

THIAGO FERNANDO SANTIAGO DE FREITAS

**MODELAGEM MATEMÁTICA E COMPUTACIONAL DE
ESCOAMENTOS TURBULENTOS USANDO A
METODOLOGIA MISTA PSEUDOSPECTRAL DE
FOURIER E FRONTEIRA IMERSA**



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA

2026

THIAGO FERNANDO SANTIAGO DE FREITAS

**MODELAGEM MATEMÁTICA E COMPUTACIONAL DE
ESCOAMENTOS TURBULENTOS USANDO A METODOLOGIA
MISTA PSEUDOSPECTRAL DE FOURIER E FRONTEIRA IMERSA**

Tese apresentada ao Programa de Pós-Graduação em Engenharia Mecânica da Universidade Federal de Uberlândia, como parte dos requisitos para a obtenção do título de **DOUTOR EM ENGENHARIA MECÂNICA**.

Área de concentração: Transferência de Calor e Mecânica dos Fluidos.

Orientador: Prof. Dr. Aristeu da Silveira Neto

Coorientador: Prof. Dr. Felipe Pamplona Mariano

Uberlândia, MG

2026

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

F866 Freitas, Thiago Fernando Santiago de, 1996-
2026 Modelagem Matemática e Computacional de Escoamentos
Turbulentos Usando a Metodologia Mista Pseudoespectral de
Fourier e Fronteira Imersa [recurso eletrônico] / Thiago Fernando
Santiago de Freitas. - 2026.

Orientador: Aristeu da Silveira Neto.

Coorientador: Felipe Pamplona Mariano.

Tese (Doutorado) - Universidade Federal de Uberlândia, Pós-
graduação em Engenharia Mecânica.

Modo de acesso: Internet.

DOI <http://doi.org/10.14393/ufu.te.2026.614>

Inclui bibliografia.

Inclui ilustrações.

1. Engenharia mecânica. I. Silveira Neto, Aristeu da ,1955-,
(Orient.). II. Mariano, Felipe Pamplona,1981-, (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:	Tese de Doutorado, nº 424, PPGEM				
Data:	04/08/2026	Hora de início:	09:30	Hora de encerramento:	13:00
Matrícula:	12323EMC011				
Nome do Discente:	Thiago Fernando Santiago de Freitas				
Título do Trabalho:	Modelagem Matemática e Computacional de Escoamentos Turbulentos Usando a Metodologia Mista Pseudoespectral de Fourier e Fronteira Imersa				
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:	Não se aplica				

Reuniu-se por videoconferência a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Engenharia Mecânica, assim composta: Prof. Dr. Leandro Franco de Souza - USP; Prof. Dr. Elisan dos Santos Magalhães - ITA; Prof. Dr. Gilmar Guimarães - UFU; Prof. Dr. Francisco José de Souza - UFU; Prof. Dr. Felipe Pamplona Mariano - UFG (Co-orientador) e Prof. Dr. Aristeu da Silveira Neto - UFU; orientador do candidato.

Iniciando os trabalhos, o presidente da mesa, Prof. Dr. Aristeu da Silveira Neto, apresentou a Comissão Examinadora e o candidato, 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 presidente concedeu a palavra, pela ordem sucessivamente, aos examinadores, que passaram a arguir o candidato. Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o candidato:

Aprovado.

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

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 **Aristeu da Silveira Neto, Professor(a) do Magistério Superior**, em 04/08/2026, às 13:10, 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 **Felipe Pamplona Mariano, Usuário Externo**, em 04/08/2026, às 13:10, 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 **Francisco José de Souza, Professor(a) do Magistério Superior**, em 04/08/2026, às 13:10, 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 **Gilmar Guimarães, Professor(a) do Magistério Superior**, em 04/08/2026, às 13:11, 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 **ELISAN DOS SANTOS MAGALHAES, Usuário Externo**, em 04/08/2026, às 13:12, 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 **LEANDRO FRANCO DE SOUZA, Usuário Externo**, em 04/08/2026, às 13:16, 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 **7489618** e o código CRC **3B8291D8**.

À minha mãe, eu não chegaria tão longe sem o seu incentivo.

