects, material anisotropy, and dynamic coupling, which are often neglected in simplified analytical models. As a result, FEM has become an essential tool for analyzing FSI phenomena in diverse engineering applications, from heat exchanger tube arrays to aerospace and biomedical systems (BAZILEVS; TAKIZAWA; TEZDUYAR, 2013; KALITA; RAO, 2023).

Bazilevs, Takizawa e Tezduyar (2013) explain how FEM is especially useful for FSI because it can handle challenges like large movements of structures and the complex forces at the interface between fluids and solids. Methods like the Arbitrary Lagrangian-Eulerian (ALE) approach make it easier to simulate situations where boundaries move, such as the flow around flexible structures. Space-time techniques further improve FEM by allowing it to solve problems that change over time.

The finite element method is also a widely used computational approach in engineering that provides a systematic way to analyze and solve problems involving complex geometries and material behaviors (BELYTSCHKO; LIU; MORAN, 2000). According to Bathe (2016), FEM begins with the idealization of a physical problem into a mathematical model, typically governed by differential equations. These equations are then discretized into smaller, simpler elements, allowing the numerical solution of the system under various boundary conditions and constraints.

Bathe (2016) emphasizes that FEM achieves its best performance when a balance is maintained between the complexity of the mathematical model and the computational cost. Its reliability stems from its ability to approximate physical phenomena within acceptable error margins, provided the model is carefully defined. Furthermore, Bathe classifies FEM procedures based on their application domains, such as structural analysis, heat transfer, and fluid-structure interactions, showcasing its versatility and wide applicability in engineering (BAZILEVS; TAKIZAWA; TEZDUYAR, 2013; BATHE, 2016).

2.1.5 Research on FSI at MFLab

Within the scope of this work, which focuses on finite element modeling for fluid-structure interaction, previous studies carried out with MFSim stand out, whose scope is similar to that of the present study. One of these is by Neto (2016), which investigates fluid-structure interaction in submarine pipelines subjected to external flows, emphasizing the influence of seabed proximity on flow-induced vibrations. Through the immersed boundary method, the author numerically coupled the structural modeling, based on Timoshenko beam theory, to the fluid dynamics of incompressible flows. The simulations analyzed different distances between the pipeline and the seabed, highlighting how small variations in the structure's position affect the fluid dynamic forces and the resulting vibrations.

The study by Souza (2020) addresses the development of numerical methodologies for fluid-structure interaction. In the flow modeling, the LES (Large Eddy Simulation) approach was employed with finite-volume discretization, suitable for the simulation of incompressible and turbulent flows. The solid structure, in turn, was modeled using the finite element method based on the Mindlin-Reissner plate theory. The coupling between the fluid and structural domains was achieved through the immersed boundary method, enabling an integrated fluid-structure interaction. The methodology was tested and validated in several scenarios, demonstrating its generality and robustness for industrial applications.

The work by Ávila (2021) presents the mathematical and computational modeling of fluid-structure interaction in three-dimensional curvilinear shell elements. The author developed and implemented a computational model in the MFSim code, enabling the analysis of coupled multiphysics phenomena between fluids and complex structures. The structural modeling was performed using the finite element method, while the fluid was treated with a Eulerian approach based on finite volumes. The coupling between the domains was achieved through the immersed boundary methodology, allowing for an integrated analysis of structural deformations and incompressible flows.

The work by Morales (2023) presents a mathematical and computational model for fluid-structure interaction, utilizing hexahedral solid elements for industrial applications. The coupling between the domains was achieved using the immersed boundary method. Both Souza (2020), Ávila (2021), and Morales (2023) conducted fluid structure analyses of a pipeline in the Fluid Catalytic Cracking Unit (UFCC) at Petrobras. Additionally, Souza (2020) and Morales (2023) performed analyses considering compressible flow and variable physical properties.

2.2 Dynamics and Mechanisms of Fluid-Structure Interaction

Fluid-Structure Interaction (FSI) plays a fundamental role in the behavior of industrial systems composed of cylindrical structures, such as heat exchangers, risers, boilers, and cooling systems. These structures are typically immersed in a fluid—either gas or liquid—and their interaction with the surrounding flow results in periodic forces that can induce unwanted vibrations and even lead to structural failure (NEUMEISTER; BARBIERI, 2022).

The dynamical response of structures due to FSI varies depending on several factors, including the spacing between cylinders, their orientation relative to the flow, their geometric dimensions, and the flow velocity. In heat exchanger tube banks, for example, the presence of multiple cylinders in proximity alters the flow pattern, making the interaction more complex compared to an isolated cylinder scenario. These factors influence vortex shedding, wake formation, and oscillatory forces, which must be carefully analyzed in engineering applications to ensure structural integrity and performance (PAïDOUSSIS; PRICE; LANGRE, 2011; BLEVINS, 1990).

This section presents a review of key FSI mechanisms that are relevant to cross-flow-induced vibrations in tube arrays, with particular emphasis on industrial applications involving heat exchanger tube banks. The discussion will highlight distinct response mechanisms observed in bluff-body flows, contrasting them with the industrial case studied in this work. The following subsections delve into the primary fluid-structure interaction phenomena, including vortex shedding, galloping, and wake-induced instabilities.

2.2.1 Vortex Shedding

Vortex shedding is a key phenomenon in fluid-structure interaction (FSI) and plays a fundamental role in generating flow-induced vibrations. This mechanism occurs when the boundary layer separates from the surface of a bluff body, forming alternating vortices in the wake. These vortices induce periodic variations in pressure, which can lead to structural oscillations. The phenomenon is particularly relevant in the context of cylindrical structures, where it can trigger significant vibrations that affect the integrity and performance of engineering systems.

According to Blevins (1990) and Meneghini (2002), vortex-induced vibrations (VIV) arise when the predominant excitation force in an FSI system is linked to the periodic shedding of vortices. This is particularly critical in heat exchanger tube arrays, marine risers, and bridge cables, where fluid forces interact with the structural response. The formation of a vortex street, commonly referred to as the von Kármán vortex street, is a well-documented effect of this process.

The frequency of vortex shedding is primarily governed by the Strouhal number, defined as:

$$St = \frac{f_s D}{U}, \tag{2.1}$$

where f_s is the vortex shedding frequency, D is the characteristic dimension (such as the diameter of a cylinder), and U is the free-stream velocity. Studies by Nakamura et al. (2013) have shown that in cross-flow conditions, transverse forces exhibit a dominant frequency equal to the vortex shedding frequency, as the vortices are shed alternately from each side of the body. In contrast, inline forces oscillate at twice this frequency because they are influenced by the magnitude of the pressure fluctuations rather than their direction. Since each vortex shedding event — whether from the top or bottom — induces a local increase in drag, the inline force experiences two peaks per shedding cycle, effectively doubling its frequency compared to the transverse force oscillations.

Synchronization, or lock-in, is a critical aspect of vortex shedding. It occurs when the vortex shedding frequency matches the natural frequency of the structure, leading to an increase in oscillation amplitude. As demonstrated by Pettigrew et al. (1998), when synchronization is established, the structure's motion influences vortex shedding, reinforcing large-amplitude oscillations. This self-sustaining loop can be highly detrimental to the mechanical integrity of components subjected to unsteady flow conditions.

Furthermore, Norberg (2003) noted that the oscillatory behavior of a structure subjected to vortex shedding depends on multiple factors, including mass ratio, damping characteristics, and Reynolds number. At low mass-damping ratios, the structure is more prone to high-amplitude oscillations, whereas at higher damping levels, the response is mitigated.

The impact of vortex shedding on heat exchanger tube arrays is particularly complex due to the interaction between multiple wake structures. Goyder (2002) analyzed how varying the tube spacing and arrangement affects shedding patterns, influencing aerodynamic loading on downstream structures. Similarly, Moretti (1993) demonstrated that in tube bundles, vortex shedding is governed not only by the Strouhal number but also by geometric and fluid dynamic parameters.

2.2.2 Wake Induced Vibration

Wake Induced Vibration (WIV) is a phenomenon in which the wake shed by an upstream body influences the dynamic response of a downstream structure. This effect is particularly relevant in arrangements involving multiple cylinders, such as tube arrays in heat exchangers, where the interaction between the wake and the adjacent cylinders can lead to complex vibratory behavior (NEUMEISTER; BARBIERI, 2022).

In general, wake-induced vibrations occur when the wake of an upstream body creates fluctuating pressure forces on a downstream structure, leading to oscillatory motion. Unlike vortex-induced vibrations (VIV), which are driven by alternating vortex shedding, WIV

is primarily a result of the instability in the wake flow and the interference with the downstream structure. The amplitude and frequency of WIV are influenced by several parameters, including the Reynolds number, the spacing between cylinders, and the orientation relative to the incoming flow (BLEVINS, 1990).

One of the key characteristics of WIV is that it can lead to sustained oscillations even when vortex shedding synchronization (lock-in) is not present. This makes WIV a significant concern in industrial applications, where tube arrays are subjected to cross-flow conditions. Studies have shown that depending on the flow velocity and the relative positioning of the cylinders, WIV can either amplify or suppress structural vibrations (GOYDER, 2002).

Bokaian e Geoola (1984) conducted experiments on pairs of cylinders with different transverse and longitudinal spacings, observing distinct vibratory behaviors. When the upstream cylinder was flexibly mounted while the downstream cylinder was fixed, they noted a strong dependence of WIV on the spacing between the cylinders. At small spacings ($T/D < 0.5$ and $L/D < 1.5$), galloping behavior was observed, where the vortex shedding from the upstream cylinder ceased, and the primary excitation mechanism became the asymmetry of the flow. This condition, known as wake galloping, can lead to large-amplitude oscillations (SUMER; FREDSOE, 2006; NEUMEISTER; BARBIERI, 2022).

In another scenario where the downstream cylinder was free to vibrate, the interaction with the wake of the fixed upstream cylinder led to a combination of vortex-induced vibrations and wake galloping. At low flow velocities, vortex shedding dominated the vibratory response, while at higher velocities, wake galloping took over, resulting in even greater vibration amplitudes (ASSI; BEARMAN; MENEGHINI, 2010; ASSI et al., 2010b).

Industrial applications involving WIV include heat exchanger tube bundles, offshore risers, and cooling systems, where the presence of closely spaced cylindrical structures can lead to unwanted vibrations. The mitigation of WIV often involves optimizing the spacing between tubes, introducing flow-interrupting devices, or altering the structural damping to suppress excessive oscillations (PETTIGREW et al., 1998).

Despite extensive research, WIV remains a complex and challenging phenomenon to predict accurately due to its dependence on multiple interacting factors. Further studies are required to refine numerical models and improve experimental techniques to better understand and mitigate its effects in engineering applications. Recent advancements, such as those proposed by Awadallah et al. (2025), demonstrate how refined numerical simulations and experimental validation can not only enhance predictability but also harness WIV for applications like energy harvesting, showcasing a shift from mitigation to optimization in engineering design. Additionally, the work by Song et al. (2024) contributes to this effort by developing and comparing multiple predictive models for WIV, incorporating instantaneous structural feedback and nonlinear self-excited forces. Their

study highlights the importance of experimental validation through wind tunnel testing and numerical simulations, improving the reliability of WIV predictions.

2.2.3 Turbulent buffeting

Turbulent buffeting is an excitation mechanism characterized by turbulence phenomena and is associated with the generation of random pressure fluctuations around the surface of structures exposed to fluid flow. These fluctuations induce vibrations as a result of the broad spectrum of velocity components in the flow. Excitation can originate from local turbulence in the surrounding fluid, known as near-field vibration, or from downstream and upstream components such as surface roughness, bends, nozzles, and other pipe accessories, classified as far-field vibration (PETTIGREW et al., 1998).

This excitation mechanism can be classified into single-phase and multiphase buffeting. In single-phase buffeting, the tube agitation results solely from turbulence in the surrounding fluid, leading to random excitations. In multiphase systems, changes in the fluid density also contribute to the excitation forces on the cylinders (GOYDER, 2002).

Flow turbulence can either amplify or mitigate vibration amplitudes, since high-velocity fluctuations can broaden the range of excitation frequencies. Turbulence alters the pressure distribution on the cylinder surface and influences wake galloping, stabilizing, or destabilizing the structure depending on the governing parameters and geometric configurations (PAïDOUSSIS; PRICE; LANGRE, 2011).

The characterization of turbulent excitation in cylinders is conducted through power spectrum analysis of the forces induced by flow turbulence. The drag and lift forces depend on several parameters, such as aspect ratio, cylinder spacing, and flow characteristics, including Reynolds number and turbulence intensity. However, available experimental data remain limited, particularly for high Reynolds numbers, which restricts more in-depth analyses of this mechanism (BLEVINS, 1990).

To estimate the general trend of flow turbulence effects, turbulence intensity is commonly used. Nakamura et al. (2013) demonstrated that a cylinder subjected to forces outside the synchronization region is inevitably excited due to the wide frequency range of the flow turbulence components. When the natural frequency of the cylinder is significantly different from the dominant vortex shedding frequency, the structure is excited by frequency components that are closer to its natural frequency.

For a single, freely vibrating cylinder, the effect of turbulence intensity is most pronounced in the synchronization region. Studies by So et al. (2008) indicate that increased turbulence intensity results in greater energy transfer to the cylinder, reaching a peak in the synchronization zone before rapidly declining outside this region.

In cylinder pairs, Wang, Wong e Zhou (2019) observed that vibration amplitudes increased with turbulence intensity due to the instability of shear layers and their interaction

with neighboring cylinders. The incident flow fluctuations resulted in larger vibration amplitudes, highlighting the critical role of turbulence in flow-induced vibrations.

To mitigate the impact of turbulent excitation on vibration, several strategies can be implemented, such as increasing the natural frequency of structures by reducing tube spacing, lowering the flow velocity, or installing baffles to restrict fluid movement in tube arrays (NAKAMURA et al., 2013).

Although turbulent excitation alone does not necessarily lead to amplification of oscillations, it frequently occurs in conjunction with vortex shedding or acoustic excitation, potentially leading to higher vibration amplitudes. As a result, this mechanism must be carefully considered in engineering designs to ensure the reliability and integrity of structures exposed to turbulent flow conditions (PETTIGREW et al., 1998).

Neto et al. (2019) investigates the interaction between the structure of a horizontal pipeline and the surrounding fluid flow, focusing on the proximity to the seabed, which affects the vibrations of the structure. These vibrations are generated by fluctuations in the forces induced by vortex shedding in the flow. Simulations for different distances between the pipeline and the seabed were performed using coupled numerical models to solve the fluid-structure interaction and employing an immersed boundary methodology.

2.2.4 Fluid-Elastic Instability

Fluid-elastic instability (FEI) is a self-excited vibration phenomenon that occurs when the energy transferred from the fluid to a structure exceeds the system's damping capacity, leading to large-amplitude oscillations. This mechanism is particularly critical in tube arrays found in heat exchangers, boilers, and steam generators, where high flow velocities can induce destructive vibrations (PETTIGREW et al., 1998). Unlike vortex-induced vibrations (VIV), which depend on external periodic forcing, fluid-elastic instability is governed by an internal feedback mechanism between forces and structural motion (GOYDER, 2002).

The onset of fluid-elastic instability is characterized by a velocity threshold, often referred to as the critical velocity, beyond which the system becomes dynamically unstable. According to Païdoussis, Price e Langre (2011), the instability occurs when the fluid forces acting on the structure remain in phase with the velocity of the oscillating body, continuously amplifying the motion. The governing parameters for predicting fluid-elastic instability include reduced velocity (U^*), the mass-damping parameter ($m^*\zeta$), and the Scruton number (Sc), which influence the stability threshold (PETTIGREW; TAYLOR, 2003).

Experimental and numerical studies have shown that the fluid-elastic instability is highly dependent on the geometric configuration of tube arrays. In aligned tube arrays, instability tends to develop at lower flow velocities compared to staggered configurations,

due to the stronger coupling between adjacent tubes (ZIADA; OENGÖREN, 1992). The mechanism of instability can be classified into two main types:

Stiffness-Controlled Mechanism: This mechanism arises when multiple tubes move coherently, leading to a coupled-mode instability. The forces exerted by the fluid act in phase with the tube displacement, further reinforcing the oscillation. This effect is particularly pronounced in tube arrays with low mass ratios and flexible supports (NEUMEISTER; BARBIERI, 2022).

Damping-Controlled Mechanism: In this case, the fluid forces are in phase with the tube velocity, resulting in energy extraction from the flow. This mechanism is dominant in configurations with higher mass ratios and increased damping (MORETTI, 1993).

Recent research has highlighted that spatially bent tube structures, as found in intermediate heat exchangers, exhibit fluid-elastic instability behavior that differs from conventional straight tube configurations. According to Wang et al. (2024a), numerical simulations using Computational Fluid Dynamics (CFD) and fluid-structure coupling models indicate that the critical velocity for instability in bent tube sections is lower than that of straight sections. This is due to increased turbulence and complex flow interactions in curved regions. Such findings underscore the importance of considering tube geometry when evaluating the stability of heat exchanger systems (WANG et al., 2024a).

The determination of the critical velocity for fluid-elastic instability is commonly performed using semi-empirical models. A widely accepted formulation is given by the quasi-static model:

$$U_{pc} = K \left(\frac{m\delta}{\rho D^2} \right)^n , \tag{2.2}$$

where U_{pc} is the critical flow velocity, m is the mass per unit length, δ is the logarithmic decrement of damping, and K and n are empirical coefficients that depend on the tube array configuration (PAïDOUSSIS; PRICE; LANGRE, 2011). The model suggests that increasing the natural frequency or structural damping can mitigate the onset of instability, thereby enhancing the operational limits of engineering systems.

Advanced machine learning techniques have been proposed to enhance the predictive capabilities of fluid-elastic instability models. Zhao et al. (2024) introduced a deep neural network (DNN) combined with a genetic algorithm (GA) to optimize tube array configurations for higher critical velocities. By training on extensive experimental datasets, this approach improves the efficiency of FEI predictions and enables safer, cost-effective design modifications in nuclear and thermal power applications (ZHAO et al., 2024).

Recent numerical simulations and experimental investigations have highlighted the role of turbulence intensity in modulating fluid-elastic instability. Studies by Wang, Wong e Zhou (2019) indicate that high turbulence levels can destabilize the flow-structure interaction, reducing the critical velocity and increasing the risk of instability. Conversely, certain

geometric modifications, such as optimizing tube spacing and adding flow baffles, have been proposed as effective strategies for suppressing fluid-elastic instability in industrial applications (ZIADA; OENGÖREN, 1992).

Despite significant advances in understanding fluid-elastic instability, challenges remain in accurately predicting its onset in complex flow conditions. The interplay between turbulence, structural nonlinearity, and multiphase effects introduces uncertainties that require further investigation.

3 Methodology

In this chapter, the foundation for the development of this work will be structured, focusing on the methods used to address the proposed industrial problem. First, the industrial case that serves as the motivation for this dissertation, proposed by Petrobras, will be described in detail, emphasizing the need for a computational approach to model the fluid-structure interaction phenomenon. Following that, the necessary numerical and computational developments will be outlined to enable the simulation using the Multiphysical Simulator Software (MFSim). Additionally, the chapter will present the steps required to integrate MFSim with external structural models from the finite element software Salome-Meca.

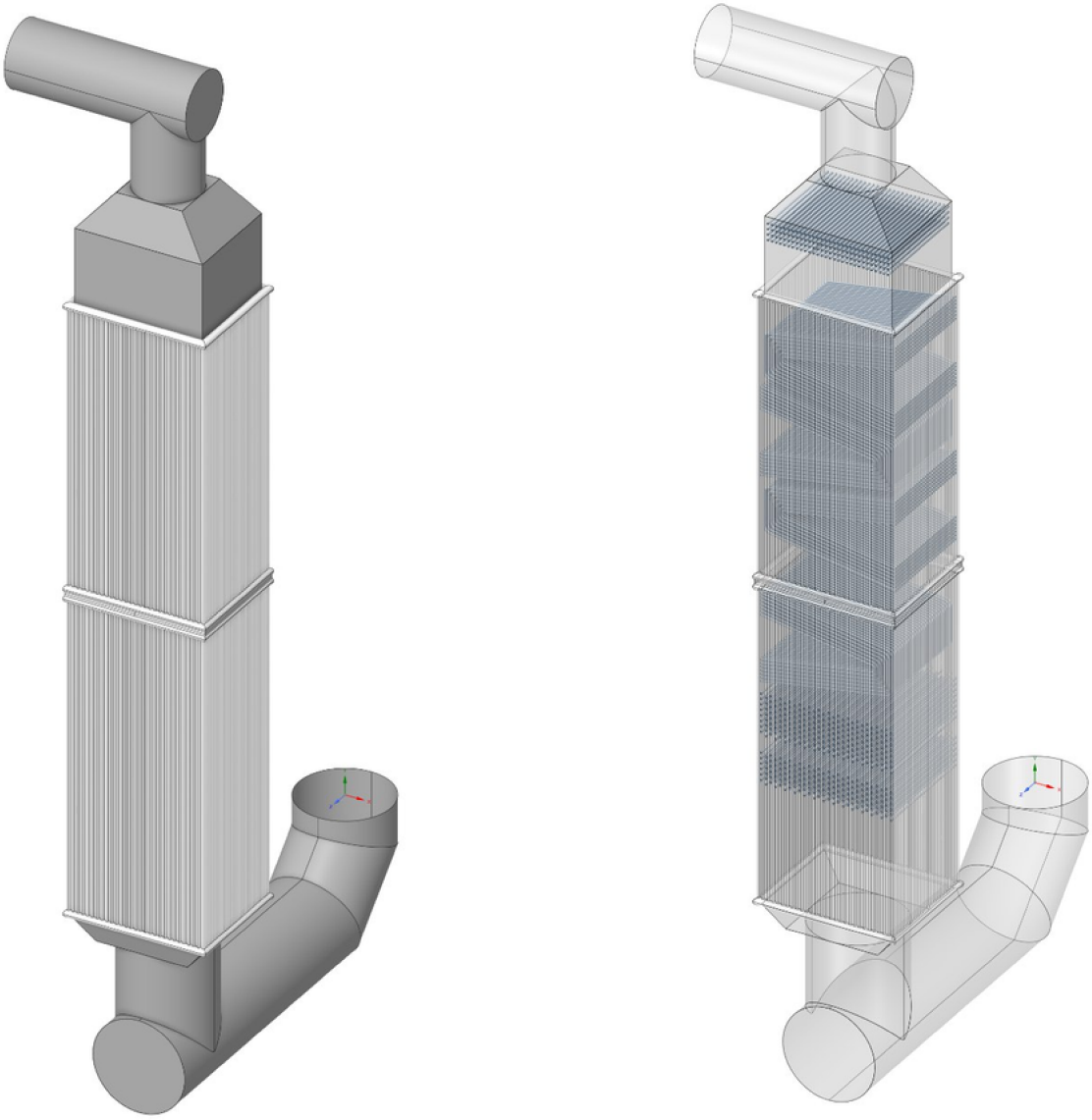
3.1 Description of the Industrial Case

The industrial case that motivates the present study is the investigation of vortex-induced vibrations in a Petrobras heat recovery boiler, a research project conducted by the laboratory. A heat recovery boiler, also known as a Heat Recovery Steam Generator (HRSG), is a type of boiler designed to capture and utilize waste heat from industrial processes, gas turbines, or other heat sources to generate steam. These boilers are commonly used in combined cycle power plants, cogeneration systems, and industrial facilities to improve energy efficiency by recovering heat that would otherwise be lost to the environment (GANAPATHY, 2003; RAYAPROLU, 2012).

The boiler can be seen in Fig. 2. In Fig. 2 (a), the external view of the boiler is presented through its CAD model. It is possible to observe the ducts through which the flow occurs, as well as the tubular walls of the assembly. In Fig. 2 (b), the internal tubes of the system are displayed, highlighting the quantity and similarity between the different tube assemblies.

This study focuses on the development of computational tools to support the fluid-structure interaction analysis of the system, considering the operating conditions of the heat recovery boiler. Currently, the finite volume model consists of approximately 100 million cells, while the Lagrangian mesh used in the immersed boundary method contains about 15 million Lagrangian points. The finite element model, on the other hand, includes around 7 million nodes and 8 million elements, resulting in a structural system with more than 40 million degrees of freedom.

Computational modeling has posed challenges related to technological development, computational cost, and storage requirements.



(a) Isometric external view (b) Isometric internal view

Figure 2 – Computer-Aided Design (CAD) model of the recovery boiler developed by MFLab in partnership with Petrobras.

3.2 Approach to the case study

Given the complexity of the industrial case, both in terms of computational cost and structural modeling, challenges arise that demand more advanced approaches than simple implementations. To more accurately represent the boiler's operation under real conditions, it is essential to employ finite element models that consider factors such as the pressure and mass of the fluid inside the pipes, thermal effects on the material, the self-weight of the structure, and consequently, the impact of these variables on the system's dynamic response.

The plate finite element libraries in MFSim offer only linear elements for quadrangular

meshes, restricting modeling to the exclusive use of QUAD4-type elements, which have a quadrangular shape and four nodes (SOUZA, 2020; MORALES et al., 2023). As a result, geometries must be entirely discretized using this type of element, making it unfeasible to use automatic mesh generation algorithms, as these algorithms typically employ triangular elements for better adaptation to the model's geometry.

Given this scenario, it becomes essential to develop or implement numerical computational methods that allow MFSim to work with structural models that include the effects of loading on the system. The objective is to enable the discretization of the same geometry using different types of finite elements, ensuring the selection of the most suitable element for each situation while guaranteeing their coupling without compromising the physical representation of the system.

Thus, this section presents the proposed approach for the industrial case, enabling the simulation of the fluid-structure interaction of the recovery boiler within MFSim. To achieve this, the MFSim and Salome-Meca software are introduced, discussing their functionalities and potential. Additionally, an overview of the information exchange process between the two platforms is provided.

3.2.1 MFSim: a multiphysics simulation software

MFSim is a multiphysics simulation software that has been continuously developed since 2007 by the Fluid Mechanics Laboratory of the Federal University of Uberlândia (UFU), in collaboration with Petrobras. Its architecture is based on a modular philosophy, allowing different physical phenomena to be represented independently through specialized modules. Each module contains dedicated computational numerical methods, ensuring flexibility in modeling complex systems.

In recent years, the MFSim code has been used for the three-dimensional simulation of various phenomena, as shown in Tab. 1.

3.2.2 Salome-Meca

Salome-Meca is an integrated environment for numerical simulation that combines the Salome platform with the Code_Aster finite element solver. Developed primarily by the Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA) and Électricité de France (EDF), this package provides a unified graphical interface for model construction, mesh generation, analysis execution, and results visualization (EDF, 2024; EDF; CEA, 2013).

By integrating preprocessing, numerical solution, and post-processing tools into a single environment, Salome-Meca streamlines the engineering workflow, enabling everything from geometric creation to in-depth analysis of structural, thermal, and dynamic phenomena.

Table 1 – MFSim Code Applications and References (MOTA, 2023; VASCONCELLOS, 2024).

Phenomenon/Application	**Reference(s)**
Gas-solid flows	(PIVELLO et al., 2014)
Turbulent flows	(DAMASCENO; VEDOVOTO; SILVEIRA-NETO, 2015)
Hybrid methods	(MELO, 2017)
Studies on gas-liquid flows	(BARBI et al., 2018)
Reactive flows	(DAMASCENO; SANTOS; VEDOVOTO, 2018)
Turbulence modeling approaches	(ELIAS, 2018)
Gas-solid flows	(SANTOS, 2019)
Further investigations of gas-liquid flows	(PINHEIRO et al., 2021)
Reactive flows	(CASTRO et al., 2021)
Multiphase flows	(SOUZA et al., 2022)
Corrosion processes in fluid dynamic systems	(MOTA, 2023)
Forensic applications (non-Newtonian viscosity models)	(VASCONCELLOS, 2024)
Computational modeling of turbulent flows in wind turbines	(MARTINI, 2021; STIVAL et al., 2022)

It stands out for its modularity, allowing the inclusion of external solvers and adapting to various industrial and academic needs.

The use of Salome-Meca facilitates interaction between the geometry (GEOM), meshing (SMESH), and post-processing (ParaViS) modules, while also supporting open formats such as MED, ensuring interoperability with other numerical tools (AUBRY, 2019).

Further details on Salome-Meca, as well as the Code_Aster and Salome software, can be found in Appendix A.

3.2.3 Coupling MFSim with Salome-Meca

The coupling between MFSim and Salome-Meca will be unidirectional. Salome-Meca will be used as a finite element library, with the entire model construction process—from mesh generation to finalization—being carried out within this platform, as shown in Fig. 3. After this stage, the structural model data will be exported for fluid-structure interaction analysis in MFSim.

Thus, it is necessary to define how the data will be extracted from Salome-Meca and imported into MFSim, as well as how this information will be processed within MFSim for fluid-structure interaction analysis. These aspects will be addressed in Chapter 4, which focuses on the development of the coupling. This discussion is included in that chapter because the equations governing the structural analysis of fluid-structure interaction are

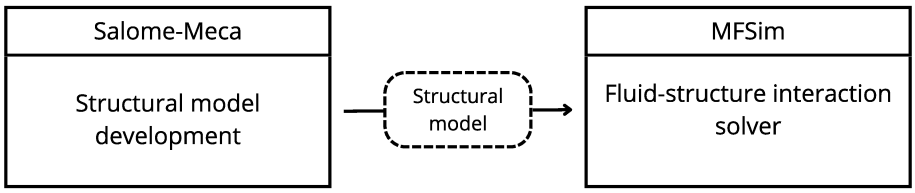


Figure 3 – Workflow of the unidirectional coupling process between Salome-Meca and MFSim.

presented in the Chapter 3, providing the reader with the necessary background information to better understand the coupling.

3.3 Mathematical Modeling

This section presents the set of equations related to the fluid-structure interaction phenomenon. The approach considers both the fluid and structural domains in a general manner, highlighting, even if only in an introductory way, some of the main physical variables involved in this analysis.

3.3.1 Fluid Domain

3.3.1.1 Mass Balance

The mass conservation equation ensures that the variation of mass within a control volume is balanced by the net mass flux across its boundaries. Written in divergent form and using index notation, the continuity equation is given by:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_i} = 0. \tag{3.1}$$

Here, ρ is the fluid density, t represents time, and u_i are the velocity components in the respective spatial directions x_i.

3.3.1.2 Linear Momentum Balance

The momentum conservation equation, also known as the Navier-Stokes equation, describes the relationship between the velocity field and the forces acting on the fluid. It accounts for inertial effects, pressure gradients, viscous stresses, and external forces:

$$\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_i u_j)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + f_i. \tag{3.2}$$

In this equation, p is the pressure field, τ_{ij} represents the viscous stress tensor, and f_i denotes external body forces acting on the fluid.