AGRADECIMENTOS

Ao fim dessa etapa acadêmica, a minha emoção principal, dentre inúmeras, é gratidão. Em especial, gostaria de agradecer à minha mãe, que sempre me apoiou e praticamente foi minha psicóloga ao longo dos inúmeros episódios de incertezas ao longo da minha pós-graduação. Também quero agradecer ao meu irmão Joaquim Neto pelas piadas questionáveis, ao meu irmão Pedro Henrique pelas nossas conversas profundas e à minha irmã pela saudade sempre presente. Enfim, sou grato a toda a minha família que sempre esteve comigo nessa caminhada.

Também quero expressar um muito obrigado à minha primeira orientadora, Andreia Nascimento, que me introduziu à pesquisa científica e construiu comigo as bases para o profissional que sou.

Não poderia deixar de fazer um agradecimento especial aos meus orientadores, Aristeu da Silveira Neto e Felipe Pamplona Mariano. A pesquisa científica na pós-graduação pode ser solitária, mas nunca me senti dessa forma. Afinal, sempre pude contar com os meus dois orientadores.

As excelentes conversas que tive com o Aristeu em sua sala sempre foram esclarecedoras e me fizeram refletir sobre as definições mais básicas dentro da disciplina de mecânica dos fluidos. Quando me recordo delas, sempre tenho um misto de gratidão, saudades e nostalgia.

Os *insights* do Felipe sobre a metodologia IMERSPEC e as suas ideias sobre como proceder quando nada dava certo são recordações valiosas que guardo comigo. Nesse período da pós-graduação em que raramente nos encontrávamos pessoalmente, ele sempre esteve a um clique de distância.

Um agradecimento também à todos os meus colegas do MFLab que me ajudaram ao compartilhar os nossos desafios e frustrações. Às vezes, apenas desabafar é o suficiente.

Do mesmo modo, gostaria de destacar o apoio que tive dos Professores Elisan dos Santos Magalhães e Francisco José de Souza com a programação utilizando GPUs.

Também sou muito grato à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, CAPES, pelo suporte ao longo do meu doutorado. Gostaria de salientar que o presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

Por fim, ainda gostaria de estender um muito obrigado à Faculdade de Engenharia Mecânica (FEMEC) da Universidade Federal de Uberlândia (UFU), juntamente com a Coordenação do seu programa de Pós-Graduação (POSFEMEC), ao Laboratório de Mecânica dos Fluidos (MFLab), à Escola de Engenharia: Elétrica, Mecânica e de Computação (EMC) da Universidade Federal de Goiás (UFG) e, em especial, ao seu Laboratório Multiusuário de Computação de

Alto Desempenho (LaMCAD), pelo suporte e infraestrutura necessários para a realização desse trabalho.

RESUMO

A turbulência em escoamentos de fluidos é um dos fenômenos físicos mais complexos e relevantes da engenharia, e sua modelagem computacional demanda metodologias de alta fidelidade e eficiência computacional. Nesse contexto, na presente tese propõe-se o desenvolvimento, a verificação e a validação de dois novos códigos computacionais tridimensionais baseados na metodologia IMERSPEC, hibridização do Método Pseudoespectral de Fourier (MPF) e do Método da Fronteira Imersa (MFI), para de modelar escoamentos laminares e turbulentos complexos com a presença de paredes. Os códigos desenvolvidos, denominados IMERSPEC MPI e IMERSPEC CUDA, distinguem-se pelas técnicas de paralelização adotadas: no primeiro emprega-se a biblioteca MPI com paralelização em duas das três direções do escoamento, enquanto no segundo explora-se o paralelismo heterogêneo CPU/GPU por meio da API CUDA, realizando os cálculos integralmente na unidade de processamento gráfico. Ambos os códigos foram verificados computacionalmente pelo Método de Soluções Manufaturadas (MSM) para o MPF isolado e para o MPF com MFI em configurações coincidentes e não coincidentes, apresentando exatidão espectral e quarta ordem de convergência, respectivamente. Adicionalmente, propôs-se e investigou-se o Método de Imposição Espectral Direta (MIED), alternativa integralmente espectral ao MDFM para malhas não coincidentes, cujos resultados indicaram exatidão inferior ao MDFM com custo computacional similar. A validação física de ambos os códigos foi realizada mediante a modelagem do escoamento laminar sobre o degrau, um caso de modelagem de um escoamento complexo clássico, com resultados consistentes com os dados dos experimentos materiais. Comparando-se os dois códigos para as mesmas condições de simulação em diferentes *hardwares*, o IMERSPEC CUDA apresentou desempenho superior ao IMERSPEC MPI em todas as configurações de malha avaliadas, com uma redução de custo computacional de aproximadamente vinte e nove vezes para a modelagem laminar. Enfatiza-se que esse resultado é obtido quando se compara a execução do código IMERSPEC MPI em um *cluster* com 64 processos e o IMERSPEC CUDA executado junto a placa NVIDIA A100. Por outro lado, salienta-se que foi identificada a necessidade de determinação de parâmetros ótimos de execução do código IMERSPEC CUDA de acordo com a GPU a ser executada, *i.e.*, os parâmetros ótimos de execução são dependentes do *hardware* usado. Com base na seleção do código com melhor performance, os desenvolvimentos subsequentes para a modelagem usando Simulação de Grandes Estruturas (LES) concentraram-se exclusivamente no IMERSPEC CUDA. Na modelagem turbulenta, investigou-se a interação entre a metodologia IMERSPEC e a abordagem LES, identificando-se e corrigindo-se aspectos da implementação que impediam a correta reprodução do escoamento turbulento, tais como a reestruturação do domínio complementar, a adoção de filtro exponencial para o processo de *dealiasing* e a definição das condições de contorno de topo do domínio útil de modelagem. Três modelos funcionais de fechamento submalha foram implementados e comparados no escoamento turbulento sobre o degrau com expansão brusca e com número de Reynolds $Re_H = 5.100$: o modelo de Smagorinsky clássico

com funções de amortecimento de Van Driest, o modelo WALE (do inglês, *Wall-Adapting Local Eddy-viscosity*) e o modelo dinâmico de Germano-Lilly^{1,2}. Os resultados obtidos com as malhas de $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos de colocação foram comparados com dados computacionais e materiais de referência. O modelo WALE apresentou o melhor desempenho global, com erro de apenas 0,5% na estimativa do comprimento de recolamento da camada limite, além do menor custo computacional entre os três modelos avaliados, constituindo-se como a melhor escolha para a presente modelagem ainda que este modelo submalha apresente perfis de velocidade média e componentes da tensão média de Boussinesq-Reynolds com significante discrepâncias dos trabalhos de referência. Com o modelo dinâmico de Germano-Lilly obteve-se resultados ligeiramente superiores ao modelo de Smagorinsky clássico, com erro de 11% e 13% no comprimento de recolamento, respectivamente, com custo computacional maior. O IMERSPEC CUDA com abordagem LES e modelo dinâmico de Germano-Lilly, o modelo mais caro computacionalmente, reproduziu 450 s de tempo físico em menos de 72 horas na GPU NVIDIA A100, demonstrando a viabilidade da metodologia para simulações de alta ordem em tempo competitivo.

Palavras-chave: método pseudoespectral de Fourier; método da fronteira imersa; paralelização MPI; paralelização CPU/GPU; escoamento sobre o degrau; modelo de Smagorinsky com funções de Van Driest; modelo WALE; modelo dinâmico de Germano-Lilly.