3.3.1.3 Thermal Energy Balance

The thermal energy equation governs the temperature distribution within the fluid. It takes into account the convective transport of energy, heat conduction, and work done by pressure forces:

$$\frac{\partial(\rho c_p T)}{\partial t} + \frac{\partial(\rho c_p T u_j)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(k\frac{\partial T}{\partial x_j}\right) + \Phi + \frac{\partial p}{\partial t} + u_j\frac{\partial p}{\partial x_j}. \tag{3.3}$$

In this formulation, T is the temperature, k represents the thermal conductivity, and c_p is the specific heat at constant pressure. The term Φ accounts for the viscous dissipation of energy within the fluid.

3.3.2 Structure Domain

3.3.2.1 Stress-Strain Relationship

In the elastic regime of materials, Hooke's Law states that the normal stress (σ) is directly proportional to the specific strain (ε), as long as the material does not exceed its proportional limit. This linear relationship is fundamental to solid mechanics and structural engineering (BEER et al., 2015).

Normal Stress

Normal stress is defined as the axial force P applied per unit area A:

$$\sigma = \frac{P}{A}. \tag{3.4}$$

In the International System of Units, the unit of σ is the Pascal (Pa), equivalent to $\mathrm{N/m^2}$.

Specific Strain

The specific strain is the relative change in the original length L_0 and can be expressed as:

$$\varepsilon = \frac{\Delta L}{L_0}, \quad \text{with} \quad \Delta L = L - L_0 \tag{3.5}$$

Experimental Origin

Robert Hooke, through experiments with various materials, observed that the elongation of a bar subjected to an axial load was proportional to the applied force. This experimental observation forms the basis for the modern formulation of Hooke's Law (RICHARDS, 2001).

Derivation of Hooke's Law in One Dimension

Based on the previous definitions, the one-dimensional formulation of Hooke's Law is:

$$\sigma = E \cdot \varepsilon \tag{3.6}$$

where E is the elastic modulus (or Young's modulus), which characterizes the stiffness of the material.

To illustrate this relationship, consider an infinitesimal element of a bar subjected to an axial load. For a small segment of length dx, the differential strain is defined by:

$$d\varepsilon = \frac{du}{dx}, \tag{3.7}$$

where du represents the infinitesimal change in displacement along the axis of the bar. Assuming proportionality between stress and strain for small displacements, we have:

$$d\sigma = E \cdot d\varepsilon. \tag{3.8}$$

Integrating along the length reaffirms the relationship:

$$\sigma = E \cdot \varepsilon. \tag{3.9}$$

Generalization for Three-Dimensional Materials

To describe the elastic behavior of a material in three dimensions, a tensorial formulation is used, which takes into account the interrelationships between strains and stresses in different directions (TIMOSHENKO; GOODIER, 1951; RICHARDS, 2001).

Strain Tensor

In a continuous medium, strains are described by the strain tensor ε_{ij}, defined by:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{3.10}$$

where u_i is the displacement in the i direction and x_j is the coordinate in the j direction.

Stress Tensor

The state of stress at a point is represented by the stress tensor σ_{ij}, which encompasses both the normal stresses (when $i = j$) and the shear stresses (when $i \neq j$).

Linear Constitutive Relationship

For linear and isotropic materials, the relationship between the stress and strain tensors is linear and can be written as:

$$\sigma_{ij} = C_{ijkl}\,\varepsilon_{kl} \tag{3.11}$$

where C_{ijkl} is the fourth-order constitutive tensor. Due to isotropy, this tensor takes the form:

$$C_{ijkl} = \lambda\,\delta_{ij}\delta_{kl} + \mu\,(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) \tag{3.12}$$

with δ_{ij} being the Kronecker delta (equal to 1 if $i = j$ and 0 otherwise) and λ and μ the Lamé coefficients.

Substituting the expression for C_{ijkl} into the constitutive relationship, we obtain the tensorial form of Hooke's Law (TIMOSHENKO; GOODIER, 1951; RICHARDS, 2001):

$$\sigma_{ij} = \lambda\,\delta_{ij}\,\varepsilon_{kk} + 2\mu\,\varepsilon_{ij} \tag{3.13}$$

where the term $\varepsilon_{kk} = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}$ represents the volumetric strain (the trace of the strain tensor).

Component Form Expressions

Normal Components:
In the principal directions (x, y, and z), the normal stresses are expressed as:

$$\begin{aligned}
\sigma_{xx} &= \lambda\,(\varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz}) + 2\mu\,\varepsilon_{xx} \\
\sigma_{yy} &= \lambda\,(\varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz}) + 2\mu\,\varepsilon_{yy} \\
\sigma_{zz} &= \lambda\,(\varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz}) + 2\mu\,\varepsilon_{zz}
\end{aligned} \tag{3.14}$$

Shear Components:
For the shear components, the relationship is:

$$\begin{aligned}
\sigma_{xy} &= 2\mu\,\varepsilon_{xy} \\
\sigma_{yz} &= 2\mu\,\varepsilon_{yz} \\
\sigma_{zx} &= 2\mu\,\varepsilon_{zx}
\end{aligned} \tag{3.15}$$

Relationship with Additional Elasticity Parameters

In addition to Young's modulus (E), the elasticity of materials is characterized by other fundamental parameters:

- **Poisson's Ratio (ν)**: Relates lateral strain to axial strain.

- **Shear Modulus (G)**: Relates to the material's resistance to shear and, for isotropic materials, is given by:

$$G = \frac{E}{2(1+\nu)}. \tag{3.16}$$

The Lamé coefficients can be related to E and ν by the expressions:

$$\lambda = \frac{E\,\nu}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)}. \tag{3.17}$$

Superposition Principle and Limitations

In the linear elastic regime, the superposition principle is valid: the effects of separately applied forces can be summed, as long as the resulting stress remains within the elastic limit of the material. However, it is important to note that Hooke's Law only applies as long as the material behaves linearly. When the stress exceeds the proportional limit, irreversible plastic deformation occurs, and the relationship $\sigma = E \cdot \varepsilon$ is no longer valid. Furthermore, in some materials the response may depend on time and temperature, characterizing viscoelastic behavior (RICHARDS, 2001; HASHIGUCHI, 2020).

3.3.2.2 Equation of Motion

The equation of motion for a mechanical system can be obtained using the Lagrange formulation. This approach allows one to model systems with multiple degrees of freedom by incorporating the effects of kinetic energy, potential energy, and external forces (SHABANA, 1997; INMAN, 2014).

Consider a system with n degrees of freedom, described by the generalized coordinates $q_1, q_2, \ldots, q_n$. These coordinates are chosen such that they completely describe the configuration of the system.

The kinetic energy of the system can be expressed as:

$$T = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} m_{ij}\dot{q}_i\dot{q}_j, \tag{3.18}$$

or, in matrix notation:

$$T = \frac{1}{2}\dot{\mathbf{q}}^T\mathbf{M}\dot{\mathbf{q}}, \tag{3.19}$$

where $\mathbf{M}$ represents the mass matrix of the system, which may depend on the coordinates q_i.

The potential energy associated with conservative forces, such as gravity and elasticity, is given by:

$$U = U(q_1, q_2, \ldots, q_n). \tag{3.20}$$

The Lagrangian of the system is defined as:

$$L = T - U. \tag{3.21}$$

The Lagrange equation is given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i, \tag{3.22}$$

where Q_i represents the external generalized forces acting on the system.

First, we differentiate the kinetic energy term:

$$\frac{\partial L}{\partial \dot{q}_i} = \sum_j M_{ij}\dot{q}_j. \tag{3.23}$$

Taking the time derivative gives:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) = \sum_j M_{ij}\ddot{q}_j + \sum_{j,k} \frac{\partial M_{ij}}{\partial q_k}\dot{q}_k\dot{q}_j. \tag{3.24}$$

The derivative of the potential energy term is:

$$\frac{\partial L}{\partial q_i} = -\frac{\partial U}{\partial q_i}. \tag{3.25}$$

Substituting these expressions into the Lagrange equation, we obtain the equation of motion:

$$\sum_j M_{ij}\ddot{q}_j + \sum_{j,k} \frac{\partial M_{ij}}{\partial q_k}\dot{q}_k\dot{q}_j - \frac{\partial U}{\partial q_i} = Q_i, \tag{3.26}$$

or, in matrix notation:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \frac{\partial U}{\partial \mathbf{q}} = \mathbf{Q}, \tag{3.27}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ contains the fictitious acceleration terms, $\frac{\partial U}{\partial \mathbf{q}}$ represents the restoring forces, and $\mathbf{Q}$ denotes the external generalized forces (INMAN, 2014; MEIROVITCH, 2001).

If $\mathbf{M}$ is constant and U represents a linear elastic system, then we have:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{Q}, \tag{3.28}$$

where $\mathbf{K}$ is the stiffness matrix of the system, defined as:

$$\mathbf{K} = \frac{\partial^2 U}{\partial \mathbf{q}^2}. \tag{3.29}$$

If the system exhibits proportional damping, it can be modeled by adding a damping term to the equation of motion. Assuming that the damping is proportional to the generalized velocities, we introduce the damping matrix $\mathbf{C}$, resulting in the equation:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{Q}. \tag{3.30}$$

Here, $\mathbf{C}$ is the proportional damping matrix, which can be expressed in terms of the mass and stiffness matrices (INMAN, 2014; CHOPRA, 2014):

$$\mathbf{C} = \alpha\mathbf{M} + \beta\mathbf{K}, \tag{3.31}$$

with α and β being damping coefficients determined experimentally or based on design criteria.

The Lagrange formulation enables the systematic derivation of the equation of motion for systems with multiple degrees of freedom, considering kinetic energy, potential energy, and external forces. In its most general form, the final equation can be expressed as (INMAN, 2014):

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{Q}. \tag{3.32}$$

The matrix $\mathbf{M}$ is the mass (or inertia) matrix, representing the inertia of the system. The vector $\ddot{\mathbf{q}}$ is the acceleration vector, i.e., the second derivative of the displacement vector. The matrix $\mathbf{C}$ is the damping matrix, which incorporates the dissipative forces in the system. The vector $\dot{\mathbf{q}}$ is the velocity vector, i.e., the first derivative of the displacement vector. The matrix $\mathbf{K}$ is the stiffness matrix, representing the structural rigidity of the system. The vector $\mathbf{q}$ is the displacement vector, indicating the deformations of the system. Finally, the vector $\mathbf{Q}(t)$ represents the external force vector, which in this work corresponds to the fluid-dynamic forces that may vary with time t.

Shabana (1997) presents the relationship between Hamilton's Principle and the Lagrangian formulation, demonstrating that the former can be used to deduce Lagrange's equation.

Hamilton's principle is expressed as:

$$\delta \int_{t_1}^{t_2} (T - V)\, dt + \delta W_{\text{nc}} = 0, \tag{3.33}$$

where:

- T is the kinetic energy,
- V is the potential energy,
- W_{nc} is the virtual work of the non-conservative forces.

This principle states that the variation of the integral of the kinetic and potential energies, added to the integral of the virtual work of the non-conservative forces, must be zero over any time interval (SHABANA, 1997).

The book shows that, by applying this principle and integrating by parts, it is possible to directly obtain Lagrange's equation:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i, \tag{3.34}$$

or, more compactly:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i, \tag{3.35}$$

where $L = T - V$ is the Lagrangian of the system.

Therefore, Lagrange's equation can be derived from Hamilton's principle, making this approach an equivalent alternative for the formulation of the equations of motion (SHABANA, 1997).

3.3.2.3 Modal Analysis

Modal analysis is a technique that allows expressing the dynamic response of a structural or mechanical system as a sum of the individual contributions of its inherent vibration modes. Each mode is characterized by a natural frequency and an associated mode shape (INMAN, 2014; CHOPRA, 2014).

The central idea of modal analysis is to transform a multiple degree-of-freedom (MDOF) system into a set of independent single degree-of-freedom (SDOF) systems. This transformation is possible because the differential equations governing the system can be diagonalized using the eigenvectors (vibration modes) of the system (MORETTI, 1993).

Vibration Modes and Natural Frequencies

Vibration modes are the specific patterns of motion that a system assumes when oscillating at its natural frequencies. They are the solutions of the eigenvalue problem associated with the governing differential equations. Each mode describes a characteristic deformation or oscillation pattern, where all parts of the system move in a coordinated manner, even though the amplitudes are different (PAZ, 2004).

The natural frequency is the rate at which a system oscillates freely, without external excitation or damping. For a discrete MDOF system, the natural frequencies can be obtained by solving the eigenvalue problem associated with the stiffness matrix $\mathbf{K}$ and the mass matrix $\mathbf{M}$. This problem is usually expressed as (INMAN, 2014):

$$(\mathbf{K} - \omega^2 \mathbf{M})\boldsymbol{\phi} = \mathbf{0}. \tag{3.36}$$

In this equation, $\boldsymbol{\phi}$ represents the eigenvector (vibration mode) and ω is the natural frequency (the square root of the eigenvalue).

Derivation of the Modal Analysis Equations

Consider a linear undamped system. The equation of motion for a MDOF system is given by:

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}.(t) = 0 \tag{3.37}$$

Assume that the response can be written as a linear combination of the eigenmodes:

$$\mathbf{u}(t) = \mathbf{\Phi}\mathbf{q}(t), \tag{3.38}$$

where $\mathbf{\Phi}$ is the matrix whose columns are the eigenvectors (mode shapes) and $\mathbf{q}(t)$ is the vector of modal coordinates.

Substituting this expression into the equation of motion yields:

$$\mathbf{M}\mathbf{\Phi}\ddot{\mathbf{q}}(t) + \mathbf{K}\mathbf{\Phi}\mathbf{q}(t) = 0. \tag{3.39}$$

Multiplying the above equation on the left by the transpose of the modal matrix $\mathbf{\Phi}^T$ and using the orthogonality properties (i.e., $\mathbf{\Phi}^T\mathbf{M}\mathbf{\Phi}$ becomes a diagonal matrix — often normalized to the identity — and $\mathbf{\Phi}^T\mathbf{K}\mathbf{\Phi}$ transforms into a diagonal matrix $\mathbf{\Lambda}$ containing the eigenvalues ω_n^2), we obtain:

$$\ddot{\mathbf{q}}(t) + \mathbf{\Lambda}\mathbf{q}(t) = 0. \tag{3.40}$$

For each mode n, the equation becomes:

$$\ddot{q}_n(t) + \omega_n^2 q_n(t) = 0. \tag{3.41}$$

In summary, modal analysis decomposes the dynamic response of complex systems into independent modal contributions, thereby simplifying the problem by reducing it to a set of uncoupled SDOF systems (CHOPRA, 2014). By identifying the natural frequencies and vibration modes, it is possible to understand and predict the behavior of structures subject to dynamic excitations.

3.4 Fluid-Structure Interaction Coupling

Fluid-Structure Interaction (FSI) occurs when the motion of a fluid influences a structure and vice versa. The numerical modeling of FSI problems requires coupling techniques that link the governing equations of fluid flow and structural mechanics. Two main approaches are typically used for coupling: weak coupling and strong coupling (SIGRIST, 2015; CROLET, 2017).

3.4.1 Weak Coupling

The weak coupling method, also known as partitioned or loosely coupled FSI, treats the fluid and structure solvers separately, exchanging information at discrete time steps. The fluid solver calculates the pressure and shear stresses on the structure, which then deforms accordingly. This updated structural configuration is passed back to the fluid solver for the next iteration (SIGRIST, 2015).

Although computationally efficient, weak coupling may lead to instabilities in simulations with high fluid-structure interaction effects, particularly in cases of added mass effect, where fluid forces significantly alter the structural response. The numerical stability of weak coupling can be improved by adjusting the time step size and applying relaxation techniques to mitigate divergence issues (CROLET, 2017).

3.4.2 Strong Coupling

Strong coupling methods fully integrate the fluid and structural solvers at each time step, ensuring that the solution satisfies both governing equations simultaneously. This is achieved using iterative schemes where equilibrium conditions are imposed at the fluid-structure interface until convergence is reached (SIGRIST, 2015).

Compared to weak coupling, strong coupling provides better numerical stability and accuracy, especially for problems where fluid forces have a significant impact on structural motion. However, the computational cost is higher due to the iterative nature of the coupling scheme. Strongly coupled methods are widely used in high-fidelity simulations, including aeroelasticity, biomedical applications, and industrial fluid-structure interactions (CROLET, 2017).

3.4.3 Numerical Integration

Numerical integration plays a crucial role in FSI simulations, particularly in solving the governing equations of fluid and structural motion. Various integration schemes, such as explicit and implicit methods, are used depending on the problem requirements and computational efficiency (DAVIS; RABINOWITZ, 1984).

Explicit methods, like the forward Euler and Runge-Kutta schemes, are conditionally stable and require small time steps, making them suitable for transient fluid-structure interactions with high-frequency structural responses. Implicit methods, such as backward Euler and Crank-Nicholson, offer unconditional stability and are preferred for strongly coupled FSI problems where numerical stability is essential (DAVIS; RABINOWITZ, 1984).

Computational precision is another critical aspect in FSI simulations, as numerical errors can accumulate over time, affecting solution accuracy. Adaptive time-stepping and error control mechanisms are commonly implemented to ensure solution robustness (DAVIS; RABINOWITZ, 1984).

4 Numerical and Computational Modeling

4.1 Immersed Boundary Method

The Immersed Boundary Method (IBM) was proposed by Peskin (1972) in the 1970s to simulate fluid-structure interaction in cardiac valves (PESKIN, 2002). This technique enables the modeling of complex surfaces without requiring a mesh adapted to the structure's contour, by using a fixed Eulerian mesh for the fluid and a distribution of Lagrangian points to represent the structural interface.

The main advantage of IBM lies in its ability to handle moving and highly deformable interfaces without the need for mesh reconstruction, making it an efficient alternative for modeling fluid-structure interaction (FSI) in engineering and biomedical applications (GRIFFITH; PATANKAR, 2020). Although its prominence was established in the 1970s, the method's foundations date back to the 1960s, with early developments in fictitious domain and penalization methods for treating immersed interfaces (VERZICCO, 2023). Since then, several variations have been proposed, among which the following stand out:

- **Penalization methods**: Employ a restoring force that forces the fluid velocity field to match that of the structure (GRIFFITH; PATANKAR, 2020);
- **Immersed interface methods**: Impose boundary conditions directly at the points that compose the interface;
- **Direct interpolation methods**: Exemplified by the Multi-Direct Forcing (MDF) technique, which improves the enforcement of the no-slip condition (WANG; FAN; LUO, 2008).

In IBM, the Navier–Stokes equations are solved in the fluid domain with the inclusion of a force term f in the momentum equation:

$$\frac{\partial u}{\partial t} + u \cdot \nabla u = -\frac{1}{\rho}\nabla p + \nu \nabla^2 u + f. \tag{4.1}$$

The force f is determined from the Lagrangian interface and distributed to the Eulerian points through a regularized delta function:

$$f(x,t) = \int F(X,t)\, \delta(x - X)\, dX. \tag{4.2}$$

where $F(X,t)$ represents the force on the interface and $\delta(x - X)$ is the regularized delta function used to spread this force over the fluid mesh (GRIFFITH; PATANKAR, 2020).

4.1.1 Multi-Direct Forcing (MDF) Method

The Multi-Direct Forcing (MDF) Method was developed to enhance the accuracy of IBM in enforcing the no-slip condition at the fluid-structure interface. Through successive iterations, MDF adjusts the interaction force until the interface condition is satisfied within a predetermined numerical error. The basic MDF equation is:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + f_i. \tag{4.3}$$

where the term f_i is updated iteratively until the desired precision is achieved (CHEN et al., 2025).

4.1.2 Eulerian-Lagrangian Coupling

IBM is characterized by the coupling between the Eulerian domain (fluid) and the Lagrangian domain (structure). In this approach, the fluid is solved on a fixed mesh while the structure is represented by a collection of moving points, which proves particularly effective in FSI simulations involving significant structural deformations (GRIFFITH; PATANKAR, 2020).

In MFSim, the IBM and MDF methodologies are employed to efficiently model fluid–structure interaction. IBM allows the modeling of the interface without the need to adjust the mesh to the contour of the solid structure, by using two distinct meshes: a fixed Eulerian mesh for the fluid solution and a moving Lagrangian mesh to represent the structural interface. This approach avoids the high computational costs associated with re-meshing, facilitating the modeling of complex interfaces (MORALES et al., 2023).

Meanwhile, MDF enhances IBM by ensuring a more rigorous enforcement of the no-slip condition at the interface. The iterative method enables the calculation of the fluid dynamic forces at the interface of the immersed body until the condition is satisfied with minimal numerical error (MORALES et al., 2023; CHEN et al., 2025). These forces are then interpolated to the finite element model's mesh, thereby enabling the determination of the structural response. An illustrative scheme of this information flow is presented in Fig. 4.

According to Fig. 4, the content to be addressed in the remainder of the current chapter can be outlined, beginning with the theoretical definitions of the Eulerian and Lagrangian frames. In the Eulerian frame, the flow field is described by fixing a coordinate system in space and analyzing the variation of fluid properties, such as velocity and pressure, as the fluid moves through this reference system. This method is widely employed in solving the Navier–Stokes equations in fixed domains, making it ideal for problems that require the analysis of flow behavior in a specific region of space. In computational applications, it is commonly used with a fixed mesh, allowing the fluid to move through it (BLAZEK, 2001).

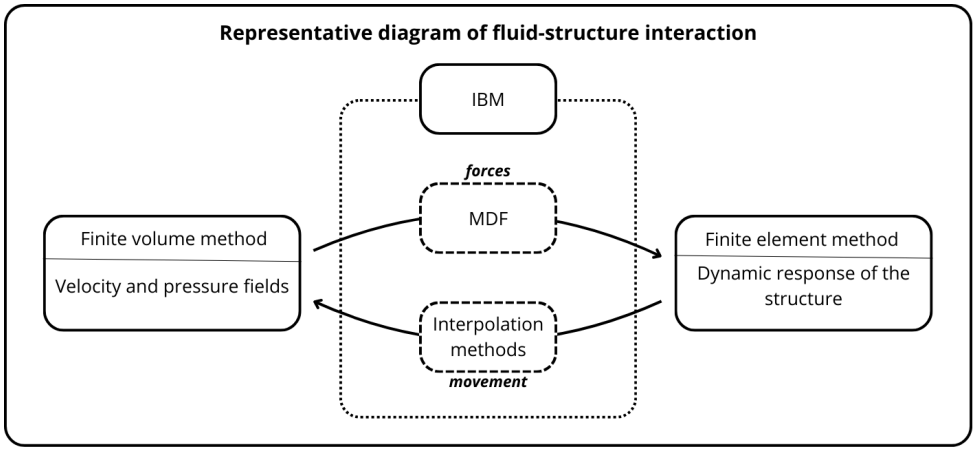


Figure 4 – Diagram illustrating the computational framework for fluid–structure interaction (FSI) using the immersed boundary method (IBM), multi–direct forcing (MDF), the finite volume method (FVM), and the finite element method (FEM).

In contrast, the Lagrangian frame focuses on tracking individual fluid particles, allowing the observation of the evolution of their properties over time. Instead of analyzing fixed points in space, the displacement of the particles is monitored, which makes this approach particularly useful in the modeling of discrete particles, structural deformations, and fluid–structure interactions, where the mesh can move and deform according to the flow (BLAZEK, 2001; SIGRIST, 2012).

The need to define these two concepts arises because the immersed boundary method utilizes a mesh to represent the immersed surface. It is common to refer to the mesh used in IBM as the Lagrangian mesh, in order to differentiate it from the Eulerian mesh employed in the finite volume method. From this point onward, the mesh used in the immersed boundary method will be referred to as the Lagrangian mesh. The format of the mesh and its characteristics will be presented in the following section.

4.2 IBM-FEM Coupling

In the previous section, the coupling between the immersed boundary method and the finite element method was briefly discussed. The following section details this coupling and presents the developments implemented in the MFSim software for integrating the two approaches.

Initially, the Lagrangian mesh, representing the immersed surface, is introduced. Within the MFSim framework, the mesh used to represent the submerged geometry must consist of three-node triangular elements, as shown in Fig. 5. The Lagrangian mesh is composed of triangular elements, and the geometric center of each triangle is referred to as the Lagrangian point, while the nodes constituting the triangle are called vertices. The Lagrangian points (highlighted in orange) correspond to the geometric centroids of each

triangle. The zoomed-in view in Fig. 5 provides a detailed visualization of the triangular elements and their associated points.

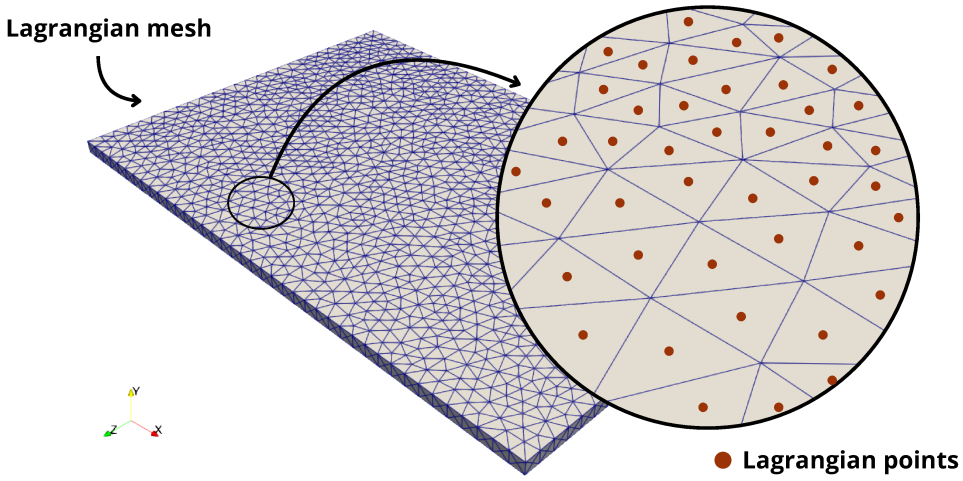


Figure 5 – Representation of a Lagrangian mesh and Lagrangian points.

In the data structure storing the Lagrangian mesh, the centroids of the triangles serve as the unique elements, as they are assigned exclusive indices, in contrast to the vertices. This organization contrasts with that of a finite element mesh, in which each structural element—regardless of its shape—is assigned a unique index, as are the nodes that define it.

The coupling between the immersed boundary method and the finite element model is based on a bidirectional flow of information. Initially, the forces determined in each element of the Lagrangian mesh are transmitted to the structural model. Through the solution of the equation of motion—as addressed in the numerical integration section—the model defines the new position of the structure (performing temporal integration in the modal domain) and calculates its velocity, expressed in the displacement and velocity vectors in the physical domain. Subsequently, these displacements and velocities are transferred back to the Lagrangian mesh, thus completing the communication cycle.

It is important to note that this flow of information depends on a topological association between the Lagrangian mesh and the finite element mesh. This correspondence is fundamental because, as the Lagrangian mesh represents the immersed surface, the loading applied to that surface must be correctly transmitted to the corresponding structural elements.

Although the term *applying load to structural elements* is used as an analogy — in practice, the loads are applied to the nodes of the structural elements — it serves to illustrate the concept of a topological association between the surface of the Lagrangian mesh and that of the finite element mesh. In practical terms, the finite element mesh is overlaid by the Lagrangian mesh.

Finally, this topological association must be addressed from an algorithmic perspective. A robust method for associating the meshes must be established, one that is capable of handling geometries with complex contours, potential singularities, and possible limitations of the method. Moreover, the method must be efficient in processing large volumes of data and applicable in parallel execution environments, meeting the efficiency requirements in cases that demand high-performance computing (HPC).

The following subsections will discuss the key developments and contributions of this work, focusing on the algorithm for associating the Lagrangian mesh with the structural mesh, as well as the development of interpolation functions used to transfer the structure's motion to the Lagrangian mesh.

4.2.1 Lagrangian mesh association

The association between the Lagrangian mesh and the structural mesh should be approached algorithmically, enabling the correct assignment of immersed boundary points to the corresponding structural elements. For this, it is essential to have the coordinates of the nodes that define the position of the structure in space, as well as the connectivity list of the structural elements. This list specifies which nodes belong to each element, allowing the classification of the elements and their proper assignment to the immersed boundary points.

The classification of the type of element present in the structural mesh depends on the format in which the data is exported in the connectivity list, which may vary depending on the software used. Generally, this classification is based on the number of nodes and the geometric dimension of the structural elements, which can be described as:

- **One-dimensional (1D):** elements with a single predominant dimension, such as bars or beams;
- **Two-dimensional (2D):** elements with two dimensions, such as triangles, quadrilaterals, or other polygons;
- **Three-dimensional (3D):** elements that occupy a volume, such as tetrahedrons, hexahedrons (cubes), prisms, or pyramids.

For the implementation of the association between the Lagrangian mesh and the structural mesh, two main approaches can be considered. The first, called orthogonal projection search, consists of projecting the points of the Lagrangian mesh onto the elements of the structural mesh, identifying the closest element. The second approach proposes representing the structural mesh as a graph, allowing the use of operations and algorithms inherent to this data structure to facilitate the association between the immersed boundary points and the structural elements.

With the purpose of simplification, the examples considered in the algorithm and mathematical development focus on a quadrilateral mesh element. However, the concepts presented are extrapolated to other structural mesh element formats, ensuring the applicability of the methodologies to different types of discretization.

4.2.1.1 Search for Orthogonal Projection

Given a vertex of the Lagrangian mesh, $\mathbf{v}_{\text{IBM}} \in \mathbb{R}^3$, the first step is to project it orthogonally onto the elements of the structural mesh. Each element of the structural mesh, denoted by e_{FEM}, is considered as a subspace of the three-dimensional space. The projection of $\mathbf{v}_{\text{IBM}}$ onto the element e_{FEM}, denoted by $\pi_{e_{\text{FEM}}}(\mathbf{v}_{\text{IBM}})$, is valid only if the projected point lies within the interior of the element (for example, by using barycentric coordinates in triangles (ERICSON, 2004)).

The shortest distance between $\mathbf{v}_{\text{IBM}}$ and the subspace defined by e_{FEM} occurs at the orthogonal projection $\pi_{e_{\text{FEM}}}(\mathbf{v}_{\text{IBM}})$. Thus, for each element in which the projection is valid, one computes (BERG et al., 2008; ERICSON, 2004):

$$\text{dist}(\mathbf{v}_{\text{IBM}}, e_{\text{FEM}}) = \left\| \mathbf{v}_{\text{IBM}} - \pi_{e_{\text{FEM}}}(\mathbf{v}_{\text{IBM}}) \right\|. \tag{4.4}$$

The structural element that minimizes this distance is selected as the one closest to $\mathbf{v}_{\text{IBM}}$.

Sign Method for Verifying Point Containment in a Quadrilateral

To determine whether a projected point P is contained in a quadrilateral defined by the vertices A, B, C, and D, the directional vectors of the edges are first constructed:

$$\vec{AB} = B - A \tag{4.5}$$

$$\vec{BC} = C - B \tag{4.6}$$

$$\vec{CD} = D - C \tag{4.7}$$

$$\vec{DA} = A - D \tag{4.8}$$

Next, the vectors connecting the point P to the vertices are defined:

$$\vec{AP} = P - A \tag{4.9}$$

$$\vec{BP} = P - B \tag{4.10}$$

$$\vec{CP} = P - C \tag{4.11}$$

$$\vec{DP} = P - D \tag{4.12}$$

Afterwards, the cross products between the edge vectors and the corresponding vectors from P to the vertices are computed:

$$\vec{N}_{AB} = \vec{AB} \times \vec{AP} \tag{4.13}$$

$$\vec{N}_{BC} = \vec{BC} \times \vec{BP} \tag{4.14}$$

$$\vec{N}_{CD} = \vec{CD} \times \vec{CP} \tag{4.15}$$

$$\vec{N}_{DA} = \vec{DA} \times \vec{DP} \tag{4.16}$$

The relative orientation of the point P with respect to the edges is determined by the consecutive dot products (BERG et al., 2008):

$$S_1 = \vec{N}_{AB} \cdot \vec{N}_{BC} \tag{4.17}$$

$$S_2 = \vec{N}_{BC} \cdot \vec{N}_{CD} \tag{4.18}$$

$$S_3 = \vec{N}_{CD} \cdot \vec{N}_{DA} \tag{4.19}$$

$$S_4 = \vec{N}_{DA} \cdot \vec{N}_{AB} \tag{4.20}$$

The point P is considered to be contained within the quadrilateral if, for $i = 1, 2, 3, 4$, the following condition is satisfied:

$$S_i \geq -\varepsilon \quad \text{for } i = 1, 2, 3, 4 \tag{4.21}$$

This approach, which uses the sign method, has a computational complexity of $O(1)$ for quadrilaterals and ensures robustness in the containment verification, even in degenerate cases, by using the tolerance parameter ε.

Next, some algorithms in the form of pseudocode that integrate the steps of orthogonal projection and containment verification are presented. Each algorithm is accompanied by a brief explanation of its functionality and the steps performed.

Orthogonal Projection Verification

The "Orthogonal Projection Verification" algorithm iterates over each element of the structural mesh M and performs the orthogonal projection of the point P onto the element E. If the distance between P and its projection is less than the tolerance ε and if the sign method confirms that the point is contained in E, that element is considered a candidate. In the end, the algorithm returns the element that has the smallest distance relative to point P.

Algorithm 1 Orthogonal Projection Verification

Input: Point P, Structural Mesh M, Tolerance ε
Output: Best Structural Element and Minimum Distance
$Best\_Element \leftarrow$ NULL
$Best\_Distance \leftarrow \infty$
for each element $E \in M$ **do**
  $Projection\_P \leftarrow$ Project_Point_OrthogonallyP, E
  if $Distance(P, Projection\_P) < \varepsilon$ **then**
    if Verify_Sign_Method$Projection_P, E$ **then**
      $Current\_Distance \leftarrow Distance(P, Projection\_P)$
      if $Current\_Distance < Best\_Distance$ **then**
        $Best\_Distance \leftarrow Current\_Distance$
        $Best\_Element \leftarrow E$
      end if
    end if
  end if
end for
return $(Best\_Element, Best\_Distance)$

Orthogonal Point Projection

This algorithm takes a point P and an element E from the structural mesh and projects P orthogonally onto the plane defined by element E. To do this, it first determines the plane in which E lies and then projects the point onto that plane.

Algorithm 2 Orthogonal Point Projection

Input: Point P, Element E
Output: Projected Point P'
$Plane\_E \leftarrow$ Define_Element_Plane$E$
$Projection\_P \leftarrow$ Project_P_Onto_Plane$P, Plane_E$
return $Projection\_P$

Sign Method

The "Verify Containment by the Sign Method" algorithm determines if the projected point $Projection\_P$ is contained within the element E (in this case, a quadrilateral). To do so, it obtains the vertices of the element, calculates the edge vectors and the vectors connecting $Projection\_P$ to each vertex. Then, it computes the cross products and finally checks the signs of the resulting dot products to confirm the containment of the point.

Algorithm 3 Verify Containment by the Sign Method

Input: Projected Point $Projection\_P$, Element E
Output: True if contained, False otherwise
$Vertices \leftarrow$ Get_Vertices$E$ {Assuming $E = \{A, B, C, D\}$}
$AB \leftarrow B - A$
$BC \leftarrow C - B$
$CD \leftarrow D - C$
$DA \leftarrow A - D$
$AP \leftarrow Projection\_P - A$
$BP \leftarrow Projection\_P - B$
$CP \leftarrow Projection\_P - C$
$DP \leftarrow Projection\_P - D$
$Cross\_AB\_AP \leftarrow AB \times AP$
$Cross\_BC\_BP \leftarrow BC \times BP$
$Cross\_CD\_CP \leftarrow CD \times CP$
$Cross\_DA\_DP \leftarrow DA \times DP$
if $(Cross\_AB\_AP \cdot Cross\_BC\_BP < 0)$ **or** $(Cross\_BC\_BP \cdot Cross\_CD\_CP < 0)$ **or** $(Cross\_CD\_CP \cdot Cross\_DA\_DP < 0)$ **or** $(Cross\_DA\_DP \cdot Cross\_AB\_AP < 0)$ **then**
  return False
end if
return True

4.2.1.2 Graph-Based Structural Mesh Representation Method

The structural mesh is represented by a graph. Before projecting a vertex, a search is performed for the closest structural node to it. If this distance is less than a reference distance, the structural elements that contain that node are determined. The projection will then be carried out only on the structural elements that contain the closest structural node to the vertex.

Representing the structural mesh as a graph brings significant computational and algorithmic advantages. A graph is a data structure composed of a set of vertices (or nodes) and a set of edges that connect pairs of these vertices, allowing for the natural modeling of the connectivity and adjacency relationships present in the mesh and facilitating operations such as searches, traversals, and neighborhood queries (SEDGEWICK, 2002).

In the context of projecting vertices from the Lagrangian mesh onto the structural mesh, the approach consists of first searching for the closest structural node to the vertex before projecting it; if the distance between the vertex and that node is less than a defined reference value, the structural elements that contain that node are determined, thereby restricting the projection to only those elements.

Algorithm 4 Graph-Based Mesh Association

Input: Vertex v (from the Lagrangian mesh), Structural Mesh Graph G, Reference Distance d_{ref}
Output: Best Structural Element E_{best} and Projection Point P_{best}
$node \leftarrow \text{FindClosestNode}(v, G)$
$d \leftarrow \text{Distance}(v, node.position)$
if $d \geq d_{ref}$ **then**
  return `NULL`
end if
$E \leftarrow \text{GetAssociatedElements}(node)$
$d_{min} \leftarrow \infty$
$E_{best} \leftarrow$ `NULL`, $P_{best} \leftarrow$ `NULL`
for each element $e \in E$ **do**
  $P \leftarrow \text{ProjectVertexOntoElement}(v, e)$
  $d_e \leftarrow \text{Distance}(v, P)$
  if $d_e < d_{min}$ **then**
    $d_{min} \leftarrow d_e$
    $E_{best} \leftarrow e$
    $P_{best} \leftarrow P$
  end if
end for
return (E_{best}, P_{best})

This strategy reduces the number of orthogonal projections by decreasing the search space and enabling the application of efficient graph theory algorithms, even in complex meshes, such as breadth-first search (BFS) and shortest path algorithms (WILSON, 1996; BONDY; MURTY, 2008). In this way, representing the structural mesh as a graph leverages the benefits of an organized structure and optimized algorithms, enhancing the association between the Lagrangian and structural meshes and making the process faster.

4.2.1.3 Challenges and Limitations of the Projection Method

The projection method is used to verify whether the vertex of the Lagrangian mesh, when projected, lies within a structural element. By treating the structural mesh as a graph, the association algorithm becomes more efficient in terms of cost and computational time.

In MFSim, the Search for Orthogonal Projection approach has already been implemented and is in use. However, due to the scale of the industrial case motivating this work,

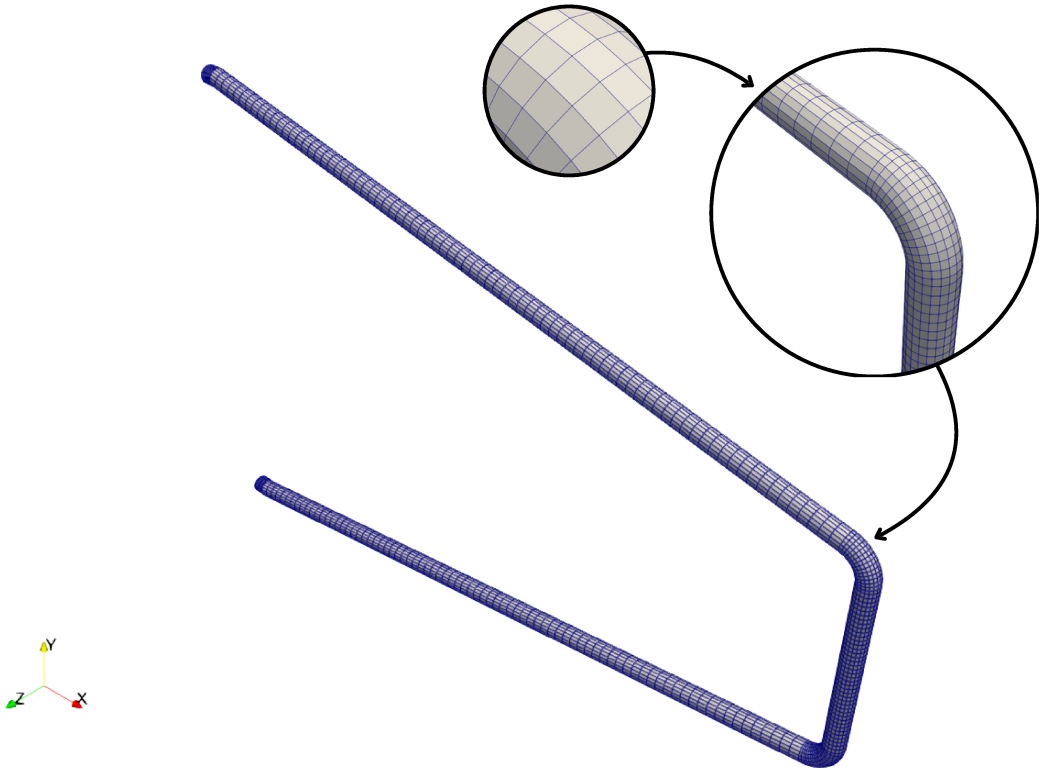
the Graph-Based Structural Mesh Representation Method was proposed, as associating a set of more than 100 million vertices from the Lagrangian mesh with approximately 7 million structural elements incurs a significant computational cost. This new approach is still being adapted, since, considering both flexibility and cost criteria, it is preferable that the mesh association for industrial cases be performed outside MFSim and then incorporated into the software via a user input file.

The use of orthogonal projection aims to ensure that the projected vertex is indeed inside the structural element. This guarantee is essential because the interpolation method—used to transfer displacements and velocities from the structure to the Lagrangian mesh—is based on spatial interpolation, a topic that will be addressed in the following sections of this work.

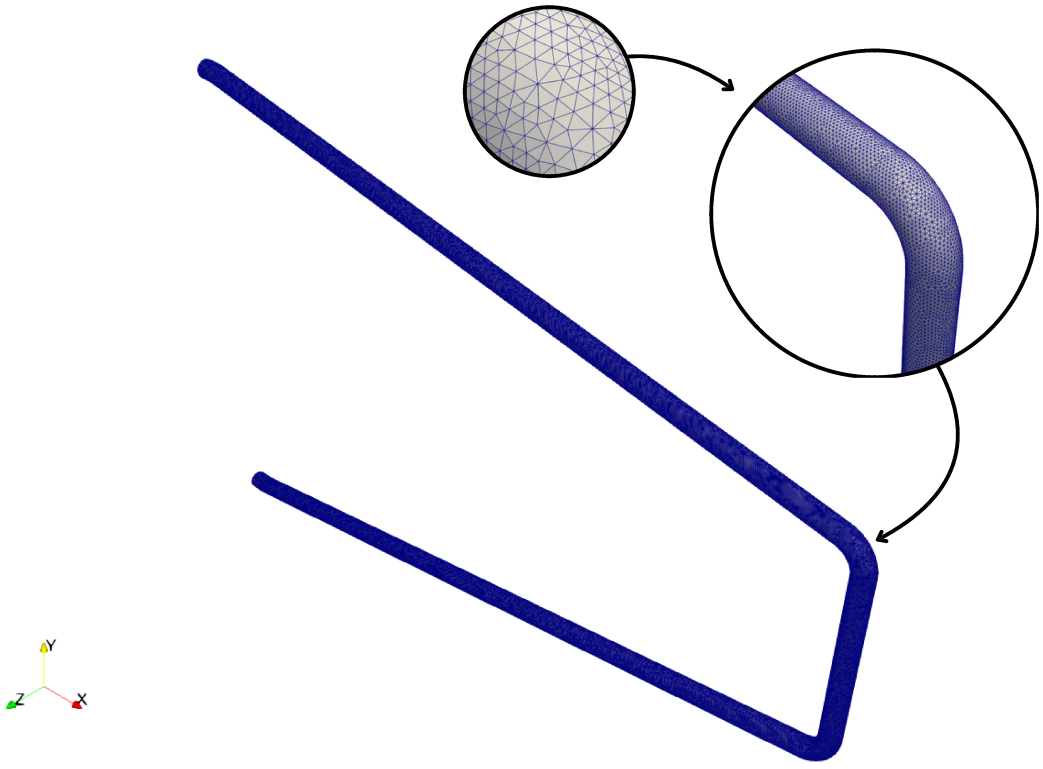
In some cases, the topological characteristics of the Lagrangian and structural meshes may hinder the application of orthogonal projection. This difficulty occurs mainly when the elements of the Lagrangian mesh are smaller than those of the structural mesh. This situation arises because the size of the Lagrangian elements is directly related to the creation of the Eulerian mesh (finite volume mesh). In the adopted one-to-one relation, the side of the triangular Lagrangian element must be equal to or smaller than the side of the finite volume at the finest level of the Eulerian mesh. However, the structural mesh elements are usually larger, which necessitates specific treatments in the Lagrangian mesh to ensure that the projection of the vertices onto the structural elements is correct.

Before detailing these treatments, it is useful to compare the dimensions of the elements in the structural and Lagrangian meshes. To this end, a characteristic geometry of heat exchanger systems was created to be used as a case study, inspired by the work of Wang et al. (2024b), which can be seen in Fig. 6. Upon analyzing the figure, a difference in the refinement level between the two meshes is observed, which directly impacts the orthogonal projection algorithm. Although the algorithm has been previously introduced, this section aims to synthesize and deepen the discussion, considering the relationship between the geometry of structural elements and the Lagrangian mesh.

In the case of beam or bar finite elements, these are represented as line segments and will be referred to in this work as 1D elements (having a single geometric dimension). In practice, the orthogonal projection does not occur directly onto a line segment but rather onto an infinite line (KRYLOV; LOBANDIYEVSKY; MEN, 1968).



(a) Structural mesh



(b) Lagrangian mesh

Figure 6 – Representation of the structural mesh (a) and the Lagrangian mesh (b) for the pipeline of a heat exchanger system.

Thus, to project a vertex of the Lagrangian mesh onto the line segment representing a 1D element, the procedure follows the steps illustrated in Fig. 7.

- Project the point onto the infinite line defined by the element nodes;
- Verify whether the projected point lies within the limits of the segment defined by the element nodes;
- Calculate the orthogonal distance between the vertex and the infinite line.

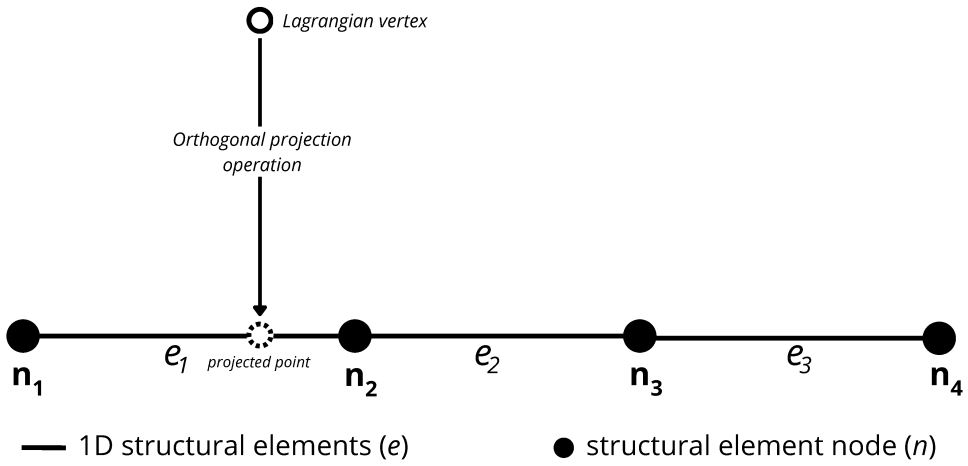


Figure 7 – Scheme of the orthogonal projection process for 1D structural elements. The Lagrangian mesh vertex is orthogonally projected onto the structural element, resulting in a projected point within the element domain. Structural nodes (n) are represented by black circles, while 1D structural elements (e) are indicated by black lines connecting the nodes.

For planar elements, such as plate and shell finite elements, the orthogonal projection process follows an approach similar to that used for 1D elements. The vertex of the Lagrangian mesh is orthogonally projected onto the mid-plane defined by the nodes of the structural element. In this work, elements with planar geometry will be referred to as 2D elements, since they are defined in two dimensions.

The projection of the vertex onto the mid-plane of the structural element is performed using the orthogonal projection search algorithm, as shown in Fig. 8. Next, it is verified whether the projected point lies within the structural element using the sign method, as previously presented.

This procedure applies to structural elements with a quadrilateral mesh topology of four nodes (QUAD4) and triangular elements with three nodes (TRIA3), associated with linear finite element theory. The algorithm can be easily extended to quadratic and cubic theory elements, which include quadrilateral elements with 6, 8, or 9 nodes and triangular elements with 6 or 7 nodes (COOK et al., 2001). The choice of topology depends on the finite element theory implemented in each software.

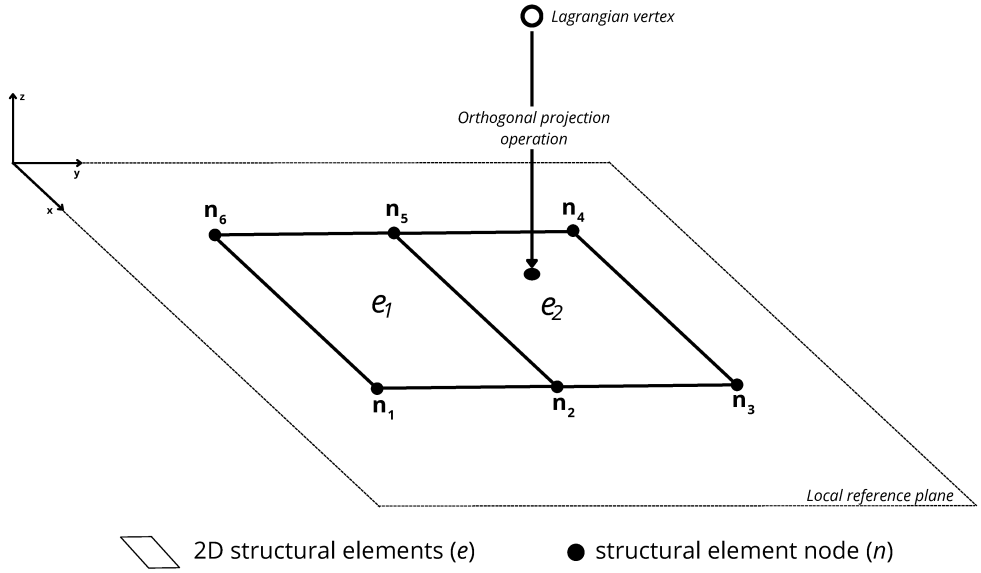


Figure 8 – Scheme of the orthogonal projection process for 2D structural elements. The Lagrangian mesh vertex is orthogonally projected onto the structural element, resulting in a projected point within the element domain. Structural nodes (n) are represented by black circles, while 2D structural elements (e) are indicated by black quadrilateral shapes.

For structural elements geometrically represented by a volume, such as hexahedral and pyramidal solid finite elements, the algorithm requires a slight modification. The orthogonal projection must be performed on the face that "sees" the vertex of the Lagrangian mesh, that is, only on the faces of the structural elements whose normal points toward this vertex. Since 3D elements are not the focus of this work, the application of the algorithm to these cases will not be detailed.

The orthogonal projection tends to fail when the geometry features a Radius of Curvature or a Rounded Transition in its topology. Both cases are illustrated in Fig. 6. The curved pipe has a bend where the curvature in the Lagrangian mesh (Fig. 6b) is nearly smooth, while in the structural mesh (Fig. 6a), there is a non-smooth transition between the planes defined by the structural elements forming the bend. To better highlight the non-smooth transition in the bend, Fig. 9 is presented. From this figure, it is possible to see that in the 2D projection of the curve, the curvature of the structural mesh is not perfectly smooth, which means that these cases should have a well-defined solution.

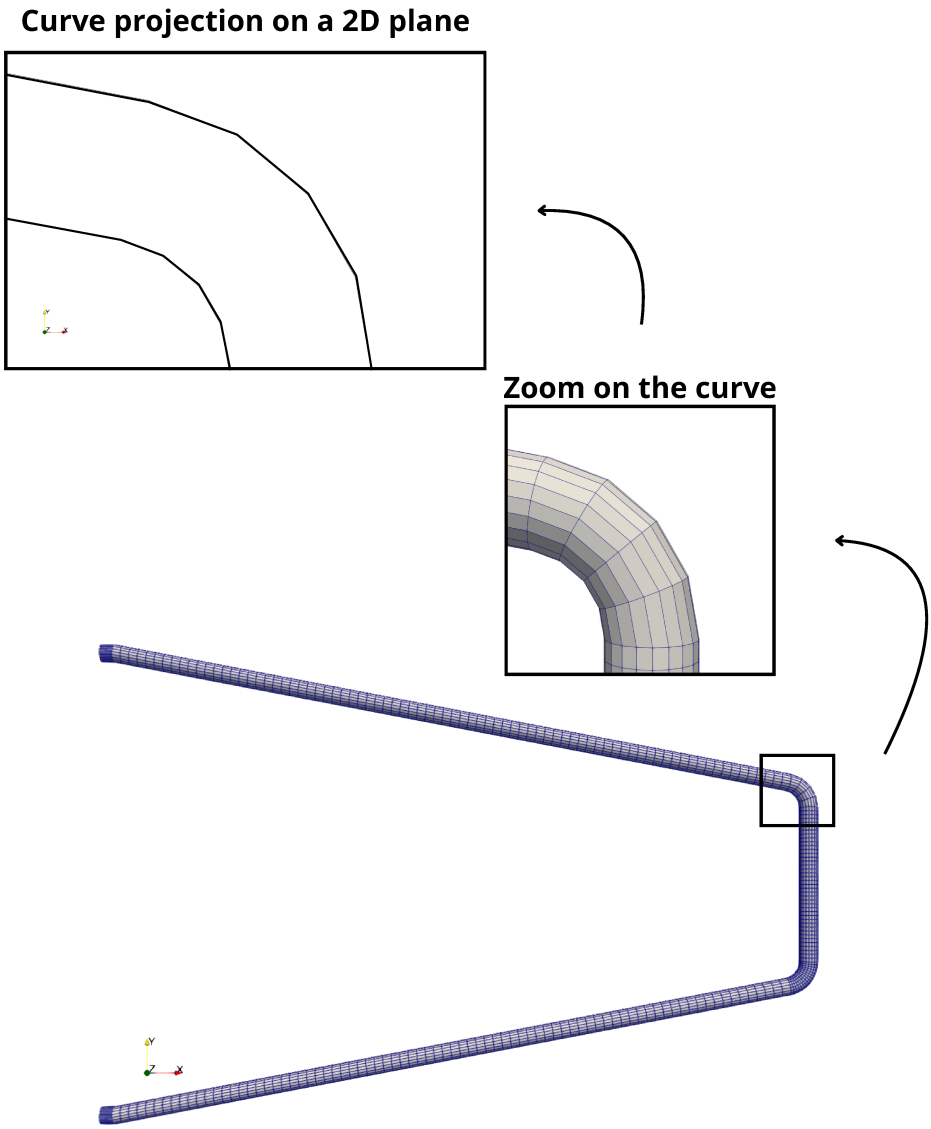


Figure 9 – Visualization of the curvature in the structural mesh. The details highlight the transition in the curved region.

The vertices of the Lagrangian mesh, when projected onto the structural elements located in curved regions, are not correctly mapped inside the structural elements. This issue is further illustrated in Fig. 10, which uses the 2D projection of the curve from Fig. 9 to highlight cases where orthogonal projection tends to fail.

In both cases presented in Fig. 10, the Lagrangian mesh vertex is topologically above the structural elements e_1 and e_2. When the vertex is projected onto structural element e_1, its projection does not lie between the structural nodes n_1 and n_2 (case a). Similarly, when the vertex is projected onto the plane containing structural element e_2, its projection does not lie between the structural nodes n_2 and n_3 (case b). If a vertex is not mapped, fluid dynamic forces are not transmitted to the structure in that region, and consequently, the structure does not inform its motion to the immersed boundary. Therefore, a robust approach is essential to handle cases where orthogonal projection fails to find the structural

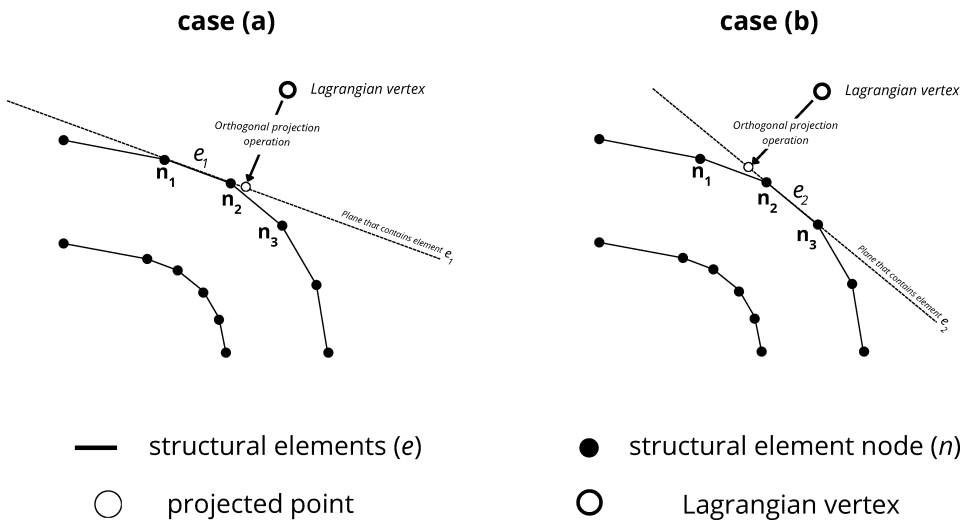


Figure 10 – Visualization of the curvature in the structural mesh. The details highlight the transition in the curved region.

element due to non-uniform curvatures.

To define the approach that addresses cases where orthogonal projection fails, it is necessary to review some key concepts. The Lagrangian vertex is part of a triangle, whose centroid is referred to as the Lagrangian point, as illustrated in Fig. 11. Essentially, this figure is an adaptation of Fig. 5. Additionally, it is important to emphasize that MFSim is a software designed for execution in a parallel environment, which has a fundamental impact on how incorrect projections occur.

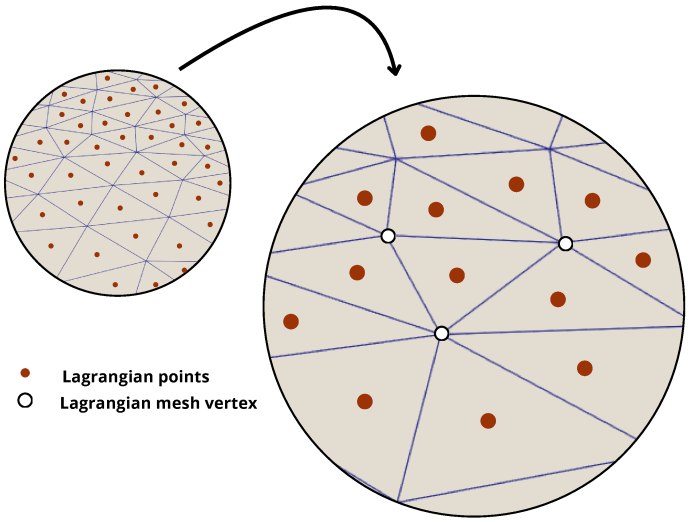


Figure 11 – Lagrangian vertex mesh visualization. Lagrangian points are shown in brown, while Lagrangian mesh vertices are represented in white.

One of the main challenges of parallel execution is that Lagrangian points are distributed across different processes. The philosophy behind parallel execution is that the information contained in each process should be used independently of the data in other processes.

However, in computational simulations, this principle cannot always be fully applied due to the physical connection between the data distributed across processes.

Chapman, Jost e Pas (2007) explain that parallel execution involves distributing work among multiple processing units, allowing tasks to be performed simultaneously. They highlight that this approach can occur at different levels, from executing multiple instructions within a single processor (instruction-level parallelism) to the concurrent execution of independent processes on different processors in a shared system.

The primary purpose of parallel execution is to enhance performance and reduce computational execution time. Chapman, Jost e Pas (2007) point out that, regardless of the speed of modern computers, the demand for greater processing power continues to grow, driven by increasingly complex applications such as scientific modeling, large-scale data analysis, and high-fidelity simulations.

Parallel execution is not only about increasing speed but also about enabling tasks that would be impractical on a single processor due to time or memory constraints. Its main objectives include reducing execution time by distributing tasks across multiple processors, scaling computational problems to larger domains by utilizing additional memory and processing power, and improving energy efficiency by reducing power consumption through simultaneous task execution. In the context of High-Performance Computing (HPC), parallel computing is essential for achieving high performance in supercomputers and server clusters (CHAPMAN; JOST; PAS, 2007; ROBEY; ZAMORA, 2021).

In practice, within the context of associating Lagrangian meshes with structural meshes, a Lagrangian point may belong to one process while its topological neighbor is assigned to another. Although these elements exist in separate processes, the two triangular elements in the Lagrangian mesh share a common vertex. Therefore, any correction method applied to address failures in the orthogonal projection must ensure consistency across processes that do not directly communicate. In other words, if the vertex is moved, the correction must be performed in such a way that, even in independent processes, it is repositioned correctly to allow for proper orthogonal projection while preserving the topological continuity between vertices.

Based on this observation, the proposed solution for correcting cases where orthogonal projection is compromised consists of a mesh adjustment algorithm designed for parallel environments. The algorithm follows a structured sequence of steps to ensure the proper association of Lagrangian mesh vertices with structural elements.

Initially, all vertices that were not correctly associated—meaning they failed to find a corresponding structural element—are stored in a dedicated data structure. These vertices undergo an iterative adjustment process in which their positions are corrected using successive residuals. The core idea of the algorithm is to incrementally modify the vertex position until its orthogonal projection successfully intersects a structural element.

Residuals are added to the original vertex coordinates, and after each iteration, the

new positioning is tested to verify whether the orthogonal projection is successful. To maintain mesh integrity, two fundamental criteria must be met: (1) the adjustment must not result in a degenerate triangular element (i.e., with zero area), and (2) the total vertex displacement must not exceed 1 mm. Additionally, residuals are summed in a balanced manner across parallel processes, ensuring that the correction applied to the same vertex in different processes is identical, thereby preserving consistency and topological continuity in the mesh.

This concludes the development phase of the topological association between Lagrangian meshes, used in the immersed boundary method, and the structural mesh, used in the finite element method. The author acknowledges that some concepts and principles in this section may not be immediately clear upon first reading. To clarify the steps involved in the mesh association process, Fig. 12 is presented.

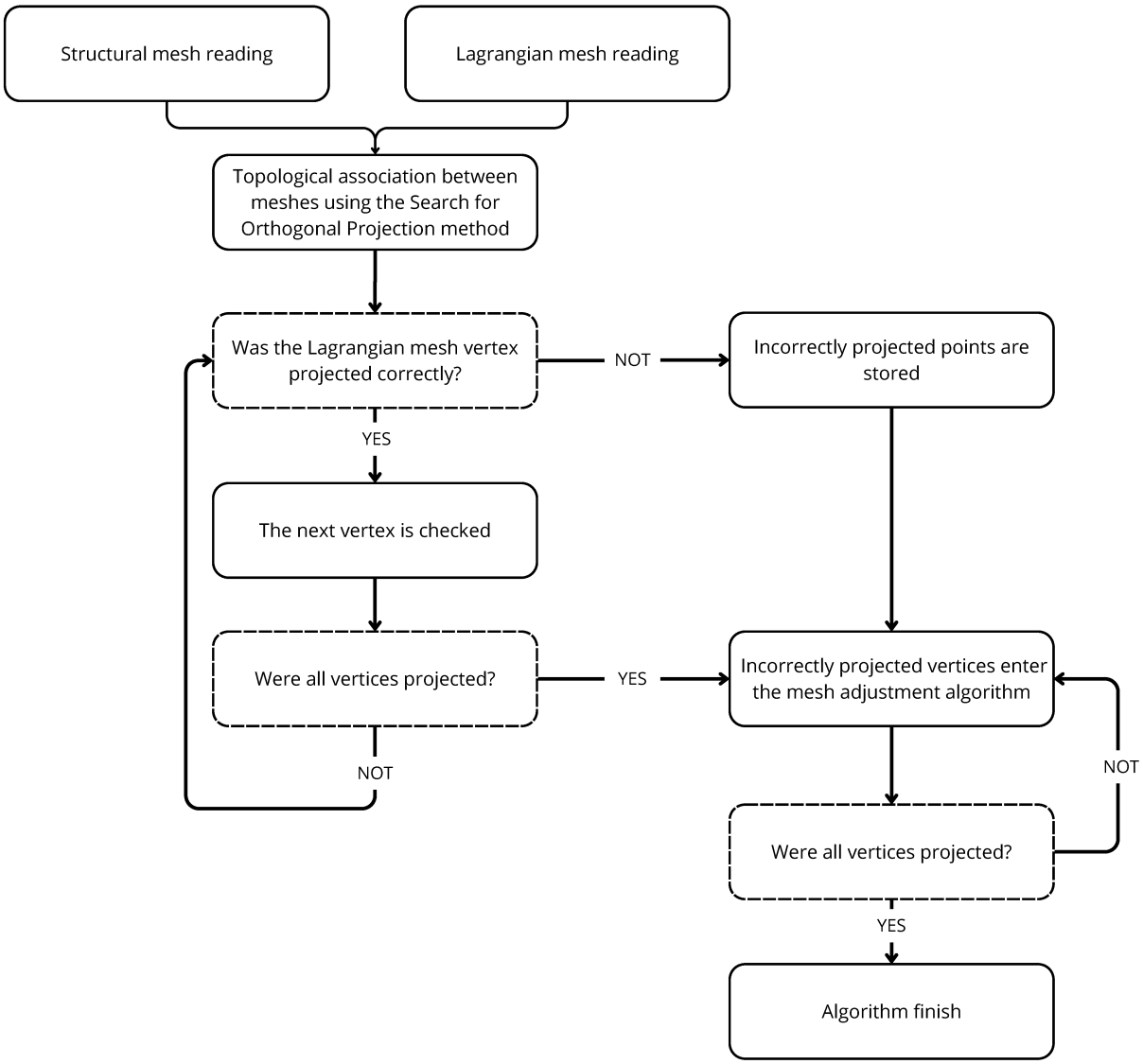


Figure 12 – Flowchart of the orthogonal projection verification and correction process.

4.2.2 Interpolation Methods

In the previous sections, the association between the Lagrangian and structural meshes was extensively discussed. This section addresses the importance of association algorithms, with an emphasis on correct orthogonal projection. Additionally, interpolation methods used in computational simulations of fluid-structure interaction are explored. Finally, the interpolation method developed to enable the simulation of this interaction in MFSim is presented, allowing the integration of finite element models imported from external sources.

With the mesh association already discussed, it is now possible to further detail the diagram presented in Fig. 4. This association enables an efficient flow of information between the fluid and structural domains, ensuring a topological connection between the mesh representing the immersed surface in the flow and the structural model mesh. As a result, the forces arising from the flow are correctly applied to the structural model, allowing for the determination of the system's dynamic response.

However, the mesh association alone does not transmit the forces calculated for each triangular element of the Lagrangian mesh, used in the immersed boundary method, to the elements of the structural mesh. In practice, its data structure only indicates which triangular elements of the Lagrangian mesh will serve as a reference for applying forces to the structural mesh elements. The next step, therefore, is to define the mechanism through which the structure's motion will be transmitted to the Lagrangian mesh, which, in turn, will communicate this movement to the flow.

For this purpose, interpolation functions are used. Fig. 13 illustrates the information flow based on this process. These functions are responsible for interpolating the motion of the structural mesh (displacements and velocities) onto the Lagrangian mesh, ensuring proper interaction between the fluid and structural models.

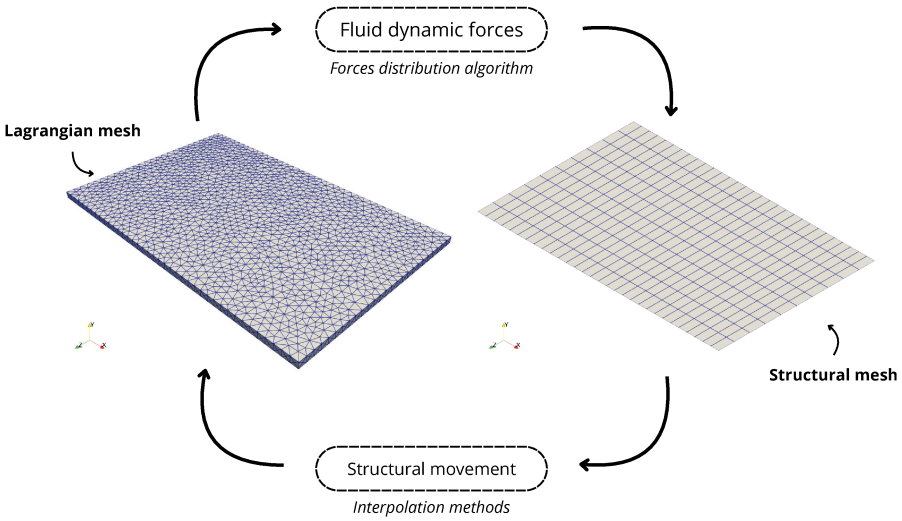


Figure 13 – Coupling scheme between the structural and Lagrangian meshes.

The displacements and velocities of the structure are obtained from the time integration algorithm of the system's equation of motion in the modal domain, as shown in section 3.4.3, Numerical Integration. In previous fluid–structure interaction studies conducted in MFSim, the structural motion was communicated to the Lagrangian mesh using shape functions (SOUZA, 2020).

A shape function is a mathematical function used in the Finite Element Method (FEM) to interpolate values within an element based on its nodal coordinates. It allows for the approximation of variables such as displacements, stresses, and other physical quantities along the element (COOK et al., 2001). Mathematically, the shape function is represented by a matrix associated with the element, known as the element shape function matrix, expressed as:

$$N_e(x) = p(x)(M_e)^{-1}, \tag{4.22}$$

where $N_e(x)$ contains the shape functions for element e (BATHE, 2016). These functions are fundamental in ensuring the continuity and accuracy of the numerical solution in structural and physical problems.

Each shape function is determined by interpolating between the element's nodes. For a two-node linear element, the shape functions $N_1(x)$ and $N_2(x)$ are expressed as:

$$N_1(x) = \frac{x_2 - x}{l_e}, \quad N_2(x) = \frac{x - x_1}{l_e}, \tag{4.23}$$

where l_e is the length of the element. These functions satisfy the Kronecker delta property, meaning they take a value of 1 at the corresponding node and 0 at the others, ensuring correct interpolation within the element (FISH, 2007).

For higher-order elements, such as quadratic ones, the shape functions are more complex polynomials based on Lagrange interpolants, providing greater accuracy in representing physical fields. In two-dimensional elements, such as triangles and quadrilaterals, the shape functions are determined based on local coordinates, such as barycentric or natural coordinates, allowing for proper interpolation of the physical properties along the element's surface (BATHE, 2016; FISH, 2007).

In the context of this work, the use of shape functions would represent a significant practical challenge. For each finite element theory imported into MFSim for fluid–structure interaction analysis, it would be necessary not only to know the structural element's shape function but also to implement it in the software. This challenge becomes even more complex when considering fluid–structure interaction analyses in finite element models that use multiple discretization theories, requiring the implementation and compatibility of different shape functions within the same computational environment.

To enable the use of structural models from external sources in the fluid–structure interaction analysis, the adopted approach was to use interpolation functions that leverage

the geometric shapes of the structural elements, thereby eliminating the need to know the shape functions associated with the specific structural element theory used.

In order to meet the requirements of the industrial case that motivates this work, the interpolation method must be robust and simple, since the simulation of fluid–structure interaction in similar cases exhibits a high computational cost. Therefore, the choice of interpolation must minimize its impact on simulation performance. Among the methods commonly used in this analysis, radial basis functions and kernel-type functions stand out, as presented below.

Radial Basis Functions (RBF) constitute a class of functions widely used in interpolation, numerical approximation, and machine learning. They depend exclusively on the distance between a point of interest and a set of predefined centers. According to (BUHMANN, 2003), RBFs are widely employed due to their flexibility and capacity to represent complex functions in high-dimensional spaces. The general form of these functions is given by:

$$\phi(|x - x_i|), \tag{4.24}$$

where ϕ is the radial basis function, x is the point at which the function is evaluated, and x_i are the centers of the radial basis.

Interpolation is a numerical method that seeks to estimate unknown values of a function based on a set of known points. In the case of RBFs, interpolation consists of finding a function $s(x)$ that satisfies:

$$s(x) = \sum_{i=1}^{N} \lambda_i \phi(|x - x_i|), \tag{4.25}$$

where the coefficients λ_i are determined by imposing that $s(x)$ assumes specific values at the known points. According to (BUHMANN, 2003), this method is efficient for the interpolation of multidimensional surfaces and is widely used in the modeling of sparse data.

Kernel-type interpolation is a generalization of interpolation and approximation methods based on kernel functions. According to (FASSHAUER; MCCOURT, 2015), kernel functions are employed to measure similarity between points and can be applied in least squares interpolation, regression, and machine learning. The general model can be expressed as:

$$s(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i), \tag{4.26}$$

where $K(x, x_i)$ is the kernel function, and the coefficients α_i are adjusted to minimize a specific error criterion.

The main difference between interpolation with RBFs and kernel-type interpolation lies in the way the functions are defined and used. While RBFs are explicitly formulated

as functions of distance, kernel functions can be more general, derived from statistical and machine learning principles (FASSHAUER; MCCOURT, 2015).

Despite the versatility and robustness of interpolation methods based on radial basis and kernel functions, to meet the objectives of this work, spatial interpolation methods were chosen. In these methods, the interpolation weights are determined based on the geometric characteristics of the structural elements, ensuring a more suitable approach to the needs of the simulation. The choice of this method is also due to the fact that its use implies a simple data structure in MFSim.

The interpolation methods implemented in MFSim were inspired by the barycentric method. The barycentric method is defined in the context of barycentric coordinates, which provide a way to express the position of a point within a simplex (such as a triangle in the plane or a tetrahedron in three-dimensional space) (HUGHES et al., 2014).

- *General Definition*: The barycentric coordinates of a point P with respect to a triangle ABC consist of three coefficients α, β, γ that satisfy the equation:

$$P = \alpha A + \beta B + \gamma C, \tag{4.27}$$

  with the condition that

$$\alpha + \beta + \gamma = 1. \tag{4.28}$$

  These coefficients indicate the contribution of each vertex in determining the position of point P (HUGHES et al., 2014).

- *Geometric Interpretation*: These coefficients can be seen as ratios of areas. For example, the coefficient α is proportional to the area of triangle PBC relative to the total area of triangle ABC:

$$\alpha = \frac{\text{Area}(\triangle PBC)}{\text{Area}(\triangle ABC)}. \tag{4.29}$$

  The same logic applies to the other two coefficients (LENGYEL, 2012).

- *Affine Invariance*: Barycentric coordinates remain invariant under affine transformations; that is, translations, rotations, and scaling preserve the barycentric relationships. This property is useful in computer graphics, mesh interpolation, and numerical methods (SCHNEIDER; EBERLY, 2003).

- *Generalization to a Simplex*: In higher dimensions, barycentric coordinates extend to any simplex, where a point can be expressed as an affine combination of the simplex's vertices (SCHNEIDER; EBERLY, 2003).

In summary, the barycentric method involves representing a point within a simplex using coefficients that sum to 1, and these coefficients can be interpreted as ratios of areas

or volumes relative to the simplex's geometric structure (HUGHES et al., 2014). Fig. 14 presents the geometric schematic of the barycentric method as an interpolation technique, which will be detailed below.

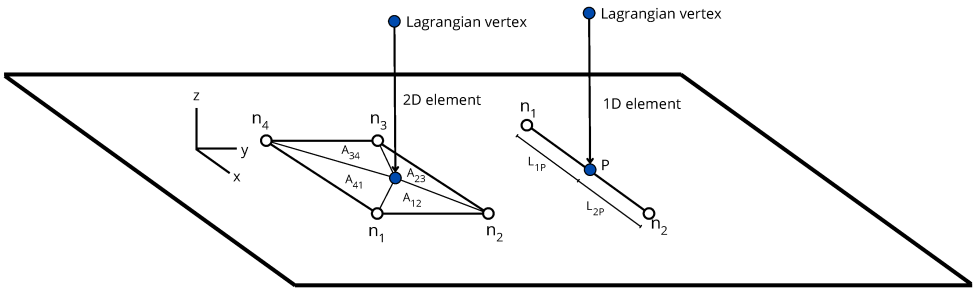


Figure 14 – Interpolation scheme using orthogonal projection and the barycentric method adapted for the geometry of 1D and 2D structural elements.

4.2.2.1 Barycentric Interpolation for 1D Structural Elements

Definition of the Coordinate System

Consider a 1D structural element (e.g., a beam or bar) in three-dimensional space, defined by two structural nodes n_1 and n_2, with position vectors:

$$\mathbf{x}_1 = (x_1, y_1, z_1), \quad \mathbf{x}_2 = (x_2, y_2, z_2). \tag{4.30}$$

For an arbitrary point P within the line segment connecting n_1 and n_2, its position vector is:

$$\mathbf{x}_P = (x_P, y_P, z_P). \tag{4.31}$$

Natural Coordinate Along the Segment We define the natural coordinate ξ along the line segment as:

$$\xi = \frac{\|\mathbf{x}_P - \mathbf{x}_1\|}{\|\mathbf{x}_2 - \mathbf{x}_1\|}, \tag{4.32}$$

where $\|\cdot\|$ denotes the Euclidean norm. The coordinate ξ takes values in the range $\xi \in [0, 1]$, where: - $\xi = 0$ corresponds to node n_1, - $\xi = 1$ corresponds to node n_2.

The barycentric weights are then defined as:

$$L_1^P = 1 - \xi, \quad L_2^P = \xi. \tag{4.33}$$

Interpolation of the Displacement Vectors

Let $\mathbf{u}_1 = (u_1, v_1, w_1)$ and $\mathbf{u}_2 = (u_2, v_2, w_2)$ be the nodal displacement vectors at nodes n_1 and n_2, respectively. The interpolated displacement at point P is given by:

$$\mathbf{u}(P) = L_1^P \mathbf{u}_1 + L_2^P \mathbf{u}_2. \tag{4.34}$$

Substituting the values of L_1^P and L_2^P:

$$\mathbf{u}(P) = (1 - \xi)\mathbf{u}_1 + \xi \mathbf{u}_2. \tag{4.35}$$

Expanding component-wise:

$$u(P) = (1 - \xi)u_1 + \xi u_2, \tag{4.36}$$

$$v(P) = (1 - \xi)v_1 + \xi v_2, \tag{4.37}$$

$$w(P) = (1 - \xi)w_1 + \xi w_2. \tag{4.38}$$

Thus, the interpolation is applied component-wise to each coordinate of the three-dimensional displacement vector.

Compact Vector Form

It is possible to express the result in matrix-vector form as:

$$\mathbf{u}(P) = (1 - \xi)\mathbf{u}_1 + \xi \mathbf{u}_2 = \mathbf{N}(\xi) \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}. \tag{4.39}$$

where $\mathbf{N}(\xi)$ is the linear shape function vector for a 1D bar element:

$$\mathbf{N}(\xi) = \begin{bmatrix} 1 - \xi & \xi \end{bmatrix}. \tag{4.40}$$

Thus, the displacement interpolation is an affine combination of the nodal displacements, weighted by the relative position of point P within the element.

4.2.2.2 Barycentric Interpolation for 2D Structural Elements

Definition of the Coordinate System

Consider a 2D quadrangular structural element in three-dimensional space, defined by four structural nodes n_1, n_2, n_3, n_4 at the corners of a quadrilateral. The position vectors of these nodes are:

$$\mathbf{x}_1 = (x_1, y_1, z_1), \quad \mathbf{x}_2 = (x_2, y_2, z_2), \tag{4.41}$$

$$\mathbf{x}_3 = (x_3, y_3, z_3), \quad \mathbf{x}_4 = (x_4, y_4, z_4). \tag{4.42}$$

For an arbitrary point P inside the quadrilateral, we determine its weight based on the areas formed between the nodes and the projected point P.

Shape Functions Based on Area Ratios

In this approach, the interpolation weights are determined by the relative areas of subregions formed between the point P and the nodes. Given a quadrilateral element, the point P forms four sub-triangles with the element nodes:

$$A_{1P} = \text{Area}(\triangle P n_2 n_3), \quad A_{2P} = \text{Area}(\triangle P n_3 n_4), \tag{4.43}$$

$$A_{3P} = \text{Area}(\triangle P n_4 n_1), \quad A_{4P} = \text{Area}(\triangle P n_1 n_2). \tag{4.44}$$

The total area of the quadrilateral is:

$$A_{\text{total}} = A_{1P} + A_{2P} + A_{3P} + A_{4P}. \tag{4.45}$$

The interpolation weights are defined as:

$$N_1 = \frac{A_{1P}}{A_{\text{total}}}, \quad N_2 = \frac{A_{2P}}{A_{\text{total}}}, \tag{4.46}$$

$$N_3 = \frac{A_{3P}}{A_{\text{total}}}, \quad N_4 = \frac{A_{4P}}{A_{\text{total}}}. \tag{4.47}$$

Interpolation of the Displacement Vector

Let $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4$ be the displacement vectors at the four nodes. The displacement at an arbitrary point P inside the element is interpolated as:

$$\mathbf{u}(P) = N_1\mathbf{u}_1 + N_2\mathbf{u}_2 + N_3\mathbf{u}_3 + N_4\mathbf{u}_4. \tag{4.48}$$

Expanding in component form:

$$u(P) = N_1u_1 + N_2u_2 + N_3u_3 + N_4u_4, \tag{4.49}$$

$$v(P) = N_1v_1 + N_2v_2 + N_3v_3 + N_4v_4, \tag{4.50}$$

$$w(P) = N_1w_1 + N_2w_2 + N_3w_3 + N_4w_4. \tag{4.51}$$

Compact Matrix Form

The interpolation can be rewritten in matrix form as:

$$\mathbf{u}(P) = \begin{bmatrix} N_1 & N_2 & N_3 & N_4 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{u}_4 \end{bmatrix}. \tag{4.52}$$

Thus, the displacement interpolation is a weighted combination of the nodal displacements, where the weights are determined by the relative areas.

Transition to the Triangular Element

To obtain the interpolation functions for a triangular element, we consider a degenerate case of the quadrilateral by collapsing one of its edges. This is done by aligning nodes n_3 and n_4 such that the quadrilateral becomes a triangle.

The point P now forms three sub-triangles with the element nodes:

$$A_{1P} = \text{Area}(\triangle P n_2 n_3), \quad A_{2P} = \text{Area}(\triangle P n_3 n_1), \tag{4.53}$$

$$A_{3P} = \text{Area}(\triangle P n_1 n_2). \tag{4.54}$$

The total area of the triangular element is:

$$A_{\text{total}} = A_{1P} + A_{2P} + A_{3P}. \tag{4.55}$$

The interpolation weights (barycentric coordinates) are then:

$$N_1 = \frac{A_{1P}}{A_{\text{total}}}, \quad N_2 = \frac{A_{2P}}{A_{\text{total}}}, \quad N_3 = \frac{A_{3P}}{A_{\text{total}}}. \tag{4.56}$$

Using these weights, the displacement interpolation follows:

$$\mathbf{u}(P) = N_1 \mathbf{u}_1 + N_2 \mathbf{u}_2 + N_3 \mathbf{u}_3. \tag{4.57}$$

This demonstrates that the triangular element is a special case of the quadrilateral element, where the interpolation remains consistent by weighting the displacement contributions by the sub-triangle areas.

The interpolation functions were derived only for the nodal displacement vector. To interpolate the nodal velocity vectors to the Lagrangian vertex, the same process can be followed.

4.3 Finite Element Method

This section discusses the development of the information flow between the finite element software Salome-Meca and the multiphysics simulation software MFSim. It presents how the structural system is computationally represented using the finite element method, with a focus on fluid-structure interaction simulation.

First, the advantages of importing finite element models from external sources rather than implementing them directly will be revisited. Then, the structural system representation will be examined based on the equation of motion and its associated variables (matrices and vectors). Finally, the necessary modeling steps for conducting the fluid-structure interaction simulation of the recovery boiler will be presented.

4.3.1 Purpose of Using External Models

The recovery boiler, the system for which this work aims to enable fluid-structure interaction simulation, requires the modeling of specific details. As discussed in Section 3.1, Description of the Industrial System, the boiler has boundary and operating conditions that must be incorporated into the structural model to ensure the most accurate representation of the physical system.

These requirements include:

- Temperature distribution along the boiler;
- Internal pressure within the tubes;
- Fluid mass inside the tubes.

Another important aspect is the spatial scale of the system. The current finite element model consists of approximately 8 million elements, discretized using a four-node quadrilateral mesh (QUAD4). In this project, or in future studies, to reduce the number of elements, the tubing could be discretized using beam elements, while the walls remain represented by planar elements. This approach would require modeling the connection at the interface between these two element types.

All these structural modeling requirements imposed by the boiler case lead to two possible approaches:

1. Extensive implementations of finite element models and their couplings, or
2. Enabling MFSim to use structural models from specialized finite element software.

This work adopts the second approach to extend fluid-structure interaction analysis in MFSim to new structural modeling techniques.

4.3.2 Structural System Representation

The structural system will be represented by the equation of motion, already presented in Section 3.3.2.2, Equation of Motion. Next, a representation that contrasts with the finite element method will be introduced using the equation of motion. In matrix form, the equation of motion is:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{Q} \tag{4.58}$$

Where:

$\mathbf{M}$ is the mass (inertia) matrix, representing the inertia of the system;

$\ddot{\mathbf{q}}$ is the acceleration vector, i.e., the second derivative of the displacement vector;

$\mathbf{C}$ is the damping matrix, which incorporates the dissipative forces in the system;

$\dot{\mathbf{q}}$ is the velocity vector, i.e., the first derivative of the displacement vector;

$\mathbf{K}$ is the stiffness matrix, representing the structural rigidity of the system;

$\mathbf{q}$ is the displacement vector, indicating the deformations of the system;

$\mathbf{Q}(t)$ is the external force vector, which in this work represents the fluid-dynamic forces that may vary with time t.

It was shown in Section 3.3.2.3, Modal Analysis, that the natural frequencies of the system and its corresponding vibration modes can be obtained from the $\mathbf{M}$ and $\mathbf{K}$ matrices.

From a structural point of view, in order to perform the fluid-structure interaction simulation, the steps shown in the flowchart in Fig. 15 must be followed.

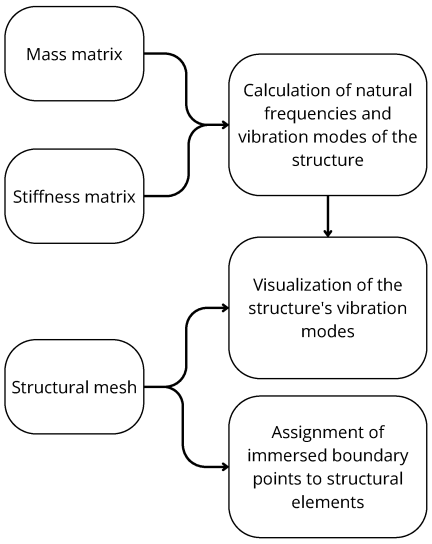


Figure 15 – Flowchart of the structural vibration analysis process, including the calculation of natural frequencies, vibration modes, and the assignment of immersed boundary points in MFSim.

Thus, the necessary data for the structural model are the finite element mesh and the mass and stiffness matrices, which are essential for modal analysis, a preliminary step for the fluid-structure interaction analysis. Based on this reasoning, the flowchart is defined, outlining the information to be extracted from the Salome-Meca software to be imported into MFSim.

The MFSim will work with the following options for importing structural model information:

1. The mass matrix $\mathbf{M}$ and stiffness matrix $\mathbf{K}$ will be exported from the Salome-Meca software, along with the finite element mesh. With this set of information, it is possible to perform both modal analysis and fluid-structure interaction analysis. All modeling details, including boundary conditions and loading, must be included in the system matrices;

2. The mass matrix $\mathbf{M}$ and stiffness matrix $\mathbf{K}$ will be exported from the Salome-Meca software, along with the vibration mode matrix $\mathbf{\Phi}$ (eigenvectors) and the natural frequencies associated with the vibration modes. When this option is chosen, MFSim will not be used as a modal solver, which can be an advantage for cases referred to in this work as large number of degrees of freedom. The computational cost associated with modal analysis of such models will be discussed later;

3. In MFSim, the time integration is performed in the modal domain. In this domain, the set of uncoupled second-order differential equations (ODEs) are solved separately, subject to initial displacement and velocity conditions. For instance, it is necessary to consider that a boiler, when operating, experiences tube expansion due to internal pressure, just as the structure is deformed under its own weight. To address this case, and because MFSim has nodal displacements related to preloadings, the nodal displacements or the vector of static nodal loadings will be exported from Salome-Meca and imported into MFSim.

With this approach, MFSim can work with external finite element models, incorporating information from both the left-hand side (LHS) of the equation of motion (system properties) and the right-hand side (RHS) (external information, more specifically initial loading).

4.3.3 Types of Structural Elements

As part of the development of this work, structural and fluid-structure interaction simulations will be conducted to validate the methods developed here. Two finite element models were selected for the tests: a beam model and a planar element model, whose theories are presented below.

4.3.3.1 Beam element theory

Introduction

The POU_D_T element is a finite element formulation for beam modeling based on Timoshenko's beam theory. Unlike Euler-Bernoulli beams, this formulation accounts for transverse shear deformation, making it suitable for beams with lower slenderness ratios where shear effects cannot be neglected (FLÉJOU, 2017).

Governing Equations

The behavior of the POU_D_T element is governed by the fundamental equations of elasticity and structural mechanics. The displacement field considers both bending and shear deformations. The primary equations include (FLÉJOU, 2017):

1. **Axial Equilibrium Equation:**

$$\frac{dN}{dx} + f_{\text{ext}} = \rho S \frac{d^2u}{dt^2} \tag{4.59}$$

   where N is the normal force, f_{ext} is the external force per unit length, ρ is the material density, S is the cross-sectional area, and u is the axial displacement.

2. **Shear and Bending Equations:**

$$\frac{dV_z}{dx} + t_z^{\text{ext}} = \rho S \frac{d^2w}{dt^2} \tag{4.60}$$

$$\frac{dM_y}{dx} - V_z + m_y^{\text{ext}} = \rho I_y \frac{d^2\theta_y}{dt^2} \tag{4.61}$$

   where V_z is the shear force, t_z^{ext} is the external shear loading, M_y is the bending moment, m_y^{ext} is the external moment, I_y is the second moment of area, and θ_y is the rotation due to bending.

3. **Constitutive Relations:**

$$M_y = EI_y \frac{d\theta_y}{dx} \tag{4.62}$$

$$V_z = k_z SG \left(\frac{dw}{dx} + \theta_y \right) \tag{4.63}$$

   where E is Young's modulus, G is the shear modulus, and k_z is a shear correction factor.

Stiffness and Mass Matrices

The stiffness and mass matrices of the POU_D_T element are obtained from the discretization of the governing equations and integrated over the element length (FLÉJOU, 2017). The general forms of these matrices are:

$$\mathbf{K} = \frac{EI_y}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \tag{4.64}$$

$$\mathbf{M} = \rho S \begin{bmatrix} \frac{13L}{35} & \frac{11L^2}{210} & \frac{9L}{70} & -\frac{13L^2}{420} \\ \frac{11L^2}{210} & \frac{L^3}{105} & \frac{3L^2}{40} & -\frac{L^3}{140} \\ \frac{9L}{70} & \frac{3L^2}{40} & \frac{13L}{35} & -\frac{11L^2}{210} \\ -\frac{13L^2}{420} & -\frac{L^3}{140} & -\frac{11L^2}{210} & \frac{L^3}{105} \end{bmatrix} \tag{4.65}$$

These matrices account for both translational and rotational degrees of freedom, including effects of transverse shear deformation (FLÉJOU, 2017).

Finite Element Discretization

The POU_D_T element is represented as a two-node beam element with six degrees of freedom per node: three translations (u, v, w) and three rotations $(\theta_x, \theta_y, \theta_z)$. The stiffness and mass matrices are derived from the governing equations and integrated over the element length (FLÉJOU, 2017).

Applications and Considerations

This formulation is particularly suited for problems where transverse shear deformation is significant, such as short beams or composite structures. For long and slender beams, the Euler-Bernoulli beam formulation (POU_D_E) may be more appropriate (FLÉJOU, 2017).

For further mathematical details and derivations, refer to the Code_Aster documentation on exact beam elements (FLÉJOU, 2017).

4.3.3.2 Plate element theory

Introduction

The DKT element (Discrete Kirchhoff Triangle) is a triangular finite element formulation for plate bending problems. It is based on the Kirchhoff-Love theory, which assumes that plane sections remain plane and perpendicular to the mid-surface of the plate

after deformation. The DKT element is particularly useful for thin plates where shear deformations are negligible (SOZA, 2016).

For cases where the mesh consists of quadrilateral elements, an alternative formulation known as the DKQ (Discrete Kirchhoff Quadrilateral) element is used. The DKQ element extends the Kirchhoff-Love plate theory to quadrilateral elements, maintaining compatibility with the DKT formulation while allowing structured meshing strategies (SOZA, 2016).

Governing Equations

The fundamental equations governing the behavior of both DKT and DKQ elements are derived from classical plate bending theory. The equilibrium equations in terms of internal forces are given by (SOZA, 2016):

1. **Membrane Equilibrium Equations:**

$$\frac{\partial N_{xx}}{\partial x} + \frac{\partial N_{xy}}{\partial y} + f_x^{\text{ext}} = \rho h \frac{\partial^2 u}{\partial t^2} \tag{4.66}$$

$$\frac{\partial N_{yy}}{\partial y} + \frac{\partial N_{xy}}{\partial x} + f_y^{\text{ext}} = \rho h \frac{\partial^2 v}{\partial t^2} \tag{4.67}$$

   where N_{xx}, N_{yy}, N_{xy} are the membrane forces per unit width, $f_x^{\text{ext}}, f_y^{\text{ext}}$ are external forces per unit area, ρ is the material density, h is the plate thickness, and u, v are in-plane displacements.

2. **Bending Equilibrium Equations:**

$$\frac{\partial^2 M_{xx}}{\partial x^2} + 2\frac{\partial^2 M_{xy}}{\partial x \partial y} + \frac{\partial^2 M_{yy}}{\partial y^2} - q = \rho h \frac{\partial^2 w}{\partial t^2} \tag{4.68}$$

   where M_{xx}, M_{yy}, M_{xy} are the bending moments, q is the transverse distributed load, and w is the transverse displacement.

3. **Constitutive Relations:**

$$\begin{bmatrix} M_{xx} \\ M_{yy} \\ M_{xy} \end{bmatrix} = D \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \begin{bmatrix} \kappa_{xx} \\ \kappa_{yy} \\ \kappa_{xy} \end{bmatrix} \tag{4.69}$$

   where $D = \frac{Eh^3}{12(1-\nu^2)}$ is the flexural rigidity, ν is Poisson's ratio, and $\kappa_{xx}, \kappa_{yy}, \kappa_{xy}$ are the curvatures.

Finite Element Discretization

The DKT element is a three-node triangular element with three degrees of freedom per node: two in-plane translations (u, v) and one transverse displacement (w). The DKQ element, on the other hand, is a four-node quadrilateral element that follows the same displacement assumptions but is constructed to ensure compatibility with structured meshes (SOZA, 2016).

Stiffness Matrix

The stiffness matrices of both DKT and DKQ elements are obtained by integrating the strain energy over the element area. Their general form is:

$$\mathbf{K} = \int_S \mathbf{B}^T \mathbf{D} \mathbf{B} \, dS \tag{4.70}$$

where $\mathbf{B}$ is the strain-displacement matrix and $\mathbf{D}$ is the constitutive matrix. The formulation for DKQ adapts this approach to quadrilateral elements while maintaining Kirchhoff's constraint (SOZA, 2016).

Mass Matrix

The mass matrices for DKT and DKQ elements follow the same approach:

$$\mathbf{M} = \rho h \int_S \mathbf{N}^T \mathbf{N} \, dS \tag{4.71}$$

where $\mathbf{N}$ is the shape function matrix. Alternatively, a lumped mass approach can be used for explicit dynamics simulations (SOZA, 2016).

Applications and Considerations

The DKT element is well-suited for unstructured triangular meshes, commonly used in irregular geometries. The DKQ element, on the other hand, is used when structured quadrilateral meshes are preferred, providing better numerical accuracy for regularly shaped domains (SOZA, 2016).

For thick plates where transverse shear deformation is significant, alternative formulations like Mindlin-Reissner plate theory (DST or DSQ elements) should be considered.

For further mathematical details and derivations, refer to the Code_Aster documentation on plate elements (SOZA, 2016).

4.3.3.3 Boundary Conditions with Discrete Elements

Introduction

In finite element modeling, discrete elements are often used to impose boundary conditions or represent local stiffness and mass effects at specific points. The elements K_TR_D_N and M_TR_D_N in Code_Aster serve this purpose, allowing for the definition of stiffness and mass properties at nodes, which can be useful for modeling supports, constraints, and added mass effects (FLÉJOU, 2015).

Kinematic Constraints with K_TR_D_N

The K_TR_D_N element is a discrete stiffness element applied to a node (POI1 type mesh). It defines a local stiffness matrix to enforce constraints on translational and rotational degrees of freedom. The stiffness matrix is given by (FLÉJOU, 2015):

$$\mathbf{K} = \begin{bmatrix} k_x & 0 & 0 & 0 & 0 & 0 \\ 0 & k_y & 0 & 0 & 0 & 0 \\ 0 & 0 & k_z & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{rx} & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{ry} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{rz} \end{bmatrix} \tag{4.72}$$

where:

- k_x, k_y, k_z are translational stiffness coefficients;
- k_{rx}, k_{ry}, k_{rz} are rotational stiffness coefficients.

These values can be adjusted to represent elastic supports or to impose full constraints (setting them to very high values) (FLÉJOU, 2015). To ensure the stability of linear solvers, finite element software generally uses values on the order of 10^9 N/m for translational stiffness and 10^9 Nm for rotational stiffness when representing full constraints.

Added Mass with M_TR_D_N

The M_TR_D_N element defines a discrete mass matrix at a node, allowing for the inclusion of concentrated masses and rotational inertias. This is particularly relevant when modeling added mass effects due to fluid-structure interaction or localized inertia contributions. The mass matrix is defined as (FLÉJOU, 2015):

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 & 0 & me_z & -me_y \\ 0 & m & 0 & -me_z & 0 & me_x \\ 0 & 0 & m & me_y & -me_x & 0 \\ 0 & -me_z & me_y & V_{xx} & V_{xy} & V_{xz} \\ me_z & 0 & -me_x & V_{xy} & V_{yy} & V_{yz} \\ -me_y & me_x & 0 & V_{xz} & V_{yz} & V_{zz} \end{bmatrix} \tag{4.73}$$

where:

- m is the added mass;
- x_g, y_g, z_g are the coordinates of the mass center relative to the node;
- I_{xx}, I_{yy}, I_{zz} are the principal moments of inertia;
- I_{xy}, I_{yz}, I_{xz} are the products of inertia.

The elements of the submatrix V are defined as:

$$V_{xx} = I_{xx} + m(e_y^2 + e_z^2) \tag{4.74}$$
$$V_{yy} = I_{yy} + m(e_x^2 + e_z^2) \tag{4.75}$$
$$V_{zz} = I_{zz} + m(e_x^2 + e_y^2) \tag{4.76}$$
$$V_{xy} = I_{xy} - me_xe_y \tag{4.77}$$
$$V_{yz} = I_{yz} - me_ye_z \tag{4.78}$$
$$V_{xz} = I_{xz} - me_xe_z \tag{4.79}$$

This formulation ensures that both translational and rotational inertia effects are accounted for, making it suitable for applications such as rigid body dynamics and floating structures.

Applications and Considerations

The use of discrete elements for boundary conditions and added mass is crucial in the accurate simulation of structural behavior. The K_TR_D_N element allows for the modeling of flexible supports, while the M_TR_D_N element facilitates the inclusion of localized mass effects. These elements are particularly useful in:

- Fluid-structure interaction simulations, where added mass effects are significant;
- Structural dynamics problems requiring precise boundary conditions;
- Modeling of supports with predefined stiffness characteristics.

For more details on the implementation and parameterization of these elements, refer to the Code_Aster documentation (FLÉJOU, 2015).

4.3.3.4 Pre-stress Modeling and Geometric Stiffness Matrix

Introduction

The geometric stiffness matrix for plate elements is derived from the formulation of linear buckling, where the equilibrium equations are adjusted to include the effects of pre-existing normal stresses. This adjustment introduces the geometric stiffness matrix $\mathbf{K_G}$, which complements the conventional elastic stiffness matrix $\mathbf{K_E}$ in the stability problem equation:

$$(\mathbf{K_E} + \lambda \mathbf{K_G})\mathbf{\Phi} = 0 \tag{4.80}$$

where λ is the critical load factor and $\mathbf{\Phi}$ represents the buckling modes of the structure (MCGUIRE; ZIEMIAN, 2014).

Theoretical Formulation

The geometric stiffness matrix is obtained from the second variation of the internal strain energy, considering the nonlinear terms of the constitutive equations. For a thin plate modeled according to Kirchhoff-Love theory, the strain field can be expressed as:

$$\varepsilon = e_L + e_{NL} + z\kappa \tag{4.81}$$

where:

- e_L represents the linear membrane strains;
- e_{NL} represents the nonlinear membrane strains;
- κ represents the linear bending strains (OKABE, 2020).

From these equations, the geometric stiffness matrix can be expressed as an integral over the plate surface:

$$\mathbf{K_G} = \int_S \begin{bmatrix} \delta w_{,x} & \delta w_{,y} \end{bmatrix} \begin{bmatrix} N_{xx} & N_{xy} \\ N_{xy} & N_{yy} \end{bmatrix} \begin{bmatrix} \delta w_{,x} \\ \delta w_{,y} \end{bmatrix} dS \tag{4.82}$$

where w is the transverse displacement and N_{xx}, N_{yy}, N_{xy} are the normal forces in the plate due to pre-stress (MCGUIRE; ZIEMIAN, 2014).

Pre-Stress Modeling and Application

The inclusion of the geometric stiffness matrix allows the consideration of pre-stress effects in structural calculations, such as buckling and vibration analysis. The adopted methodology involves activating the RIGI_MECA_GE option in the element catalog, which enables the assembly of both elastic and geometric stiffness matrices (MCGUIRE; ZIEMIAN, 2014).

For accurate modeling, the initial stresses in the structure are extracted from an initial state calculation (e.g., a static computation) and then used in the assembly of the geometric stiffness matrix. This is crucial for structures subject to residual stresses or high initial loads, such as pressurized tanks or aerospace components (OKABE, 2020).

The theory of the geometric stiffness matrix and its influence on the total stiffness of a structure are important aspects of structural analysis. The geometric stiffness matrix depends on both the geometry of the structure and the applied loading, making it essential to analyze each case individually. This approach allows for an accurate assessment of how this matrix affects the system's dynamic response, ensuring proper modeling.

4.3.4 Thermal Effects

In this section, the modeling of thermal effects on the structural system will be presented, assuming that the structure is not subjected to conditions where thermal loading and boundary constraints generate pre-stresses. The objective is to define an approach that represents these effects through the modification of the material's Young's modulus, considering it as a function of temperature.

The Young's modulus (E) is a critical parameter in the analysis of stress and strain. However, in structures subjected to temperature variations, especially under extreme conditions such as fires, the Young's modulus can undergo significant reductions, altering the material's mechanical behavior (SEIF et al., 2016).

Variation of Young's Modulus with Temperature

According to Seif et al. (2016), the Young's modulus of structural steels decreases progressively as the temperature increases. This degradation occurs due to the activation of atomic diffusion mechanisms and changes in the interatomic bonds of the material. The empirical relationship between $E(T)$ and temperature T can be expressed by an exponential function fitted to experimental data:

$$E(T) = E_0 \left(1 - Ae^{-BT}\right) \tag{4.83}$$

where:

- E_0 is the Young's modulus at ambient temperature;
- A and B are parameters fitted experimentally.

Experimental data indicate that, for conventional structural steels, the Young's modulus can be reduced to approximately 50% of its original value when the temperature reaches around 600°C, and to less than 10% for temperatures close to 1000°C.

Influence on Structural Analysis

The variation of Young's modulus directly impacts the stiffness and stability of structures. At elevated temperatures, the following changes are observed:

- Reduction in load capacity: Due to the decrease in stiffness, the structure may exhibit greater deformation under the same level of load.
- Increased risk of buckling: The degradation of Young's modulus contributes to the reduction of resistance to buckling collapse in slender elements.
- Alteration of natural frequencies: The dynamic response of the structure is affected, as the natural frequencies of vibrational modes decrease.

The curve obtained from the technical report can be seen in Fig. 16. The curve shown in black is obtained from Eq. 2.2 in the technical report. This equation describes the variation of Young's modulus $E(T)$ as a function of temperature. It is formulated to capture the temperature-dependent behavior of structural steel:

$$E(T) = E_0 \exp\left[-\frac{1}{2}\left(\frac{\Delta T}{e_3}\right)^{e_1} - \frac{1}{2}\left(\frac{\Delta T}{e_4}\right)^{e_2}\right] \tag{4.84}$$

where:

- $E(T)$: Young's modulus at temperature T,
- E_0: Young's modulus at ambient temperature (T_0), given as 206 GPa,
- $\Delta T = T - T_0$: Temperature difference relative to the ambient temperature,
- e_1, e_2, e_3, e_4: Empirical coefficients that depend on the type of steel.

The equation uses two exponential terms to model the degradation of Young's modulus as temperature increases. Each term has denominators e_3 and e_4, which represent reference temperatures associated with different physical degradation mechanisms. The exponents e_1 and e_2 control the rate of reduction of the modulus with temperature.

This equation predicts an exponential decrease in Young's modulus, reflecting the progressive weakening of the material due to thermal activation and microstructural effects.

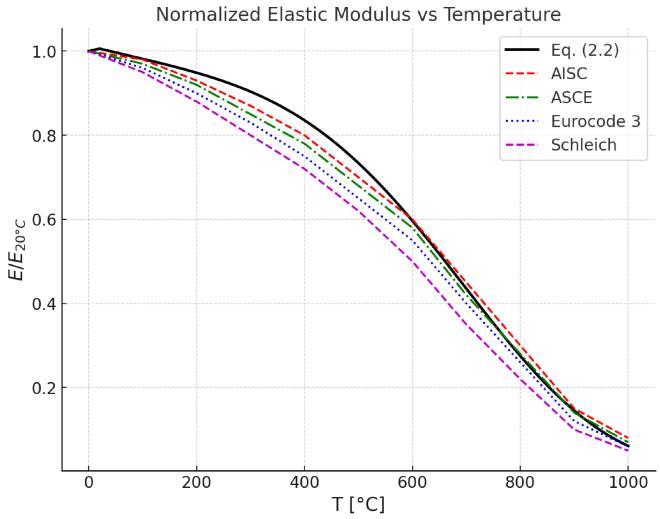


Figure 16 – Variation of Young's modulus as a function of temperature. Curve adapted from Seif et al. (2016).

Seif et al. (2016) compares its model of Young's modulus degradation with normative models, such as Eurocode 3 (CEN, 2005). The results show that the NIST's prediction for the degradation of Young's modulus is in good agreement with the values recommended by European and international standards, while allowing for a more detailed modeling based on different types of steel and loading histories.

In summary, to incorporate the effects of thermal loads into the structural model, it is necessary to determine the experimental curve of Young's modulus as a function of temperature. With this information, the effect of thermal loading can be properly applied to the structural model.

4.3.5 Coupling Between Elements of Different Theories

The coupling between finite elements from different theories, such as beam and plate, is a viable approach for representing mixed structures. The transition between these theories must ensure kinematic and static compatibility, guaranteeing that forces and moments are correctly transmitted (FLÉJOU, 2015).

Among the methods used to couple beam and plate elements, the following stand out:

RBE3 (Rigid Body Element Type 3)

RBE3 is a widely used method in structural analysis to connect a master node to multiple slave nodes. Unlike RBE2 (which imposes total rigidity), RBE3 distributes loads and displacements in a weighted manner, without completely restricting the degrees of freedom of the slave nodes (CODE_ASTER, 2015). This method is useful for modeling the coupling of a beam to a plate by distributing forces and moments across multiple points on the plate surface.

Characteristics of RBE3:

- Maintains compatible displacements and rotations without making the structure excessively rigid;
- Distributes loads proportionally to the weights assigned to the slave nodes;
- Does not add stiffness to the connection, it only imposes a kinematic relation between the degrees of freedom;
- Applicable for coupling elements with different degrees of freedom.

LIAISON_SOLIDE (Rigid Connection)

LIAISON_SOLIDE is a coupling method implemented in Code_Aster that imposes total rigidity between connected elements. It ensures that the nodes of the beam and plate move rigidly as a single body, without relative deformation (CODE_ASTER, 2016).

Characteristics of LIAISON_SOLIDE:

- All nodes at the interface share the same displacements and rotations;
- Does not allow flexibility in the connection, which may generate artificial stresses;
- Suitable for situations where the connection between elements is effectively rigid, such as welds or fixed joints.

LIAISON_ELEM (Element Coupling)

LIAISON_ELEM allows the connection of different elements by imposing either partial or complete kinematic compatibility, depending on the chosen configuration. For beam-to-plate coupling, the COQ_POU - coupling between beam and plate elements in French - option can be used to ensure the correct transmission of forces and moments between elements (CODE_ASTER, 2016).

Characteristics of LIAISON_ELEM:

- Allows modeling transitions between different finite element formulations;
- Can be adjusted to ensure continuity in displacements and rotations;
- Suitable for representing flexible or semi-rigid connections.

The choice of coupling method depends on the application and the required level of fidelity in the modeling. RBE3 is useful for distributing loads without excessively stiffening the connection (CODE_ASTER, 2015), while LIAISON_SOLIDE is suitable for completely rigid connections (CODE_ASTER, 2016). LIAISON_ELEM, on the other

hand, provides an intermediate approach, allowing adjustments as needed and enabling direct coupling between different finite element theories.

4.4 Software Coupling and Data Transfer

This section discusses the coupling of MFSim with finite element software, such as Salome-Meca, from a data flow perspective. The objective is to determine the necessary input data for MFSim to enable fluid-structure interaction analysis in industrial applications.

To achieve this, the chosen approach is to first provide a general overview of the essential information required in the structural model to be imported into MFSim, followed by a detailed discussion of each step.

Figure 17 presents the diagram illustrating how the finite element model is created and exported from the Salome-Meca software.

The first step is to discretize the geometry into a finite element mesh. In this step, the type of discretization and the order of the element (linear, quadratic, or cubic) are defined. Once the mesh is completed, it must be exported from Salome-Meca in a '.dat' text format to be imported into MFSim. A mesh-reading algorithm compatible with this format was developed in MFSim.

The second step consists of selecting the finite element theory, defining the material that composes the geometry, and modeling the boundary conditions and structural loading.

In the third step, the finite element software assembles the system's mass and stiffness matrices, which will then be imported into MFSim. If the structure is subjected to loads that generate pre-stress, a static analysis must first be performed, considering this loading. The stress field along the structure is obtained, and then the geometric stiffness matrix is created. The geometric stiffness matrix is added to the system's stiffness matrix, resulting in the final system stiffness matrix.

The fourth step involves exporting the mass and stiffness matrices from Salome-Meca. Initially, during the early stages of this study, the matrices were saved in dense format as text files. For cases with fewer than 10,000 degrees of freedom, the storage space required by these files was not a major concern. However, as the need to analyze cases with an increasing number of degrees of freedom emerged, it was decided to continue working with text-format matrices but now using the sparse matrix format. This approach provided a significant gain, but industrial cases still had a very large number of degrees of freedom, requiring an alternative solution.

Notably, in preliminary structural models of the boiler, each matrix in sparse format occupied around 100 GB of storage. The solution was to create a binary file format that standardized the matrices using the COO (Coordinate List) format. This reduced the storage space required per matrix from 100 GB to less than 40 GB.

The values of the system matrices are exported using 16 decimal places of precision, ensuring that the model maintains its representativity upon export without numerical precision loss.

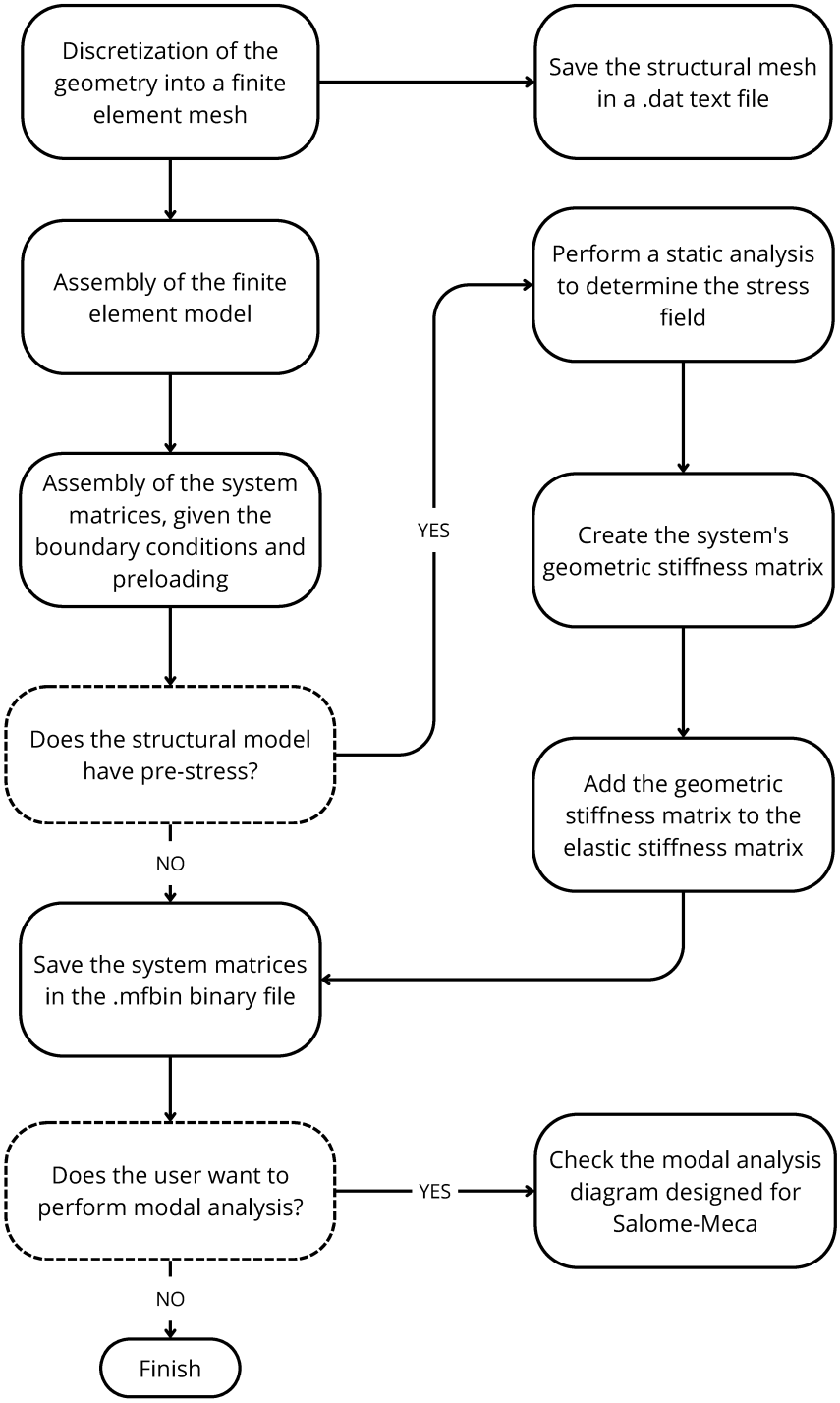


Figure 17 – Diagram illustrating how the finite element model is created and exported from Salome-Meca.

If the user does not wish to perform the modal analysis using the modal solvers of Salome-Meca, the flowchart is finalized.

Some important points should be highlighted. The static analysis step in Salome-Meca

is not required only in cases where pre-stress is present. It must also be performed when the structure is subjected to initial loads that will be imposed as initial conditions in the fluid-structure interaction analysis, such as deformation due to self-weight, added mass, and similar cases.

It is important to emphasize that the purpose of the static analysis in these cases is to obtain the nodal displacement field. For these scenarios, the geometric stiffness matrix is not created, as there is no pre-stress loading.

If the user wishes to perform the modal analysis in Salome-Meca, Fig. 18 illustrates the steps that must be followed. To carry out the modal analysis, the modal solver must be configured, specifying the mathematical model to be used in the solver and selecting the algorithm for determining the vibration modes and natural frequencies of the system. More details on this topic can be found in Code Aster (2024).

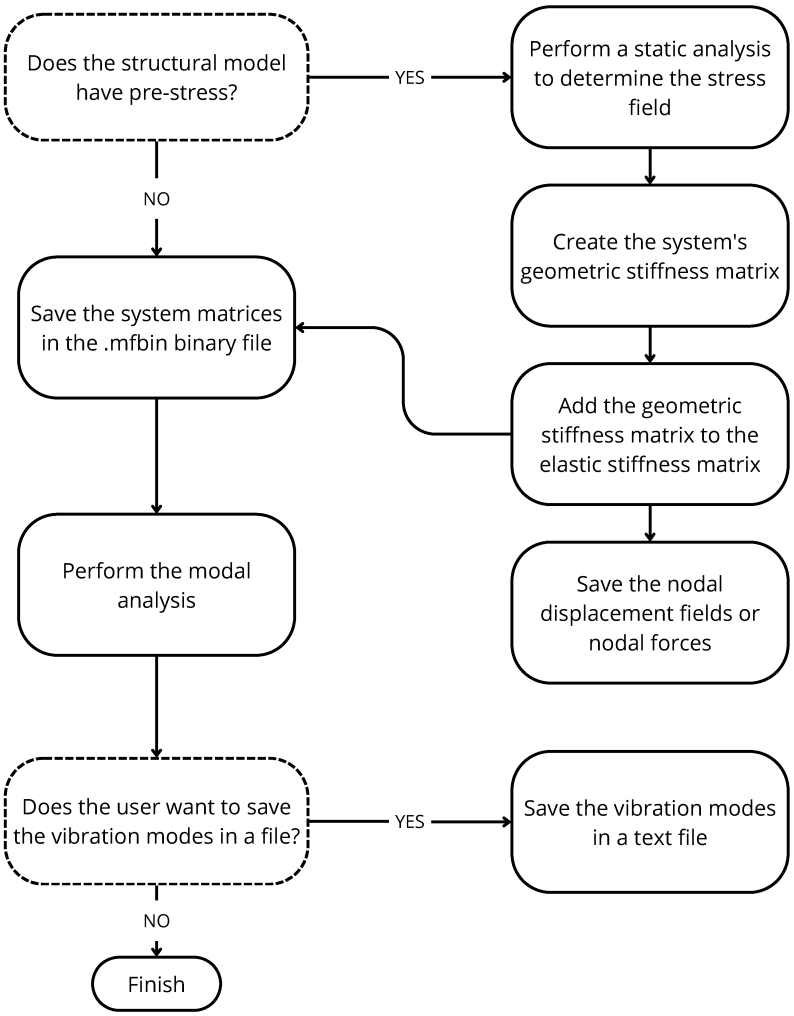


Figure 18 – Diagram illustrating how the modal analysis is performed within the Salome-Meca software.

The user also has the option to save the structure's vibration modes and natural frequencies for later import into MFSim. This approach can represent a significant

reduction in computational cost for certain cases, as it eliminates the need to use the MFSim modal solver.

Finally, the process of importing the structural model into MFSim is presented. Figure 19 illustrates the steps in which the structural model is imported and utilized in the software.

In the first step, the structural mesh is imported into the software. Immediately afterward, the association between the structural mesh and the Lagrangian mesh is performed.

Next, in the third step, the system matrices are imported into MFSim to enable the modal analysis. Once this is completed, the vibration modes calculated by the MFSim modal solver are saved in files for post-processing.

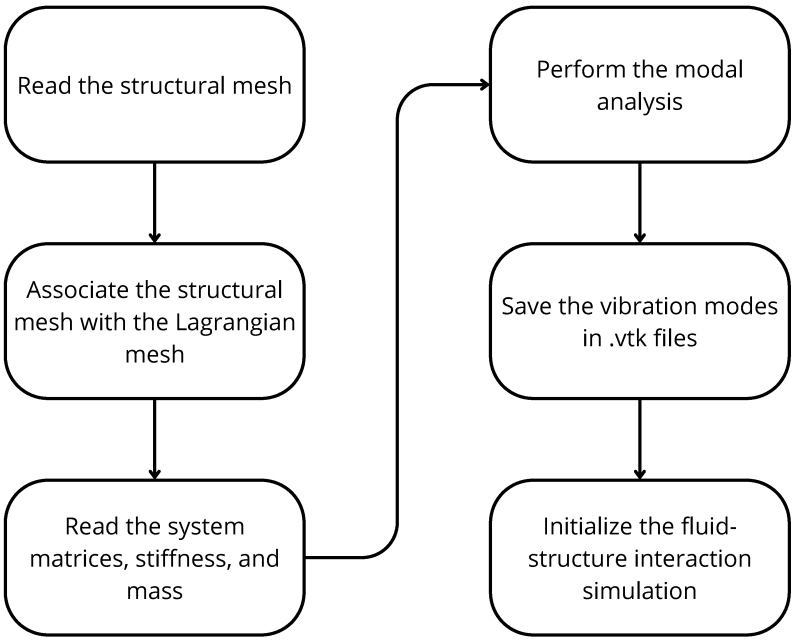


Figure 19 – Diagram illustrating how the external structural model is used in MFSim.

It is important to highlight that if the structure is subjected to initial loads, the nodal displacement fields are imported into MFSim and set as initial conditions before starting the fluid-structure interaction analysis.

5 Validations cases

This chapter discusses the validation cases of the numerical and computational methods developed in this work. All the numerical and computational tools presented in Chapter 4, Numerical and Computational Modeling, will be covered in this chapter.

5.1 Validation of the interpolation functions

This section presents the validation of the mesh association algorithm and the implemented interpolation methods. To achieve this, the following analysis is performed: a static force is applied to the pipe geometry found in heat exchanger systems, as shown in Fig. 6. In the first case, a static load of 100 N is applied in the negative z-axis direction, as illustrated in Fig. 20.

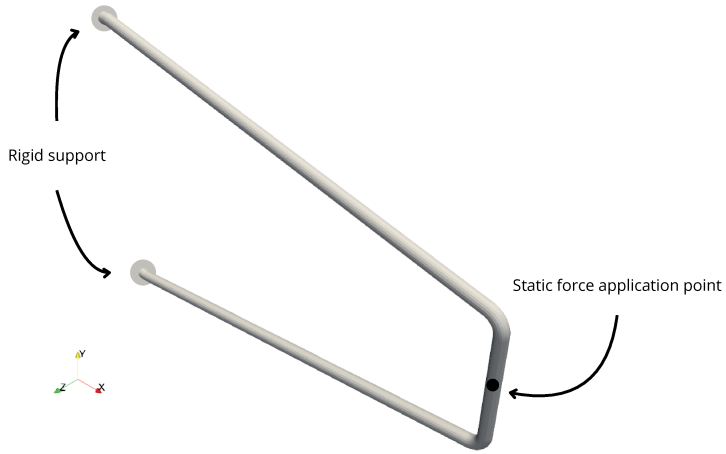
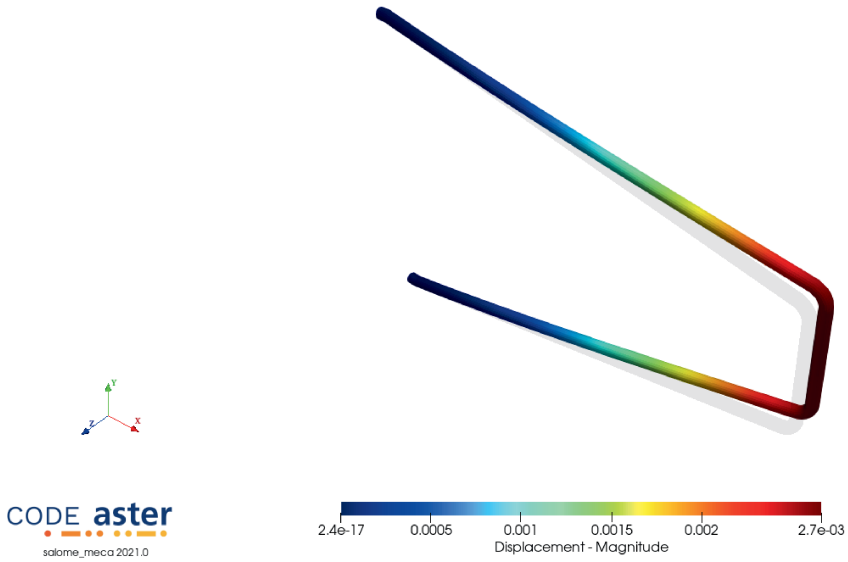


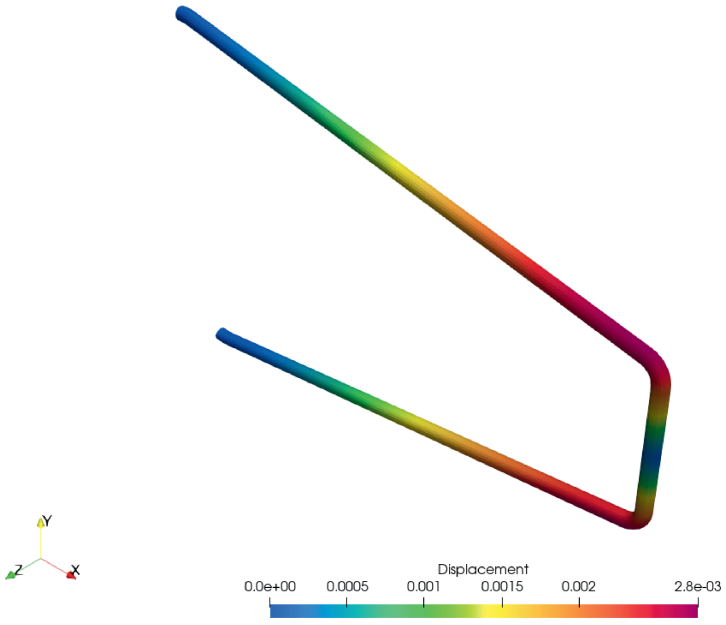
Figure 20 – Boundary conditions and local application point of the static load.

The geometry was discretized in Salome-Meca using the DKQ finite element model, Discrete Kirchhoff Quadrangular. It was assumed that the pipe is made of structural steel, with a Young's modulus of 205 GPa, a density of 7800 kg/m^3, and a Poisson's ratio of 0.3. The model consists of 5020 nodes and 5000 four-node quadrilateral elements (QUAD4).

The nodal displacement fields will be computed in both Salome-Meca and MFSim for validation purposes. These displacement fields are presented in Fig. 21. It can be observed that the nodal displacement fields obtained from Salome-Meca (Fig. 21 (a)) and MFSim (Fig. 21 (b)) are consistent. The MFSim user can choose an alternative approach: if they do not wish to solve the static structural analysis within MFSim, they can import the nodal displacement field from another software, such as Salome-Meca.



(a)

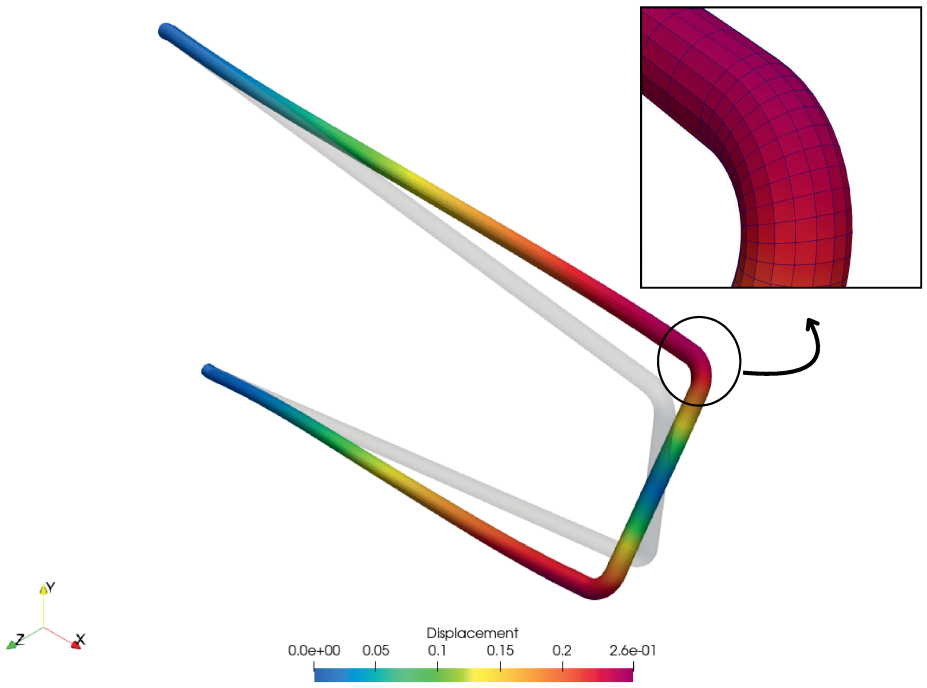


(b)

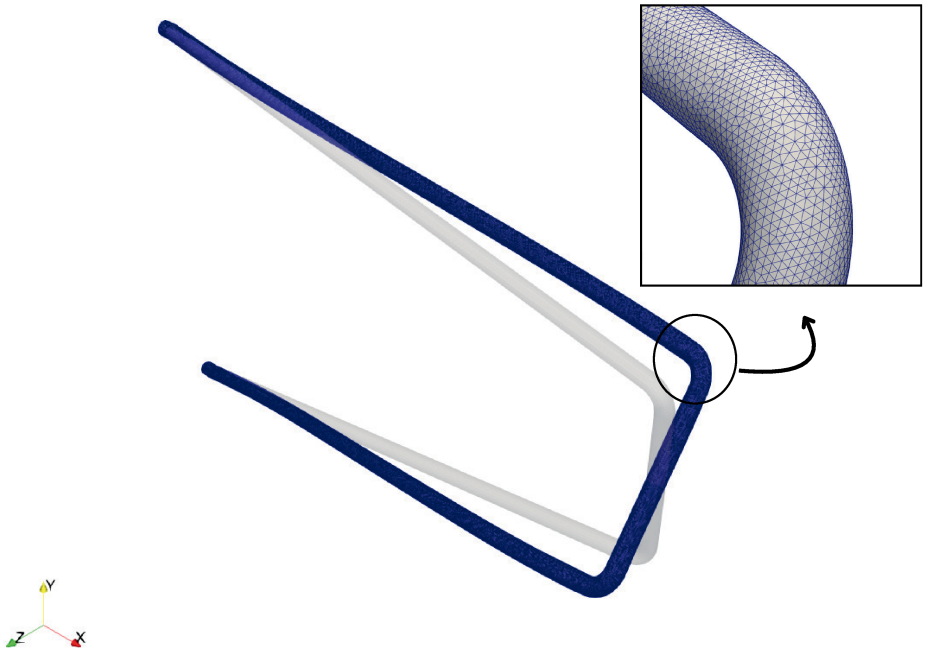
Figure 21 – Nodal displacement field obtained from Salome-Meca (a) and MFSim (b).

Based on the displacement field obtained from the static analysis solution in MFSim, the interpolation functions can be tested. The concept is that the Lagrangian mesh should follow the deformed surface of the structural mesh, maintaining the topological association between the two. To achieve this, a non-physical load is applied to the structure so that it undergoes a significant deformation, as shown in Fig. 22.

In Fig. 22 (a), the displacement field resulting from applying a static load to the heat exchanger structure is shown. The deformation can be compared to the initial state of the structure, represented by the gray shading. In Fig. 22 (b), the deformed



(a)



(b)

Figure 22 – Structural displacement obtained in MFSim (a) and the deformed immersed boundary mesh (b).

immersed boundary mesh due to the static load is displayed. A zoomed view of the Lagrangian mesh curve is provided, demonstrating that the mesh association algorithms and interpolation functions have performed correctly even in the critical region where the orthogonal projection method might fail.

An overlay of the zoomed views of the deformed structural mesh (Fig. 22 (a)) and the deformed Lagrangian mesh (Fig. 22 (b)) is presented in Fig. 23. The purpose of this image is to emphasize the discussion in Section 4.2.1 on Lagrangian mesh association. In that section, it was noted that there may be a discrepancy between the representation of curved regions by the structural mesh and the mesh used in the immersed boundary method. As observed in Fig. 23, the curve representation by the Lagrangian mesh is much smoother than that of the structural mesh. This is due to the difference in element sizes used in the two meshes.

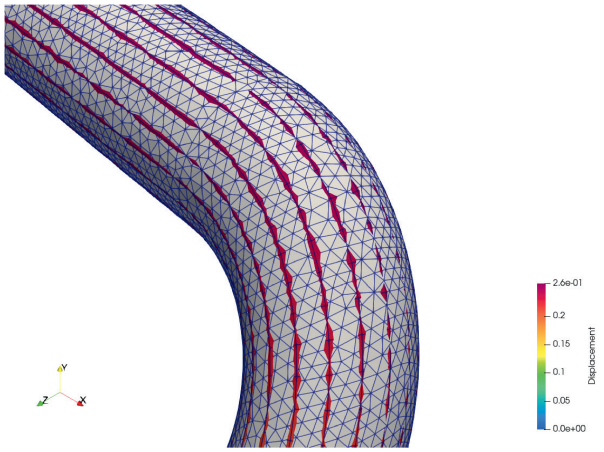
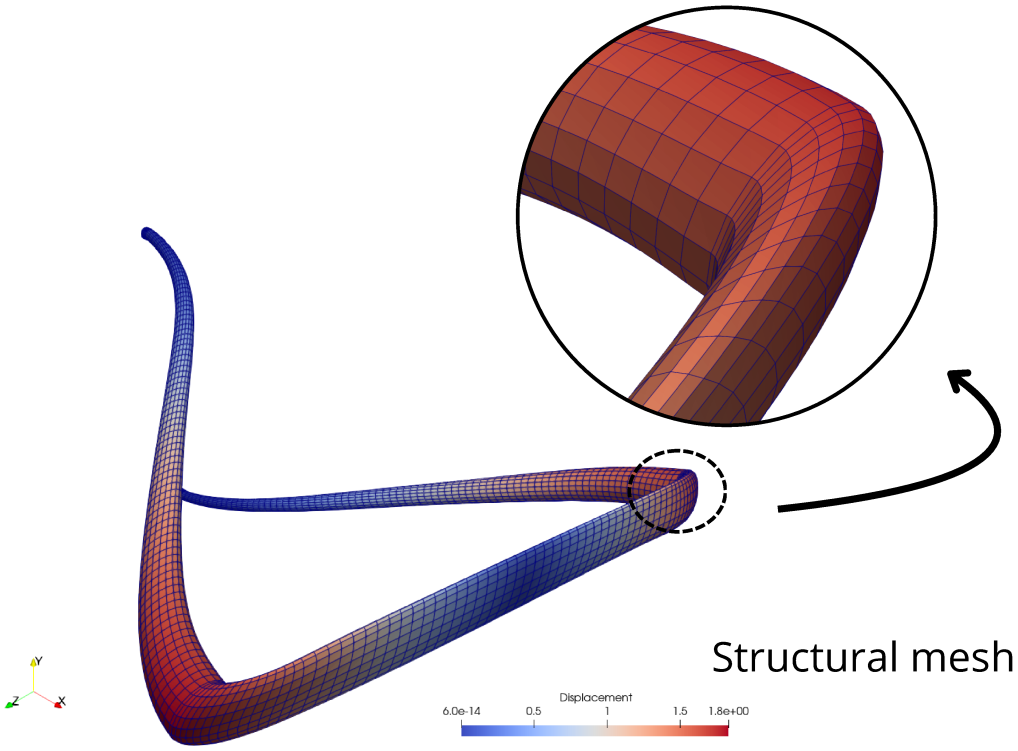


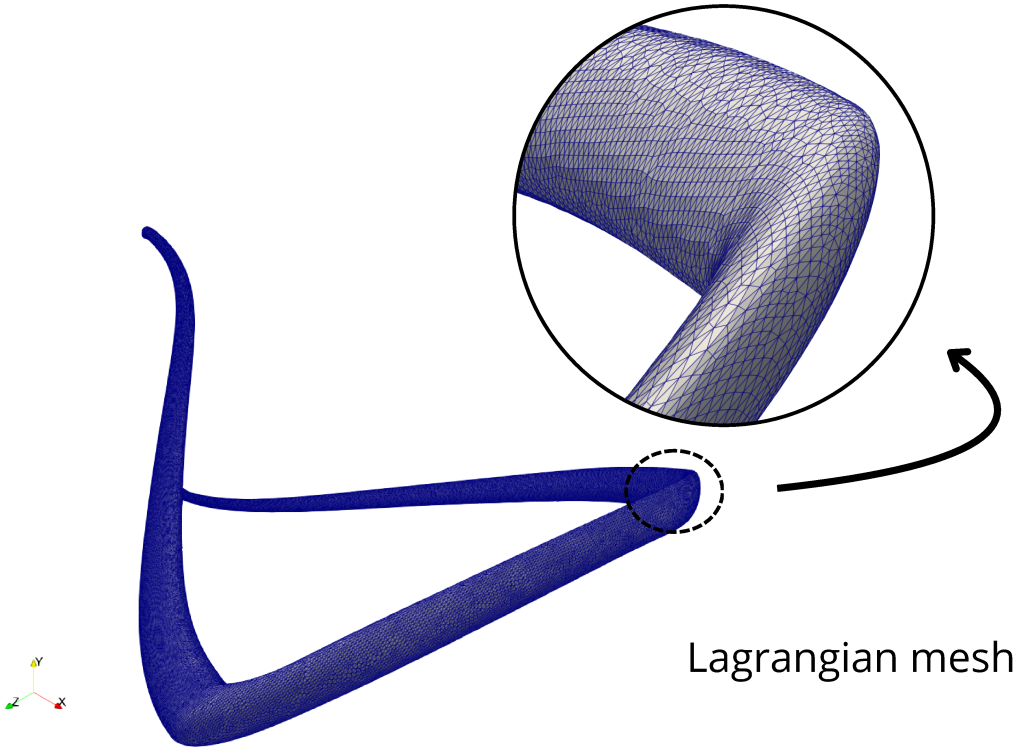
Figure 23 – Overlay of the deformed Lagrangian mesh on the deformed structural mesh.

A stress test of the interpolation functions was also conducted to evaluate their behavior under large displacements. In this stress test, a fictitious load (exceeding the material's capacity) is applied to generate a physically significant displacement, on the order of more than one meter. To achieve this, the force vector is configured so that the structure deforms extensively around the x-axis. The results can be seen in Fig. 24.

It can be observed that the interpolation functions enable the Lagrangian mesh to accurately follow the large deformation of the structural mesh, even in regions where the geometry exhibits curves.



(a) Structural mesh



(b) Lagrangian mesh

Figure 24 – Representation of the deformed structural mesh (a) and the deformed Lagrangian mesh (b) under a large fictitious load.

5.2 Modal Analysis with Mixed Structural Models

Based on the discussion in Section 4.3.5, Coupling Between Elements of Different Theories, a validation case was set up to couple finite element models discretized using different theories, referred to in this work as the mixed model.

To this end, a cylinder with a length of 800 mm, an outer diameter of 30 mm, and a thickness of 2.5 mm was discretized in three ways: using plate elements, using beam elements, and, in the third case, discretizing half of its length with beam elements and the other half with plate elements to form a mixed finite element model. The models were created in Salome-Meca and then imported into MFSim. It was assumed that the cylinder is made of structural steel, with a Young's modulus of 205 GPa, a density of 7800 kg/m^3, and a Poisson's ratio of 0.3.

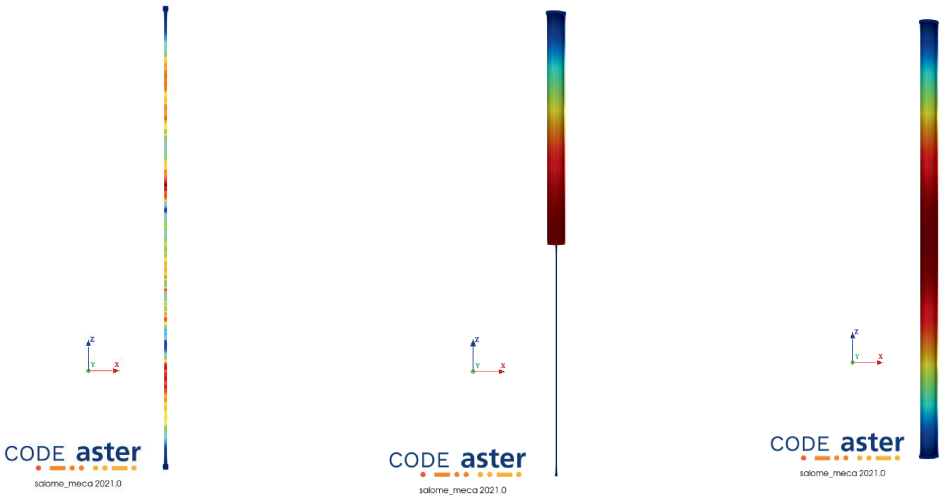
The finite element beam theory employed was POU_D_T — a Timoshenko beam element — while the plate theory used was the DKQ finite element model (Discrete Kirchhoff Quadrangular). The cylinder was constrained with fixed-fixed boundary conditions. For the translational degrees of freedom, a stiffness of 1×10^9 N/m was assigned, and for the rotational degrees of freedom, a stiffness of 1×10^9 Nm was used. The coupling between the beam and plate models was achieved using the LIAISON_ELEM modeling tool, as detailed in Section 4.3.5, Coupling Between Elements of Different Theories.

The purpose of this analysis is to determine whether the mixed model can represent the structural characteristics — such as natural frequencies — in a manner equivalent to that obtained from pure beam or pure plate models. It should be noted that this is a robustness analysis intended to evaluate the feasibility of using such mixed modeling for fluid-structure interaction analysis in MFSim. In practice, each structural model should be applied with its specific uses and limitations in mind, and the coupling between different finite element theories should be understood as a modeling tool with its own advantages and drawbacks.

Figure 25 shows the first bending mode (a) and the first torsion mode (b) for the three models (beam, shell-beam, and plate). For comparison purposes, the first 10 vibration modes of the cylinder were computed for all three models, and their natural frequencies are compared in Tab. 12.



(a) First bending mode for the beam model (left), shell-beam model (middle), and plate model (right).



(b) First torsion mode for the beam model (left), shell-beam model (middle), and plate model (right).

Figure 25 – Bending (a) and torsion (b) vibration modes obtained using Salome-Meca.

Table 12 presents the comparison of the first 10 natural frequencies among the three models. BM stands for Beam Model, SM for Shell Model, and SBM for Shell-Beam Model, while ε represents the relative error.

Table 2 – Comparison of natural frequencies for the first 10 vibration modes.

Mode	**SM** (Hz)	**BM** (Hz)	**SBM** (Hz)	ε_{SM-BM} (%)	ε_{SM-SBM} (%)	ε_{BM-SBM} (%)
1	295.49	273.10	282.41	7.58	4.43	3.41
2	295.49	273.10	282.41	7.58	4.43	3.41
3	793.78	735.20	760.95	7.38	4.14	3.50
4	793.78	735.20	760.95	7.38	4.14	3.50
5	1507.25	1399.09	1442.47	7.18	4.30	3.10
6	1507.25	1399.09	1442.47	7.18	4.30	3.10
7	1990.85	1987.13	1987.37	0.19	0.18	0.01
8	2400.69	2233.33	2300.73	6.98	4.16	3.02
9	2400.69	2233.33	2300.73	6.98	4.16	3.02
10	3205.41	3169.08	3038.22	1.13	5.22	4.13

SM: Shell Model, BM: Beam Model, SBM: Shell-Beam Model, ε: Relative Error

It is evident that the structural property — the natural frequency — can be represented equivalently by the three models, demonstrating that the coupling approach used for the mixed model is representative. It is important to emphasize, however, that the mixed modeling applied to the cylindrical geometry may not directly correspond to a practical engineering application. In this work, it is used solely for illustrative purposes.

Another important aspect that should be highlighted is that the coupling modeling used is based on Lagrange degrees of freedom. The concept of Lagrange degrees of freedom is directly related to the use of Lagrange multipliers to enforce constraints between different finite element formulations. This method allows the imposition of compatibility conditions without modifying the original differential equations, ensuring that solutions across different domains remain physically consistent and mathematically stable.

In coupling problems involving different FEM formulations, Lagrange multipliers act as fictitious forces or reactions that ensure the continuity of primary variables at the domain interface (GOCKENBACH, 2006).

Mathematically, introducing Lagrange multipliers adds a new set of equations to the FEM system, resulting in an augmented matrix that incorporates the constraints between the two methods. The major advantage of this approach is that it preserves the variational structure of the problem, unlike penalty methods, which can lead to ill-conditioned systems (GOCKENBACH, 2006).

5.3 Modal Analysis Considering Internal Pressure and Added Mass on a Pipe

In Massa et al. (2009), the influence of internal pressure on the natural frequency of a pipeline was investigated through an experimental study. A carbon steel pipe with an external diameter of 73 mm, a wall thickness of 5 mm, and a length of 11 m was welded at both ends and filled with oil (density of 900 kg/m^3). The internal pressure was incrementally increased from 0 to 30 MPa in 5 MPa steps; for each pressure variation, the natural frequency was determined while axial deformations, stresses, and the natural vibration frequencies in the vertical plane were measured.

Strain gauges placed along the neutral axis at the midpoint of the pipe recorded the axial deformations and stresses. The experiment specifically excited only the first bending mode, and the natural frequencies were extracted from the analyzed response signals.

In this work, the effects of internal pressure and the contained fluid are modeled using the finite element method. The objective is to incorporate the influence of these variables (internal pressure and fluid mass) into the model and to compare the modal analysis results with the experimental findings. For the computational modeling, the pipe was simulated using a Discrete Kirchhoff Quadrilateral (DKQ) finite element model, with stainless steel as the material. Tab. 3 summarizes the material properties and the oil density used in the analysis.

Table 3 – Material properties of the pipe and oil density.

Material	**Property**	**Value**
Stainless Steel	Young's Modulus (E) [Pa]	2.1×10^{11}
	Poisson's Ratio (ν)	0.3
	Density (ρ) [kg/m^3]	7850
Oil	Density (ρ) [kg/m^3]	900

The finite element mesh, illustrated in Fig. 26, comprises 25,025 nodes and 25,000 four-node quadrilateral (QUAD4) elements. A clamped-clamped boundary condition, modeled with elastic supports, was applied—using stiffness constants of 1.5×10^7 N/m for translations and 1×10^6 Nm for rotations.

In the finite element model, the internal pressure was applied over the plate element surfaces, while the mass of the oil was computed and evenly distributed among the structural nodes, integrating it into the global mass matrix.

Table 12 compares the natural frequencies obtained from the experimental study with those computed via the simulation.

The results indicate that increasing internal pressure enhances the stiffness of the steel pipe—a phenomenon reflected by an approximately 20% increase in the natural frequency of the first vibration mode when comparing the pressures of 5 MPa and 30 MPa. This

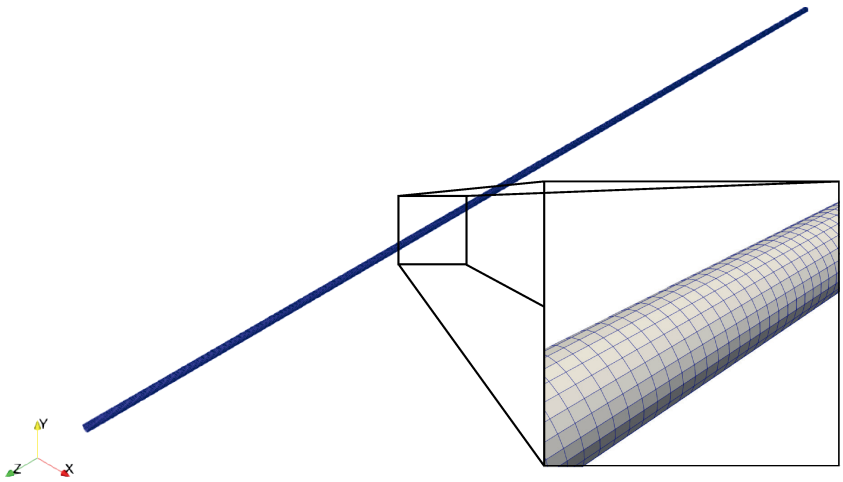


Figure 26 – Pipe structural mesh.

Table 4 – Comparison between experimental natural frequencies from Massa et al. (2009) and computational results.

P [MPa]	**Experimental [Hz]**	**Salome-Meca/MFSim [Hz]**	ε [%]
5	3.222	3.101	3.76
10	3.339	3.269	2.09
15	3.515	3.429	2.45
20	3.632	3.581	1.41
25	3.808	3.726	2.15
30	3.867	3.765	2.64

ε: Absolute relative error.

stiffening effect is captured by the geometric stiffness matrix, as discussed in Section 4.3.3.4 on Pre-stress Modeling and the Geometric Stiffness Matrix.

To incorporate this effect in the simulation, the static structural problem is first solved to obtain the stress field and the corresponding structural stiffness matrix. The geometric stiffness matrix is then derived from these stresses and added to the structural stiffness matrix, yielding the final equivalent stiffness of the system.

In summary, the modeling approach combines the effects of internal pressure and fluid mass distribution to accurately replicate the dynamic behavior of the pipe as observed experimentally.

5.4 Fluid-structure interaction analysis in a plate under a high-temperature environment

Wu et al. (2016) conducted a thermomodal experimental analysis to investigate the effects of extreme temperatures on the modal properties of a plate-like structure. The study aims to understand the dynamic behavior of structural components — such as the wings and rudders of hypersonic aircraft — under severe thermal conditions generated by aerodynamic heating at high speeds.

In the experiment, the modal parameters of a flat plate subjected to temperatures of up to 1200°C were measured. To achieve this, Wu et al. (2016) developed a thermo-vibratory testing system that combines a transient infrared radiation heating system with a vibratory excitation system.

The tested structure consisted of a rectangular plate made of nickel-based stainless steel (1Cr18Ni9Ti), cantilevered at one end. The temperature was progressively increased from 100°C to 1200°C using infrared radiation, with sensors distributed along the plate to ensure a homogeneous thermal distribution (WU et al., 2016).

Vibratory excitation was applied at the free end of the plate, while accelerometer sensors—positioned outside the high-temperature zone—recorded the vibratory responses. Ceramic rods resistant to heat were used for signal transmission. Data analysis was performed using the short-time Fourier transform (STFT), which allowed for the identification of variations in modal frequencies and vibration modes at different temperatures. Additionally, Wu et al. (2016) developed a finite element model in ANSYS to calculate the structure's modal parameters, taking into account the reduction in material stiffness, thermal stresses, and geometric nonlinearity effects.

This work aims to reproduce the experimental and numerical results obtained by Wu et al. (2016) through two main approaches. First, the effect of temperature variation on the plate will be analyzed by considering the material's modulus of elasticity as a function of the experimental temperature, as discussed in Section 4.3.4, Thermal Effects. Based on the modal analysis, the results will be compared with those of Wu et al. (2016). In the second approach, three configurations of the plate's free dynamic response will be evaluated. For this purpose, the structural model will be imported into MFSim, where a fluid-structure interaction (FSI) analysis will be performed. In this analysis, the structure will be excited by an impact test — a technique widely used in experimental modal analyses to excite a structure and measure its vibratory response (EWINS, 2000). The structure's natural frequencies will be obtained using the discrete Fourier transform (DFT), employing the fast Fourier transform (FFT) algorithm.

Table 5 presents the dimensions of the rectangular plate used in the study. The modulus of elasticity for different temperatures are documented in Tab. 1 of Wu et al. (2016). For this study, a function expressing the modulus of elasticity as a function of temperature

was determined. Curve fitting was performed using the Levenberg-Marquardt method, implemented in the SciPy library's `curve_fit` function.

Table 5 – Dimensions of the plate used in the experiment (WU et al., 2016).

Parameter	**Symbol**	**Value [mm]**
Length	L	340
Width	W	220
Thickness	t	7.5

The fitted function follows the exponential model:

$$E(T) = a \cdot e^{b \cdot T} + c \tag{5.1}$$

where a, b, and c are the parameters adjusted to minimize the difference between the experimental data and the modeled curve.

The fitted parameters are:

$$a = 1242.0367, \quad b = -0.0021, \quad c = -1057.9036 \tag{5.2}$$

Thus, the final fitted equation is:

$$E(T) = 1242.0367 \cdot e^{-0.0021 \cdot T} - 1057.9036 \tag{5.3}$$

Figure 27 shows the fitted curve along with the experimental data, where the black dots represent the measured data and the black dotted line indicates the exponential fit.

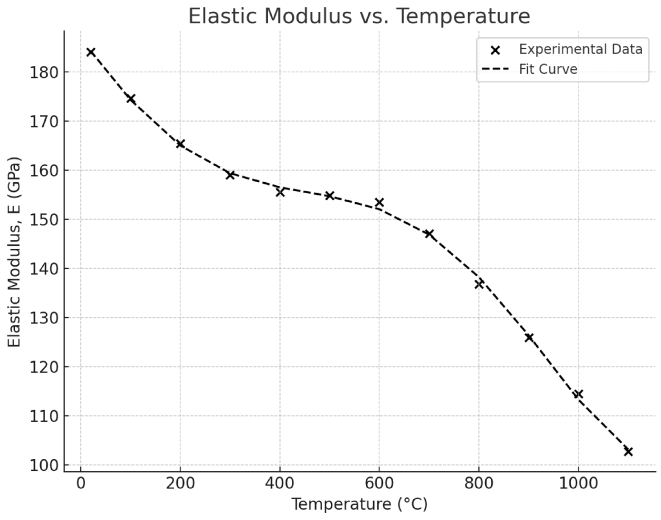


Figure 27 – Curve of the elastic modulus of stainless steel (1Cr18Ni9Ti) as a function of temperature. Black dots represent the measured data and the black dotted line indicates the exponential fit.

The geometry was discretized using the Discrete Kirchhoff Quadrilateral (DKQ) finite element model, which employs quadrilateral 4-node (QUAD4) elements. The mesh consists of 341 nodes and 300 elements, as shown in Fig. 28.

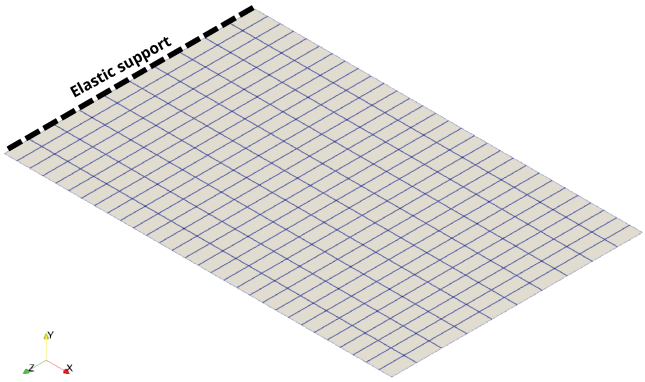


Figure 28 – Finite element mesh of the rectangular plate model.

It is assumed that the temperature field is uniform along the length of the plate, so the modulus of elasticity corresponding to the analysis temperature remains constant throughout the geometry. The finite element model was created using the Salome-Meca software, and the modal analysis was performed both in Salome-Meca and MFSim. The results of the modal analyses, compared to the experimental data from Wu et al. (2016), are presented in Tables 6 to 8.

Table 6 – Comparison of natural frequencies obtained experimentally by Wu et al. (2016) and in the present study for the first vibration mode of the plate.

T (°C)	**Wu et al. (2016) [Hz]**	**Present work [Hz]**	**A.R.E. (%)**
20	47.42	47.23	0.40
100	46.63	46.19	0.94
200	45.67	45.14	1.16
300	45.14	44.38	1.68
400	44.54	43.96	1.30
500	43.92	43.87	0.11
600	43.50	43.71	0.48
700	43.21	42.91	0.69
800	42.60	41.58	2.39
900	41.51	40.08	3.44
1000	38.92	38.43	1.26
1100	35.35	36.59	3.51

Table 7 – Comparison of natural frequencies obtained experimentally by Wu et al. (2016) and in the present study for the second vibration mode of the plate.

T (°C)	**Wu et al. (2016) [Hz]**	**Present work [Hz]**	**A.R.E. (%)**
20	164.01	174.50	6.40
100	163.97	170.29	3.85
200	163.90	166.04	1.31
300	163.44	163.01	0.26
400	162.61	161.32	0.79
500	161.35	160.98	0.23
600	159.58	160.34	0.48
700	154.84	157.17	1.50
800	147.81	151.89	2.76
900	140.84	146.05	3.70
1000	131.66	139.63	6.05
1100	123.73	132.59	7.16

Table 8 – Comparison of natural frequencies obtained experimentally by Wu et al. (2016) and in the present study for the third vibration mode of the plate.

T (°C)	**Wu et al. (2016) [Hz]**	**Present work [Hz]**	**A.R.E. (%)**
20	294.68	296.55	0.63
100	291.40	289.84	0.54
200	286.28	283.05	1.13
300	280.44	278.18	0.81
400	275.06	275.46	0.15
500	272.97	274.91	0.71
600	268.34	273.89	2.07
700	263.49	268.77	2.00
800	257.53	260.23	1.05
900	251.67	250.73	0.37
1000	240.65	240.22	0.18
1100	232.56	228.63	1.69

For the bending modes (first and third), the results show good agreement with the experimental data. However, the greatest discrepancy is observed in the second mode, which exhibits torsional characteristics. This difference may stem from the assumption of a uniform temperature field along the plate. The results from the modal analyses compared with the numerical data from Wu et al. (2016) are presented in Tables 9 to 11.

Table 9 – Comparison of natural frequencies obtained numerically by Wu et al. (2016) and in the present study for the first vibration mode of the plate.

T (°C)	**Wu et al. (2016) [Hz]**	**Present work [Hz]**	**A.R.E. (%)**
20	48.45	47.23	2.52
100	47.21	46.19	2.16
200	45.78	45.14	1.40
300	45.43	44.38	2.31
400	44.90	43.96	2.09
500	44.47	43.87	1.35
600	43.89	43.71	0.41
700	43.85	42.91	2.14
800	42.82	41.58	2.90
900	42.57	40.08	5.85
1000	39.57	38.43	2.88
1100	37.61	36.59	2.71

Table 10 – Comparison of natural frequencies obtained numerically by Wu et al. (2016) and in the present study for the second vibration mode of the plate.

T (°C)	**Wu et al. (2016) [Hz]**	**Present work [Hz]**	**A.R.E. (%)**
20	166.03	174.50	5.10
100	165.71	170.29	2.76
200	165.44	166.04	0.36
300	164.90	163.01	1.15
400	163.36	161.32	1.25
500	162.51	160.98	0.94
600	160.90	160.34	0.35
700	158.62	157.17	0.91
800	154.65	151.89	1.78
900	148.66	146.05	1.76
1000	139.93	139.63	0.21
1100	129.91	132.59	2.06

Table 11 – Comparison of natural frequencies obtained numerically by Wu et al. (2016) and in the present study for the third vibration mode of the plate.

T (°C)	**Wu et al. (2016) [Hz]**	**Present work [Hz]**	**A.R.E. (%)**
20	296.09	296.55	0.16
100	292.30	289.84	0.84
200	288.57	283.05	1.91
300	284.02	278.18	2.06
400	279.46	275.46	1.43
500	276.20	274.91	0.47
600	274.54	273.89	0.24
700	267.45	268.77	0.49
800	261.80	260.23	0.60
900	252.95	250.73	0.88
1000	243.05	240.22	1.16
1100	235.48	228.63	2.91

When comparing the present study's results with the numerical data from Wu et al. (2016), a significant reduction in the absolute error is observed compared to the differences noted when comparing with the experimental data. This suggests that, in addition to modeling the modulus of elasticity as a function of temperature, including the interaction between the structure and the heated atmospheric air in the free dynamic response may more accurately capture the physical phenomenon. To explore this hypothesis, the second phase of this validation study has been developed.

In this phase, fluid–structure interaction analyses will be conducted at temperatures of 200 °C, 600 °C, and 1100 °C. Thus, in addition to modeling the influence of the temperature field on the plate via a temperature-dependent modulus of elasticity, the interaction between the structure and the heated atmospheric air will be taken into account to enhance the representation of the structure's free dynamic response.

For this purpose, FSI cases are set up within MFSim, requiring the definition of the domain dimensions and boundary conditions. As a modeling assumption, there will be no thermal exchange between the structure and the fluid dynamic domain, as the thermal influence is already represented in the structural model. It is assumed that the air remains stationary throughout the domain, and the impact test on the structure will trigger the fluid-structure interaction.

The Eulerian domain was created based on the characteristic length of the plate. In this case, the chosen characteristic length was half of the plate's longest dimension (170 mm). It was determined that along this characteristic length, there should be 20 volumes, in order to define the Eulerian mesh's l_{top} (8.5 mm). Consequently, the computational domain has dimensions of 2.72 meters in the x-direction and 1.36 meters in both the y and z-directions. The computational domain can be seen in Fig. 29.

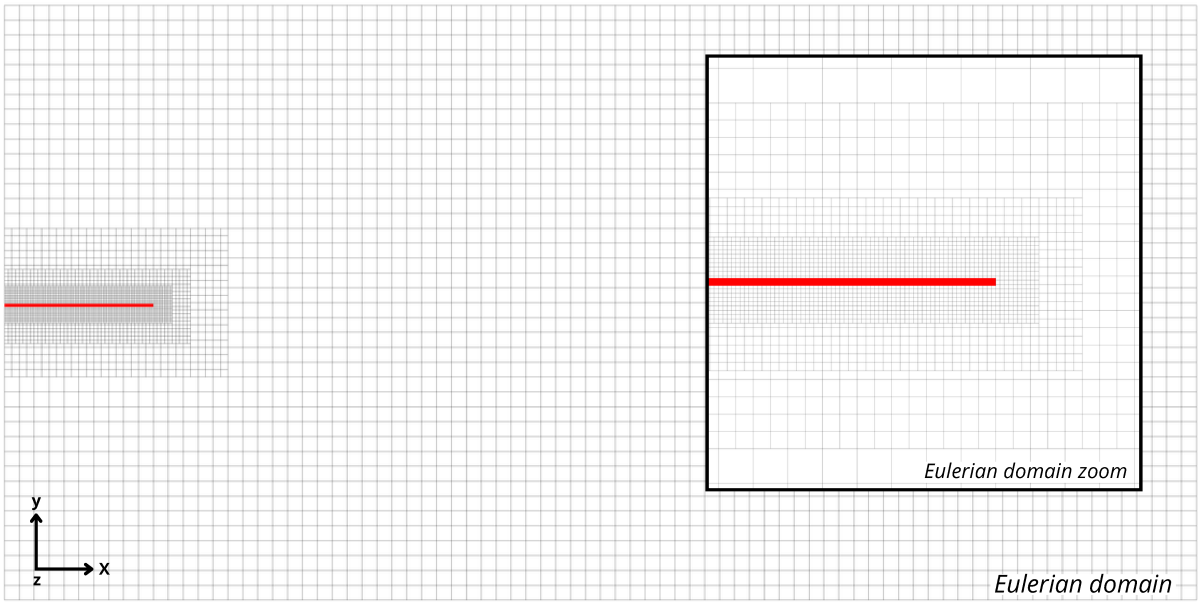


Figure 29 – Eulerian domain created for fluid-structure interaction simulation.

In Fig. 29, the geometry highlighted in red represents the position of the immersed boundary method mesh, which defines the location of the structure within the Eulerian domain. The boundary conditions on the yz-plane at the minimum x-coordinate impose a no-slip condition, setting the velocity to zero in all directions (wall condition). On the remaining domain faces, Neumann boundary conditions were applied to the velocity, while Dirichlet boundary conditions were imposed on the pressure.

The turbulence model used in the simulation is based on the Large Eddy Simulation (LES) approach, with the application of the Dynamic Smagorinsky Model and an explicit filter for LES models. The advective term was discretized using the Cubista method, and the temporal discretization of the turbulence variables follows a semi-implicit scheme. The flux construction strategy was set to traditional, and the solver used for the turbulence equations is Multigrid.

The location of the applied force, the boundary condition of the plate, and the measurement points for the dynamic response can be seen in Fig. 30. A force of 1 kN was applied in the negative y-direction and remained active in the system for 1×10^{-3} s. The simulation was performed using a fixed time step of 1×10^{-4} s, and the dynamic response was analyzed over a period of 10 seconds. Proportional damping was applied with $\alpha = 1.0$ and $\beta = 1.0 \times 10^{-5}$.

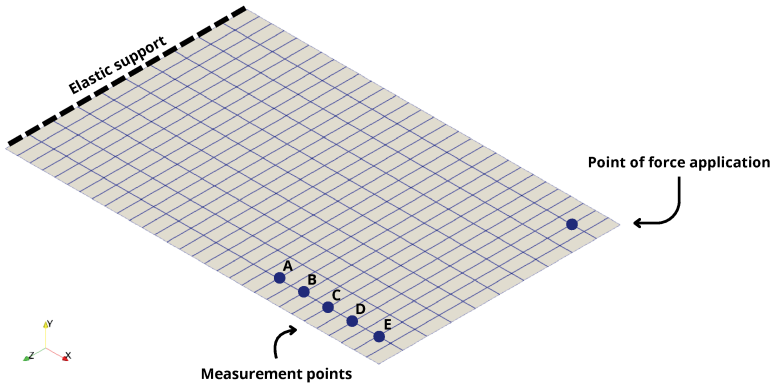
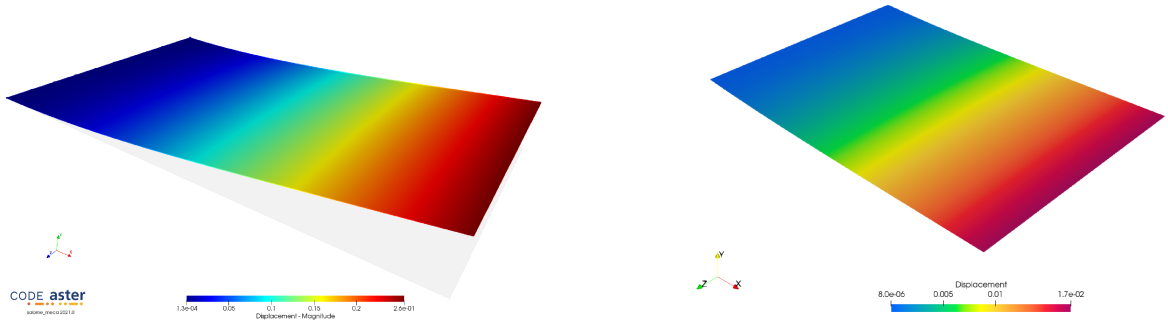


Figure 30 – Location of the applied excitation force and measurement points of the structural response.

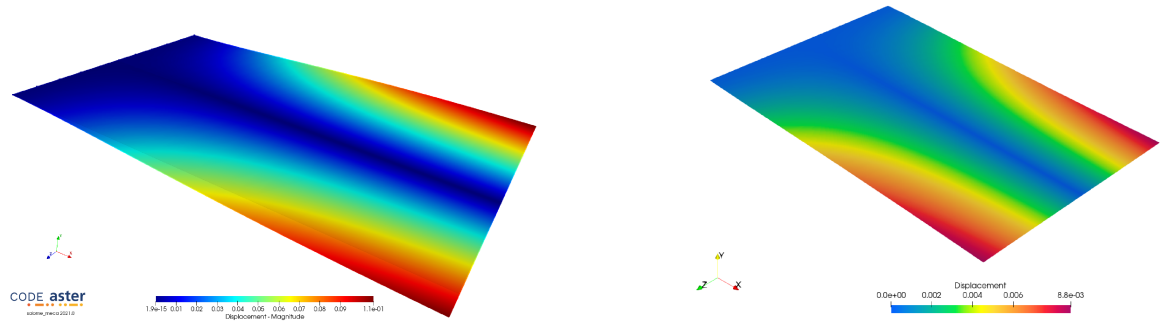
The modal analysis was performed using both Salome-Meca and MFSim. Figure 31 shows the first three vibration modes of the plate, obtained using Salome-Meca (a) and MFSim (b). The first and third modes are bending modes, while the second mode is torsional.

First vibration mode



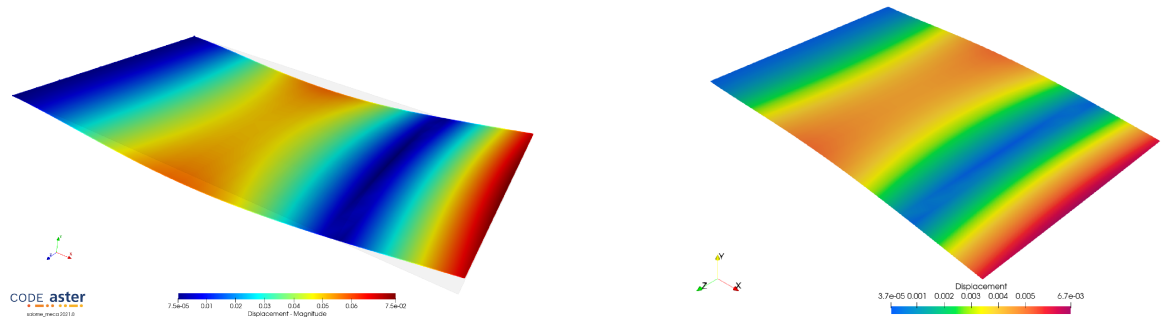
(a) (b)

Second vibration mode



(a) (b)

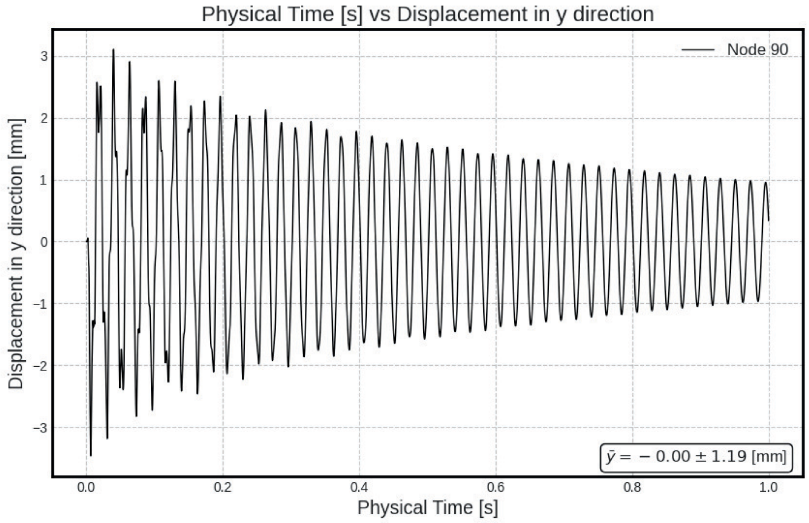
Third vibration mode



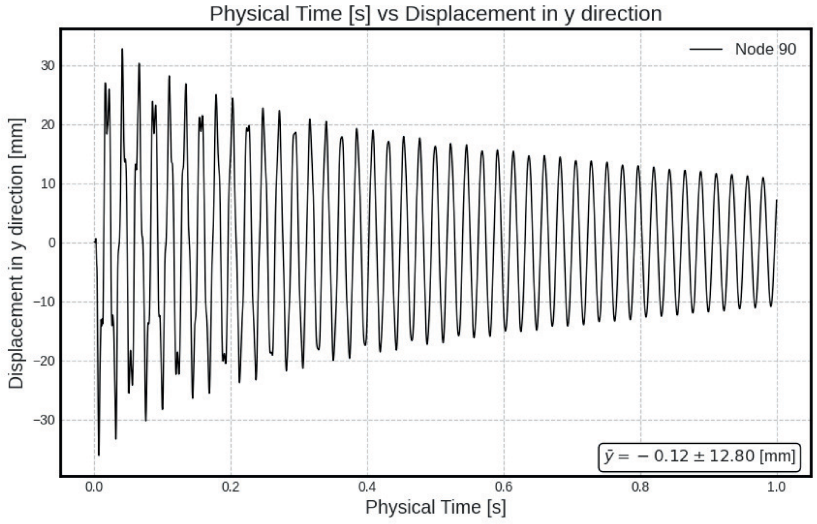
(a) (b)

Figure 31 – Comparison between the first three vibration modes of the plate obtained in Salome-Meca (a) and MFSim (b).

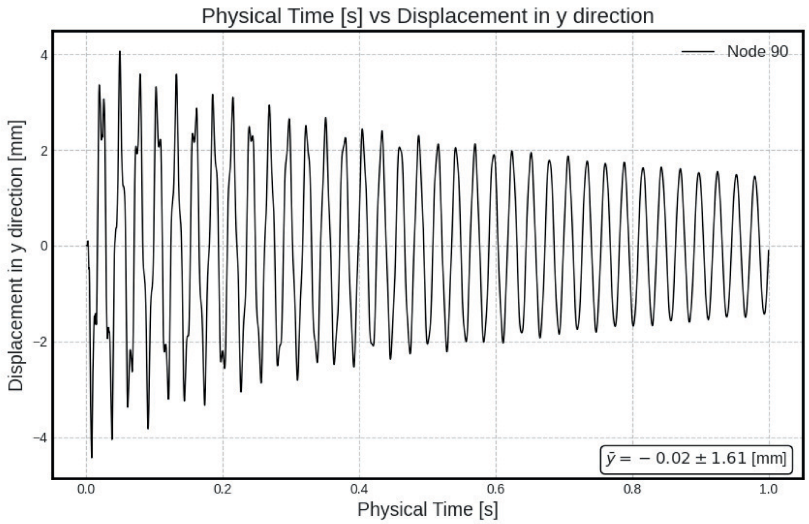
In the fluid-structure interaction analysis, nodal displacements and velocities are saved at each time step in MFSim. Based on this dataset, it is possible to plot the structural response (displacement or velocity) over the simulation time. As an example, the nodal displacements in the y-direction at measurement point E are presented, as shown in Fig. 30. The displacement results are illustrated in Fig. 32. From the time-domain signal, the Discrete Fourier Transform (DFT) can be computed using the Fast Fourier Transform (FFT) algorithm.



(a) Temperature = 200°C



(b) Temperature = 600°C



(c) Temperature = 1100°C

Figure 32 – Comparison of displacement over time at different temperatures for measurement point E.

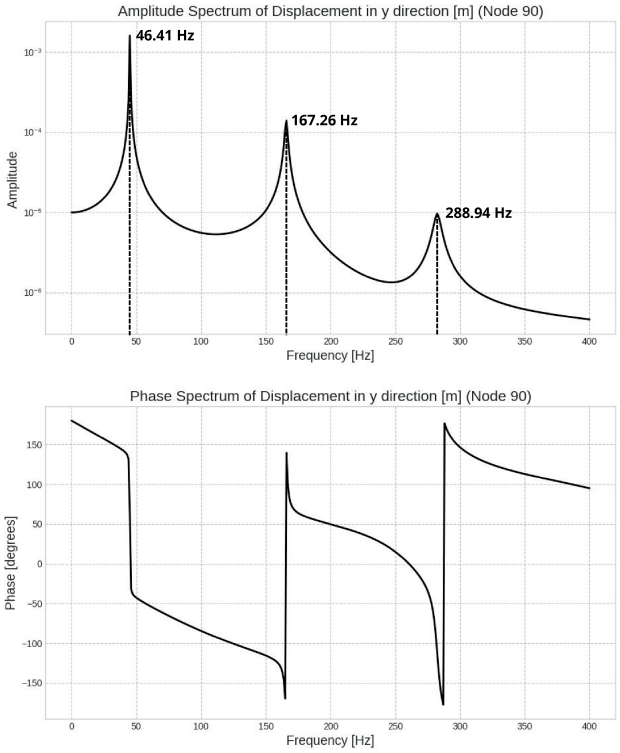
From the structural response graphs shown in Fig. 32, the frequency response diagrams of the structure, along with the phase diagram, are obtained. For each measurement point, the frequency response graph of the structure is constructed. Fig. 33 presents the structural response in the frequency domain for measurement point E at temperatures of 200°C and 600°C. The comparison between the experimental results obtained by Wu et al. (2016) and the present study using fluid-structure interaction analysis is presented in Tab. 12.

Table 12 – Comparison between the natural frequencies obtained experimentally by Wu et al. (2016) and those computed through fluid-structure interaction analysis in this study.

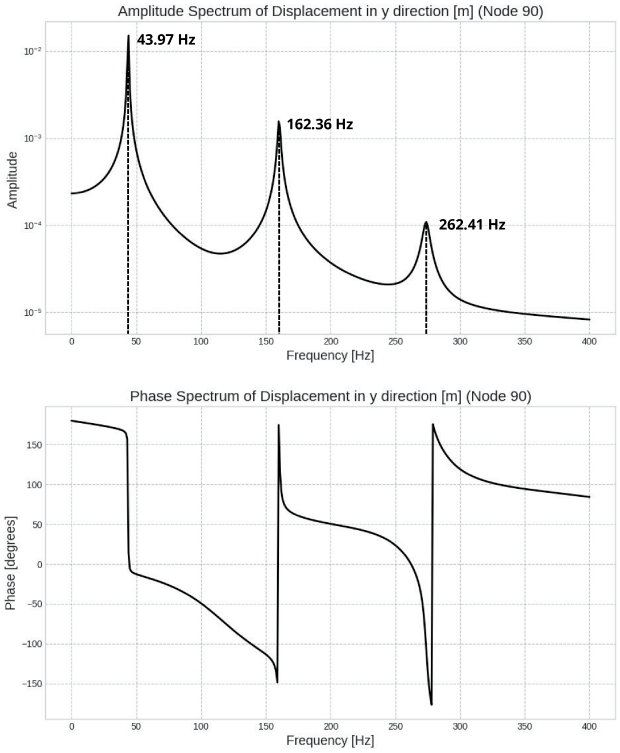
T [°C]	**Mode 1**			**Mode 2**			**Mode 3**		
	Exp. [Hz]	**FSI [Hz]**	ε [%]	**Exp. [Hz]**	**FSI [Hz]**	ε [%]	**Exp. [Hz]**	**FSI [Hz]**	ε [%]
200	45.67	46.41	1.62	163.9	167.26	2.05	286.28	288.94	0.93
600	43.50	43.97	1.08	159.58	162.36	1.74	268.34	262.41	2.21
1100	35.35	34.81	1.53	123.73	127.24	2.84	232.56	234.33	0.76

ε: Absolute relative error.

Table 12 shows that the FSI results for all three cases are in agreement with the experimental values obtained by Wu et al. (2016). This indicates that the methods developed in this study are validated for use in fluid-structure interaction simulations.



(a) Temperature = 200°C



(b) Temperature = 600°C

Figure 33 – Structural response in the frequency domain for the cases of 200°C and 600°C, measurement point E.

Analyzing Fig. 33, the peaks in the amplitude versus frequency graph represent the natural frequencies of the structure. These are the frequencies at which the system resonates when excited by an impulsive load. In the phase versus frequency graph, it can be observed that these natural frequencies are associated with a transition in the phase shift of the response, characterizing a progressive change in the phase relationship between the excitation force and the structural displacement.

This phase variation indicates that the structure is responding at its damped natural frequency, where the phase shift between excitation and response is not exactly 90° but slightly greater than 90°, depending on the system's damping factor. This phenomenon occurs due to the gradual conversion of vibrational energy into other forms, altering the balance between potential and kinetic energy and introducing an additional delay in the structural response.

If the system were undamped, the response at the natural frequency would exhibit an exact phase shift of 90° relative to the applied force, as described in the modal analysis of oscillatory systems. However, due to the transformation of mechanical energy into heat or internal deformation, a deviation in this phase shift is observed. This deviation can be analyzed experimentally to estimate the modal damping level. This behavior is consistent with modal theory, where the phase shift at resonance is given by:

$$\phi(\omega_n) = \tan^{-1}\left(\frac{2\zeta}{\sqrt{1-4\zeta^2}}\right), \tag{5.4}$$

indicating that the greater the damping (ζ), the higher the phase shift beyond 90° at the natural frequency (EWINS, 2000).

6 Conclusion

Conclusion

The integration of structural models from external finite element software into MFSim was a crucial step toward expanding its capabilities for fluid–structure interaction (FSI) simulations. This development allowed for detailed structural modeling and the inclusion of complex mechanical behaviors and loading conditions, which are essential in industrial applications such as heat recovery boilers. A core aspect of this work was the implementation of an algorithm that couples the structural mesh with the Lagrangian mesh used in the immersed boundary method. This coupling ensured precise force transmission between fluid and structure.

The primary objectives of this work involved defining a coupling approach between MFSim and external finite element models, implementing an algorithm to align the structural and Lagrangian meshes, developing the necessary tools for FSI using these external models, and validating the new methods. Each of these aims was achieved, as demonstrated by the consistent numerical and experimental comparisons. The coupling strategy successfully integrated structural models within MFSim, the mesh-alignment algorithm provided accurate force transfers, and the final toolset showed reliability through validation cases.

The main challenges encountered in this work were related to the development and implementation of computational tools. This type of task requires a high degree of interdisciplinarity, as the methods and approaches needed to achieve the objectives are not confined to a single field of knowledge but are spread across different layers of expertise, encompassing various areas of science and engineering.

This work introduced a set of computational tools aimed at improving FSI simulations within MFSim, with a particular focus on integrating external finite element models. One of the key developments was the software integration, which enabled MFSim to read and incorporate structural models from external finite element software. This enhancement broadens the range of structural analyses that can be performed but still requires refinements to fully accommodate more complex and nonlinear structural behaviors.

Another important contribution was the implementation of a mesh coupling algorithm, designed to associate the Lagrangian mesh used in the immersed boundary method with structural meshes from finite element models. This approach ensures that the coupling between the Eulerian and Lagrangian domains is correctly established, allowing for a consistent fluid–structure interaction simulation.

To assess the reliability of the proposed methods, validation against experiments was

conducted, comparing numerical results with available experimental data. The comparisons demonstrated that the implemented techniques provide results within an acceptable range of accuracy, supporting their applicability to specific FSI problems.

Finally, this work provides a foundation for future research, offering a structured methodology that can be extended to other engineering problems involving fluid-induced structural loads. While the tools developed here contribute to the computational modeling of FSI, further developments are necessary to improve numerical stability, extend applicability to more complex structural behaviors, and refine the coupling strategies for different engineering scenarios. Overall, these developments form a platform for further exploration of fluid–structure interaction in both academic and industrial contexts. Future work may focus on refining the numerical methods, extending the approach to additional structural element types, and implementing more sophisticated material models that address nonlinear or transient thermomechanical effects.

Bibliography

Ávila, I. A. M. *Modelagem Matemática e Computacional de Interação Fluido-Estrutura com Elementos de Casca Curvilíneos Tridimensionais*. Dissertação (Master thesis) — Federal University of Uberlândia, Uberlândia, MG, Brazil, 2021.

ASSI, G. R. S. et al. Suppression of wake-induced vibration of tandem cylinders with free-to-rotate control plates. *Journal of Fluids and Structures*, v. 26, n. 7-8, p. 1045–1057, 2010b. Disponível em: <https://doi.org/10.1016/j.jfluidstructs.2010.07.004>.

ASSI, G. R. S.; BEARMAN, P. W.; MENEGHINI, J. R. On the wake-induced vibration of tandem circular cylinders: the vortex interaction excitation mechanism. *Journal of Fluid Mechanics*, v. 661, p. 365–401, 2010. Disponível em: <https://doi.org/10.1017/S0022112010002834>.

AUBRY, J.-P. *Beginning with Code_Aster: A Practical Introduction to Finite Element Method using Code_Aster and Gmsh*. [S.l.]: Framasoft (coll. Framabook), 2019. ISBN: 979-10-92674-03-3. Available at: <http://framabook.org>.

AWADALLAH, M. O. et al. Boosting energy harvesting efficiency from wake-induced vibration using a multi-cylinder configuration. *Applied Energy*, v. 381, p. 125181, 2025. Disponível em: <https://doi.org/10.1016/j.apenergy.2024.125181>.

BARBI, F. et al. Numerical experiments of ascending bubbles for fluid dynamic force calculations. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, v. 40, n. 11, p. 519, 2018. ISSN 1806-3691. Disponível em: <https://doi.org/10.1007/s40430-018-1435-7>.

BATHE, K.-J. *Finite Element Procedures*. 2nd. ed. [S.l.]: K.J. Bathe, Watertown, MA, 2016. ISBN 978-0-9790049-5-7.

BAZILEVS, Y.; TAKIZAWA, K.; TEZDUYAR, T. E. *Computational Fluid-Structure Interaction: Methods and Applications*. Chichester, West Sussex, UK: Wiley, 2013.

BEER, F. P. et al. *Mecânica dos Materiais*. 6th. ed. [S.l.]: McGraw-Hill, 2015. ISBN 978-8-579-80606-0.

BELYTSCHKO, T.; LIN, J. I.; TSAY, C.-S. Modal recovery methods for solution of fluid–structure problems with rigid wall loads. *Nuclear Engineering and Design*, v. 71, n. 1, p. 67–78, 1982. ISSN 0029-5493. Disponível em: <https://doi.org/10.1016/0029-5493(82)90170-4>.

BELYTSCHKO, T.; LIU, W. K.; MORAN, B. *Nonlinear Finite Elements for Continua and Structures*. Chichester, England: John Wiley & Sons, Ltd., 2000. ISBN 0-471-98773-5.

BERG, M. de et al. *Computational Geometry: Algorithms and Applications*. 3rd. ed. Berlin, Germany: Springer, 2008. ISBN 978-3-540-77973-5.

BLAZEK, J. *Computational Fluid Dynamics: Principles and Applications*. Amsterdam, London, New York, Oxford, Paris, Shannon, Tokyo: Elsevier, 2001. ISBN 0-08-043009-0.

BLEVINS, R. D. *Flow-Induced Vibration*. 2nd. ed. New York, NY: Van Nostrand Reinhold Company, 1990. 1–560 p. ISBN 1-57524-183-8.

BOKAIAN, A.; GEOOLA, F. Wake-induced galloping of two interfering circular cylinders. *Journal of Fluid Mechanics*, v. 146, p. 383–415, 1984. Disponível em: <https://doi.org/10.1017/S0022112084001602>.

BONDY, J. A.; MURTY, U. *Graph Theory*. Springer, 2008. v. 244. (Graduate Texts in Mathematics, v. 244). ISBN 978-1846289699. Disponível em: <https://doi.org/10.1007/978-1-84628-970-5>.

BUHMANN, M. D. *Radial Basis Functions: Theory and Implementations*. Cambridge, UK: Cambridge University Press, 2003. ISBN 978-0521633383. Disponível em: <https://doi.org/10.1017/CBO9780511543241>.

BUNGARTZ MIRIAM MEHL, M. S. H.-J. *Fluid-Structure Interaction II: Modelling, Simulation, Optimization*. Berlin, Heidelberg: Springer, 2010. v. 73. (Lecture Notes in Computational Science and Engineering, v. 73). ISBN 978-3-642-14206-2. Disponível em: <https://doi.org/10.1007/978-3-642-14206-2>.

CASTRO, L. P. d. et al. Implementation of a hybrid lagrangian filtered density function-large eddy simulation methodology in a dynamic adaptive mesh refinement environment. *Physics of Fluids*, v. 33, n. 4, p. 045126, 2021. Disponível em: <https://doi.org/10.1063/5.0045873>.

CEN, E. C. for S. *Eurocode 3: Design of Steel Structures - Part 1-2: General Rules – Structural Fire Design*. [S.l.]: CEN, 2005.

CHAPMAN, B.; JOST, G.; PAS, R. van der. *Using OpenMP: Portable Shared Memory Parallel Programming*. Cambridge, MA: The MIT Press, 2007. ISBN 978-0-262-53302-7.

CHEN, W. et al. Multi-direct forcing immersed boundary method for modelling heat and mass transfer of dynamic particles in three-dimensional fluid-solid systems. *Chemical Engineering Science*, 2025. Disponível em: <https://doi.org/10.1016/j.ces.2025.121284>.

CHOPRA, A. K. *Dynamics of Structures: Theory and Applications to Earthquake Engineering*. Fourth edition. Harlow, Essex, England: Pearson Education Limited, 2014. ISBN 978-0-273-77424-2.

Code Aster. *Code_Aster Command Documentation*. 2024. Accessed: December 19, 2024. Disponível em: <https://code-aster.org/doc/v12/en/index.php?man=commande&lang=en>.

CODE_ASTER. *Relations cinématiques linéaires de type RBE3*. [S.l.], 2015. Disponível em: <https://www.code-aster.org>.

CODE_ASTER. *Opérateurs AFFE_CHAR_MECA, AFFE_CHAR_MECA_C et AFFE_CHAR_MECA_F*. [S.l.], 2016. Disponível em: <https://www.code-aster.org>.

COLLAR, A. R. The interaction of aerodynamic and inertial forces on aircraft. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, The Royal Society, v. 204, n. 1079, p. 9–15, 1946.

COOK, R. D. et al. *Finite Element Analysis: Theory and Application with ANSYS*. 3rd. ed. [S.l.]: Prentice Hall, 2001. ISBN 978-0131890800.

COSTA, M. C. B. et al. Comparison between the fluid–structure interaction approach and the finite element method approach to analyze the leaflet flutter in bioprosthetic aortic valve. *Journal of Biomechanics*, v. 181, p. 112532, 2025. Disponível em: <https://doi.org/10.1016/j.jbiomech.2025.112532>.

CROLET, J.-M. *Fluid-Structure Interactions: Models, Analysis and Finite Elements*. Cham, Switzerland: Springer, 2017. ISBN 978-3-319-63970-3.

DAMASCENO, M.; VEDOVOTO, J. M.; SILVEIRA-NETO, A. Turbulent inlet conditions modeling using large-eddy simulations. *CMES - Computer Modeling in Engineering and Sciences*, v. 104, p. 105–132, 2015. Disponível em: <https://doi.org/10.3970/cmes.2015.104.105>.

DAMASCENO, M. M. R.; SANTOS, J. G. d. F.; VEDOVOTO, J. M. Simulation of turbulent reactive flows using a fdf methodology – advances in particle density control for normalized variables. *Computers & Fluids*, v. 170, p. 128–140, 2018. ISSN 0045-7930. Disponível em: <https://doi.org/10.1016/j.compfluid.2018.05.004>.

DAVIS, P. J.; RABINOWITZ, P. *Methods of Numerical Integration*. San Diego, CA, USA: Academic Press, 1984. ISBN 978-0-12-206360-2.

DOWELL, E. H. *A Modern Course in Aeroelasticity*. 6th. ed. [S.l.]: Springer, 2021.

EDF. *Analysis of Structures and Thermomechanics for Studies & Research*. [S.l.], 2024. Available at: <www.code-aster.org>.

EDF; CEA. *SALOME 7: The Open Source Integration Platform for Numerical Simulation*. [S.l.], 2013. Available at: <http://www.salome-platform.org>.

ELIAS, A. J. *Modelagem hibrida urans-les para escoamentos turbulentos*. Dissertação (Mestrado) — Universidade Federal de Uberlandia, Uberlandia, Brasil, 2018. Disponível em: <https://doi.org/10.14393/ufu.di.2018.1219>.

ERICSON, C. *Real-Time Collision Detection*. 1st. ed. San Francisco, CA: Morgan Kaufmann, 2004. ISBN 978-1-55860-732-3.

EWINS, D. J. *Modal Testing: Theory, Practice and Application*. Second. Baldock, Hertfordshire, England: Research Studies Press Ltd., 2000. ISBN 0-86380-218-4.

FASSHAUER, G. E.; MCCOURT, M. J. *Kernel-based Approximation Methods using MATLAB*. Singapore: World Scientific, 2015. v. 19. (Interdisciplinary Mathematical Sciences, v. 19). ISBN 978-9814630156. Disponível em: <https://doi.org/10.1142/9789814630163>.

FISH, J. *A First Course in Finite Elements Analysis*. [S.l.]: John Wiley & Sons, 2007. ISBN 978-0-471-76115-7.

FLÉJOU, J.-L. *Définition des éléments discrets pour conditions limites et masses concentrées*. [S.l.], 2015. Disponível em: <https://www.code-aster.org>.

FLÉJOU, J.-L. *Éléments "exacts" de poutres (droites et courbes)*. [S.l.], 2017. Disponível em: <https://www.code-aster.org>.

GANAPATHY, V. *Industrial Boilers and Heat Recovery Steam Generators: Design, Applications, and Calculations*. New York, USA: Marcel Dekker, 2003. ISBN 0-8247-0814-8.

GOCKENBACH, M. S. *Understanding and Implementing the Finite Element Method*. [S.l.]: Society for Industrial and Applied Mathematics (SIAM), 2006. ISBN 0-89871-614-4.

GOYDER, H. G. D. Flow-induced vibration in heat exchangers. *Chemical Engineering Research and Design*, v. 80, n. 3, p. 226–232, 2002. Disponível em: <https://doi.org/10.1205/026387602317446315>.

GRIFFITH, B. E.; PATANKAR, N. A. Immersed methods for fluid-structure interaction. *Annual Review of Fluid Mechanics*, v. 52, p. 421–448, 2020.

HAMIDA, M. B. B. Investigation of fluid-structure interaction and heat transfer dynamics in a wavy L-shaped enclosure with an elastic baffle. *International Communications in Heat and Mass Transfer*, v. 162, p. 108617, 2025. Disponível em: <https://doi.org/10.1016/j.icheatmasstransfer.2025.108617>.

HASHIGUCHI, K. *Nonlinear Continuum Mechanics for Finite Elasticity-Plasticity: Multiplicative Decomposition with Subloading Surface Model*. [S.l.]: Elsevier, 2020. ISBN 978-0-12-819428-7.

HOU JIN WANG, A. L. G. Numerical methods for fluid-structure interaction - a review. *Communications in Computational Physics*, v. 12, n. 2, p. 337–377, 2012. Disponível em: <https://doi.org/10.4208/cicp.291210.290411s>.

HUGHES, J. F. et al. *Computer Graphics: Principles and Practice*. Third. Upper Saddle River, NJ, USA: Addison-Wesley, 2014. ISBN 978-0-321-39952-6.

INMAN, D. J. *Engineering Vibrations*. Fourth edition. Upper Saddle River, New Jersey: Pearson Education, 2014. ISBN 978-0-13-287169-3.

KALITA, A.; RAO, T. An overview of fluid-structure interaction: Modelling, finite element method and applications. *Annals of Multidisciplinary Research, Innovation and Technology (AMRIT)*, Assam down town University, v. 2, n. 2, p. 93–105, 2023. ISSN 2583-4657.

KIM, S.; SHIN, S. Improved unsteady fluid–structure interaction analysis using the dynamic mode decomposition on a composite marine propeller. *Ocean Engineering*, v. 319, p. 120255, 2025. Disponível em: <https://doi.org/10.1016/j.oceaneng.2024.120255>.

KRYLOV, N.; LOBANDIYEVSKY, P.; MEN, S. *Descriptive Geometry*. Moscow: Mir Publishers, 1968.

LEE BEHNAM BAGHERI, C. J. J. Introduction to digital twin: A simulation-based architecture for smart manufacturing. *Manufacturing Letters*, v. 15, p. 18–23, 2018.

LENGYEL, E. *Mathematics for 3D Game Programming and Computer Graphics*. Third. Boston, MA, USA: Course Technology PTR, Cengage Learning, 2012. ISBN 978-1-4354-5886-4.

LI, K. et al. Flutter analysis of compressor blades under travelling wave modes using an efficient fluid–structure interaction method. *Chinese Journal of Aeronautics*, v. 36, n. 11, p. 221–232, 2023. Disponível em: <https://doi.org/10.1016/j.cja.2023.09.021>.

MARTINI, J. E. F. *Modelagem Computacional de Escoamentos Turbulentos em Turbinas Eólicas Utilizando Fronteira Imersa e Refinamento Adaptativo da Malha*. Dissertação (Master thesis) — Federal University of Uberlândia, Uberlândia, Brazil, 2021. Disponível em: <https://doi.org/10.14393/ufu.di.2021.455>.

MASSA, A. L. L. et al. The influence of internal pressure on pipeline natural frequency. In: *Proceedings of the 28th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2009)*. [s.n.], 2009. Disponível em: <https://doi.org/10.1115/OMAE2009-79666>.

MCGUIRE, W.; ZIEMIAN, R. D. *Matrix Structural Analysis*. 2nd. ed. [S.l.]: John Wiley & Sons, 2014. ISBN 9781118559256.

MEIROVITCH, L. *Fundamentals of Vibrations*. New York: McGraw-Hill, 2001. ISBN 0-07-041345-2.

MELO, R. R. S. *Modelagem e simulacao de escoamentos turbulentos com efeitos termicos, utilizando a metodologia de fronteira imersa e malha adaptativa*. Tese (Doutorado) — Universidade Federal de Uberlandia, Uberlandia, Brasil, 2017.

MENEGHINI, J. R. Mecânica da geração e desprendimento de vórtices no escoamento ao redor de cilindros. In: *Anais III Escola de Primavera Transição e Turbulência*. [S.l.: s.n.], 2002.

MORALES, F. A. P. *Modelagem Matemática e Computacional de Interação Fluido-Estrutura Utilizando Elementos Sólidos com Aplicação na Indústria*. Tese (PhD thesis) — Federal University of Uberlândia, Uberlândia, MG, Brazil, 2023.

MORALES, F. A. P. et al. Fluid–structure interaction with a finite element–immersed boundary approach for compressible flows. *Ocean Engineering*, v. 290, p. 115755, 2023. Disponível em: <https://doi.org/10.1016/j.oceaneng.2023.115755>.

MORETTI, P. M. Flow-induced vibrations in arrays of cylinders. *Annual Review of Fluid Mechanics*, v. 25, p. 99–114, 1993. Disponível em: <https://doi.org/10.1146/annurev.fl.25.010193.000531>.

MOSCHOPOULOS, P.; DIMAKOPOULOS, Y.; TSAMOPOULOS, J. A new finite element formulation unifying fluid-structure and fluid-fluid interaction problems. *Journal of Non-Newtonian Fluid Mechanics*, v. 336, p. 105366, 2025. Disponível em: <https://doi.org/10.1016/j.jnnfm.2024.105366>.

MOTA, P. H. A. *Modelagem numérico-computacional multiníveis de processos de corrosão em sistemas fluidodinâmicos*. Tese (Ph.D. Thesis) — Federal University of Uberlândia, Uberlândia, Brazil, 2023.

NAKAMURA, T. et al. *Flow-Induced Vibrations: Classifications and Lessons from Practical Experiences*. [S.l.]: Butterworth-Heinemann, 2013.

NETO, H. R. *Modelagem Matemática para a Interação Fluido-Estrutura Acoplando Escoamentos Incompressíveis e Viga de Timoshenko*. Dissertação (Master thesis) — Federal University of Uberlândia, Uberlândia, MG, Brasil, 2016.

NETO, H. R. et al. Influence of seabed proximity on the vibration responses of a pipeline accounting for fluid-structure interaction. *Mechanical Systems and Signal Processing*, Elsevier, v. 114, p. 224–238, 2019.

NEUMEISTER, R. F.; BARBIERI, M. R. *Vibração Induzida pelo Escoamento sobre Cilindros*. Blumenau, SC: Edifurb, 2022. Disponível em: <https://www.eesc.usp.br>.

NORBERG, C. Fluctuating lift on a circular cylinder: review and new measurements. *Journal of Fluids and Structures*, v. 17, n. 1, p. 57–96, 2003. Disponível em: <https://doi.org/10.1016/S0889-9746(02)00099-3>.

OBERKAMPF, C. J. R. W. L. *Verification and Validation in Scientific Computing*. [S.l.]: Cambridge University Press, 2010.

OKABE, Y. *Nonlinear Continuum Mechanics for Finite Elasticity-Plasticity: Multiplicative Decomposition with Subloading Surface Model*. [S.l.]: Elsevier, 2020. ISBN 9780128194287.

PAZ, M. *Structural Dynamics: Theory and Computation*. 5th. ed. Boston, MA: Springer, 2004. ISBN 978-1402077178.

PAïDOUSSIS, M. P. *Fluid-Structure Interactions, Volume 1: Slender Structures and Axial Flow*. 2nd. ed. Amsterdam, Boston, Heidelberg, London, New York, Oxford, Paris, San Diego, San Francisco, Singapore, Sydney, Tokyo: Elsevier, 2014. ISBN 978-0-12-397312-2.

PAïDOUSSIS, M. P.; PRICE, S. J.; LANGRE, E. de. *Fluid-Structure Interactions: Cross-Flow-Induced Instabilities*. Cambridge, UK: Cambridge University Press, 2011. ISBN 978-0-521-11942-9.

PESKIN, C. S. Flow patterns around heart valves: A numerical method. *Journal of Computational Physics*, v. 10, n. 2, p. 252–271, 1972.

PESKIN, C. S. The immersed boundary method. *Acta Numerica*, v. 11, p. 479–517, 2002.

PETTIGREW, M. J.; TAYLOR, C. E. Vibration analysis of shell-and-tube heat exchangers: an overview2̆014part 1: flow, damping, fluidelastic instability. *Journal of Fluids and Structures*, v. 18, n. 5, p. 469–483, 2003.

PETTIGREW, M. J. et al. Flow-induced vibration: recent findings and open questions. *Nuclear Engineering and Design*, v. 185, n. 2-3, p. 249–276, 1998. Disponível em: <https://doi.org/10.1016/S0029-5493(98)00230-5>.

PINHEIRO, A. P. et al. Modelling of aviation kerosene droplet heating and evaporation using complete fuel composition and surrogates. *Fuel*, v. 305, p. 121564, 2021. ISSN 0016-2361. Disponível em: <https://doi.org/10.1016/j.fuel.2021.121564>.

PIVELLO, M. et al. A fully adaptive front tracking method for the simulation of two phase flows. *International Journal of Multiphase Flow*, v. 58, p. 72–82, 2014. ISSN 0301-9322. Disponível em: <https://doi.org/10.1016/j.ijmultiphaseflow.2013.08.009>.

RAYAPROLU, K. *Boilers: A Practical Reference*. Boca Raton, FL, USA: CRC Press, 2012. ISBN 978-1-4665-0053-2.

RICHARDS, J. R. *Principles of Solid Mechanics*. Boca Raton, FL: CRC Press, 2001. ISBN 0-8493-0114-9.

ROBEY, R.; ZAMORA, Y. *Parallel and High Performance Computing*. Shelter Island, NY: Manning Publications, 2021. ISBN 9781617296468.

SANTOS, J. G. d. F. *Mathematical and computational modeling of gas-solid flows in dynamic adaptive mesh*. Dissertação (Mestrado) — Universidade Federal de Uberlandia, Uberlandia, Brasil, 2019.

SCHNEIDER, P. J.; EBERLY, D. H. *Geometric Tools for Computer Graphics*. San Francisco, CA, USA: Morgan Kaufmann, 2003. ISBN 1-55860-594-0.

SEDGEWICK, R. *Algorithms in C++, Part 5: Graph Algorithms*. Third. Boston, MA: Addison-Wesley, 2002. ISBN 0-201-36118-3.

SEIF, M. et al. *Temperature-Dependent Material Modeling for Structural Steels: Formulation and Application*. National Institute of Standards and Technology (NIST), 2016. Disponível em: <https://doi.org/10.6028/NIST.TN.1907>.

SEYED-AGHAZADEH, B.; EDRAKI, M.; MODARRES-SADEGHI, Y. Effects of boundary conditions on vortex-induced vibration of a fully submerged flexible cylinder. *Experiments in Fluids*, v. 60, p. 38, 2019. Disponível em: <https://doi.org/10.1007/s00348-019-2681-x>.

SHABANA, A. A. *Vibration of Discrete and Continuous Systems*. Second edition. New York: Springer-Verlag, 1997. (Mechanical Engineering Series). ISBN 978-1-4612-8474-1. Disponível em: <https://doi.org/10.1007/978-1-4612-4036-5>.

SIGRIST, J.-F. *Fluid-Structure Interaction: An Introduction to Finite Element Coupling*. 1. ed. [S.l.]: Wiley, 2012. ISBN 978-1-119-95227-5.

SIGRIST, J.-F. *Fluid-Structure Interaction: An Introduction to Finite Element Coupling*. Hoboken, NJ, USA: Wiley, 2015. ISBN 978-1-119-95227-5.

SILVA, M. L. F. da et al. Comparative study between 1-way and 2-way coupled fluid-structure interaction in numerical simulation of aortic arch aneurysms. *An Acad Bras Cienc*, v. 95, n. Suppl. 1, p. e20210859, 2023. Disponível em: <https://doi.org/10.1590/0001-3765202320210859>.

SO, R. M. C. et al. Free-stream turbulence effects on vortex-induced vibration and flow-induced force of an elastic cylinder. *Journal of Fluids and Structures*, v. 24, n. 4, p. 481–495, 2008. Disponível em: <https://doi.org/10.1016/j.jfluidstructs.2007.11.001>.

SONG, Y. et al. Modelling of wake-induced vibration of a long-span bridge with separated parallel nonidentical decks. *Structures*, v. 69, p. 107427, 2024. Disponível em: <https://doi.org/10.1016/j.istruc.2024.107427>.

SOUZA, P. R. C. *Modelagem Matemática e Computacional da Interação Fluido-Estrutura Utilizando Elementos de Placa e a Metodologia LES em um Ambiente de Malha Adaptativa*. Tese (PhD thesis) — Federal University of Uberlândia, Uberlândia, MG, Brazil, 2020.

SOUZA, P. R. C. et al. Multi-phase fluid-structure interaction using adaptive mesh refinement and immersed boundary method. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 2022.

SOZA, T. de. *Éléments de plaque : modélisations DKT, DST, DKTG*. [S.l.], 2016. Disponível em: <https://www.code-aster.org>.

STIVAL, L. J. et al. Wake modeling and simulation of an experimental wind turbine using large eddy simulation coupled with immersed boundary method alongside a dynamic adaptive mesh refinement. *Energy Conversion and Management*, Elsevier, v. 268, p. 115938, 2022. Disponível em: <https://doi.org/10.1016/j.enconman.2022.115938>.

SUMER, B. M.; FREDSOE, J. *Hydrodynamics Around Cylindrical Structures*. Revised edition. World Scientific Publishing, 2006. Disponível em: <https://doi.org/10.1142/6133>.

TIMOSHENKO, S.; GOODIER, J. N. *Theory of Elasticity*. New York, Toronto, London: McGraw-Hill Book Company, Inc., 1951.

VASCONCELLOS, L. G. S. *Advanced Computational Fluid Dynamics in Bloodstain Pattern Analysis: Investigating Non-Newtonian Properties for Forensic Applications*. 179 p. Dissertação (Master's Dissertation) — Federal University of Uberlândia, Uberlândia/MG, Brazil, 2024.

VERZICCO, R. Immersed boundary methods: Historical perspective and future outlook. *Annual Review of Fluid Mechanics*, v. 55, p. 129–155, 2023.

WANG, H. et al. Study of fluid-elastic instability of the spatially bent tube structure in fast reactor intermediate heat exchanger. *Progress in Nuclear Energy*, v. 173, p. 105254, 2024. Disponível em: <https://doi.org/10.1016/j.pnucene.2024.105254>.

WANG, H. et al. Study of fluid-elastic instability of the spatially bent tube structure in fast reactor intermediate heat exchanger. *Progress in Nuclear Energy*, v. 173, p. 105254, 2024. Disponível em: <https://doi.org/10.1016/j.pnucene.2024.105254>.

WANG, P.; WONG, C. W.; ZHOU, Y. Turbulent intensity effect on axial-flow-induced cylinder vibration in the presence of a neighboring cylinder. *Journal of Fluids and Structures*, v. 85, p. 77–93, 2019. Disponível em: <https://doi.org/10.1016/j.jfluidstructs.2019.01.001>.

WANG, Z.; FAN, J.; LUO, K. Combined multi-direct forcing and immersed boundary method for simulating flows with moving particles. *International Journal of Multiphase Flow*, v. 34, n. 3, p. 283–302, 2008.

WIJESOORIYA, K. et al. Comparison between an uncoupled one-way and two-way fluid structure interaction simulation on a super-tall slender structure. *Engineering Structures*, v. 229, p. 111636, 2021. Disponível em: <https://doi.org/10.1016/j.engstruct.2020.111636>.

WILLIAMSON, C.; GOVARDHAN, R. Vortex-induced vibrations. *Annual Review of Fluid Mechanics*, v. 36, p. 413–455, 2004. Disponível em: <https://doi.org/10.1146/annurev.fluid.36.050802.122128>.

WILSON, R. J. *Introduction to Graph Theory*. 4th. ed. [S.l.]: Addison-Wesley, 1996. ISBN 978-0582249936.

WU, D. et al. Experimental and computational investigations of thermal modal parameters for a plate-structure under 1200°c high temperature environment. *Measurement*, Elsevier, v. 94, p. 80–91, 2016. Disponível em: <https://doi.org/10.1016/j.measurement.2016.07.078>.

ZHAO, X. et al. Learning experimental data to predict fluid-elastic instability and optimize configuration of tube arrays. *Annals of Nuclear Energy*, v. 208, p. 110754, 2024. Disponível em: <https://doi.org/10.1016/j.anucene.2024.110754>.

ZIADA, S.; OENGÖREN, A. Vorticity shedding and acoustic resonance in an in-line tube bundle part i: Vorticity shedding. *Journal of Fluids and Structures*, v. 6, n. 3, p. 271–292, 1992.

A Introducing Code_Aster, Salome, and Salome-Meca

This appendix presents three tools that enable numerical engineering analyses: Code_Aster, Salome, and Salome-Meca. They can be used in an integrated manner, covering the entire simulation process: from geometry creation or import to structural, thermal, and dynamic analyses.

A.1 Code_Aster

Code_Aster is an open-source software for finite element analysis (FEA), initially developed by the Research and Development department at Électricité de France (EDF). It was released under the GNU GPL license in 2001, becoming a robust tool for the simulation of structural, thermal, and dynamic problems in various industrial and academic sectors (EDF, 2024).

A.1.1 Main Features

- **FEM-Based (Finite Element Method)**: performs the calculation of deformations, stresses, energy, and material states;
- **Static and Dynamic Simulations**: supports linear and nonlinear analyses, including fracture, fatigue, and crack propagation phenomena;
- **Thermomechanics**: enables steady-state and transient thermal conduction analyses;
- **Associated Phenomena**: covers fluid-structure interaction, acoustics, metallurgy, concrete hydration, and drying.
- **Multidisciplinary Interactions**: internal coupling (mechanics, thermomechanics, acoustics) and external integration with tools such as Code_Saturne (CFD), SYRTHES (thermal analyses), and EUROPLEXUS (fast dynamics);
- **Use of Python**: the scripting language allows for automation and greater flexibility in creating analysis cases;
- **Open-Source Culture**: a large community of developers and researchers, offering documentation and ongoing contributions.

A.1.2 Modeling and Solution Types

- **Geometric Nonlinearities**: large displacements and rotations;
- **Material Nonlinearities**: plasticity, creep, metallurgical effects, concrete hydration, and progressive damage;
- **Dynamic Analyses**: modal (with or without damping), transient and harmonic responses, seismic, and fatigue simulations;
- **Geotechnical and Civil Engineering Analyses**: constitutive models for concrete and geomaterials, fracture, and damage.

A.1.3 Computational Capabilities

- **Finite Element Library**: around 400 elements for mechanical, thermal, and physical coupling analyses;
- **Advanced Solvers**: linear methods (LDLT, MUMPS, PCG) and nonlinear (Newton and iterative), as well as explicit and implicit integrations for dynamic analyses.

A.1.4 Usage Environment

- **Integration with Salome-Meca**: provides a graphical environment for pre-processing, calculation, and post-processing;
- **Calculation Automation**: support for Python scripts to create and manage analysis cases;
- **GPL License**: promotes collaboration and continuous development, with more than 5,000 annual downloads.

A.1.5 Study Reference

One of the main references for beginners is the book *Beginning with Code_Aster* by Aubry (2019), which covers everything from geometry and mesh preparation (using Gmsh) to calculation execution and results visualization.

A.2 Salome

Salome is a generic pre-processing and post-processing platform, mainly developed by the Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA) and EDF. It

provides CAD/CAE resources, as well as meshing and post-processing tools (EDF; CEA, 2013).

A.2.1 Main Functionalities

- **3D CAD Modeling**: the GEOM module for creating and handling complex geometries;
- **Advanced Meshing (SMESH)**: generation of structured and unstructured meshes, compatible with IGES, STEP, and BREP formats;
- **Solver Integration**: facilitates multiphysics calculations and couplings between different codes (e.g., Code_Aster, Code_Saturne);
- **Post-Processing and Visualization (ParaViS)**: based on ParaView, supporting animations, slicing, and quantitative analysis;
- **MED Format**: ensures interoperability between components (meshes, physical fields, etc.);
- **HPC Support**: execution of processes on distributed networks and high-performance clusters;
- **LGPL License**: open development and external collaborations.

A.3 Salome-Meca

Salome-Meca is a package that integrates Code_Aster into the Salome platform, providing a unified graphical interface for model construction, mesh generation, analysis execution, and result visualization (EDF; CEA, 2013).

A.3.1 Main Functionalities

- **EFICAS**: a module that assists in the guided creation of Code_Aster command files;
- **Full Integration with Salome**: enables a seamless workflow from geometry to mesh and post-processing in the same environment;
- **Compatibility with External Tools**: such as Gmsh and other mesh generators;
- **User-Friendly Environment**: simplifies the learning curve compared to using Code_Aster directly via scripts.

A.4 Final Remarks

Salome-Meca is the complete solution for those who wish to harness the computing power of Code_Aster in an intuitive graphical environment. It streamlines model creation and analysis setup, making it an essential tool for engineers and researchers who seek efficiency in advanced structural and thermomechanical engineering analysis. Together, Code_Aster, Salome, and Salome-Meca form a complete toolchain for engineering analyses, from geometry design and meshing to results post-processing, as shown in Tab. 13.

Software	**Main Function**
Code_Aster	Finite element solver for high-complexity structural, thermal, and dynamic analyses.
Salome	Open CAD/CAE platform for geometric modeling, meshing, and post-processing.
Salome-Meca	Integration of Code_Aster into Salome, providing a graphical environment for advanced simulations.

Table 13 – Summary of Code_Aster, Salome, and Salome-Meca softwares functionalities.

The open-source nature and the strong community surrounding these softwares enable continuous improvements, making them solid choices for professionals and researchers seeking robustness, reliability, and flexibility in numerical simulations.