ABSTRACT

Turbulence in fluid flows is one of the most complex and relevant physical phenomena in engineering, and its computational modeling demands high-fidelity and computationally efficient methodologies. In this context, the present thesis proposes the development, verification, and validation of two novel three-dimensional computational codes based on the IMERSPEC methodology, a hybridization of the Fourier Pseudospectral Method (MPF) and the Immersed Boundary Method (MFI), for the simulation of laminar and turbulent complex flows in the presence of walls. The developed codes, referred to as IMERSPEC MPI and IMERSPEC CUDA, are distinguished by their parallelization strategies: the former employs the MPI library with parallelization along two of the three flow directions, while the latter exploits heterogeneous CPU/GPU parallelism through the CUDA API, performing all computations entirely on the graphics processing unit. Both codes were computationally verified using the Method of Manufactured Solutions (MSM) for the isolated MPF and for the MPF coupled with the MFI in coincident and non-coincident grid configurations, achieving spectral accuracy and fourth-order convergence, respectively. Additionally, the Spectral Direct Imposition Method (MIED) was proposed and investigated as a fully spectral alternative to the MDFM for non-coincident grids, whose results indicated lower accuracy than the MDFM at comparable computational cost. The physical validation of both codes was carried out through the simulation of laminar flow over a backward-facing step, a classical complex flow benchmark, with results consistent with experimental reference data. Comparing the two codes under identical simulation conditions on different hardware configurations, the IMERSPEC CUDA outperformed the IMERSPEC MPI across all evaluated mesh sizes, achieving a computational cost reduction of approximately twenty-nine times for the laminar case. It is emphasized that this result is obtained when comparing the IMERSPEC MPI executed on a cluster with 64 processes and the IMERSPEC CUDA running on the NVIDIA A100 GPU. On the other hand, it is noted that optimal execution parameters for the IMERSPEC CUDA code must be determined according to the GPU to be used, i.e., the optimal execution parameters are hardware-dependent. Based on the selection of the higher-performance code, subsequent developments for Large Eddy Simulation (LES) modeling were conducted exclusively with the IMERSPEC CUDA. In the turbulent flow simulations, the interaction between the IMERSPEC methodology and the LES approach was investigated, identifying and correcting implementation aspects that prevented the correct reproduction of turbulent flow, such as the restructuring of the complementary domain, the adoption of an exponential filter for the dealiasing procedure, and the definition of the top boundary conditions of the computational domain. Three functional subgrid-scale (SGS) closure models were implemented and compared for turbulent flow over a backward-facing step at Reynolds number $Re_H = 5100$: the classical Smagorinsky model with Van Driest damping functions, the WALE model (*Wall-Adapting Local Eddy-viscosity*), and the Germano-Lilly dynamic model^{1,2}. Results obtained with meshes of $32 \times 128 \times 256$ and $64 \times 256 \times 512$ collocation points were compared against computational and experimental

reference data. The WALE model demonstrated the best overall performance, with an error of only 0,5% in the reattachment length estimation and the lowest computational cost among the three evaluated models, making it the most suitable choice for the present simulations, although this subgrid-scale model presents mean velocity profiles and Boussinesq-Reynolds mean stress tensor components with significant discrepancies relative to reference data. The Germano-Lilly dynamic model yielded results slightly superior to those of the classical Smagorinsky model, with reattachment length errors of 11% and 13%, respectively, at a higher computational cost. The IMERSPEC CUDA with the LES approach and the Germano-Lilly dynamic model, the most computationally demanding configuration, reproduced 450 s of physical time in less than 72 hours on the NVIDIA A100 GPU, demonstrating the viability of the methodology for high-order simulations within competitive computational time.

Keywords: Fourier pseudospectral method; immersed boundary method; MPI parallelization; CPU/GPU parallelization; backward-facing step flow; Smagorinsky model with Van Driest damping functions; WALE model; Germano-Lilly dynamic model.

LISTA DE ILUSTRAÇÕES

Figura 1 – Esquema de implementação de MFI com interface nítida.	15
Figura 2 – Estruturas turbilhonares sendo amortecidas pelo domínio complementar em modelagem de jato livre circular turbulento com o IMERSPEC.	18
Figura 3 – Esquema da abordagem LES representada em um espectro $E(k)$	27
Figura 4 – Esquema de decomposição do sinal genérico $\phi(x_i, t)$	28
Figura 5 – Esquema de implementação computacional de Smagorinsky clássico com a função de Van Driest.	35
Figura 6 – Esquema de implementação computacional WALE.	37
Figura 7 – Esquema de implementação computacional do modelo dinâmico de Germano-Lilly ^{1,2}	40
Figura 8 – Esquema dos domínios Ω e Γ na modelagem de um escoamento qualquer. . .	41
Figura 9 – Diagrama de blocos do MDFM/MFI para pontos de Γ coincidentes.	45
Figura 10 – Diagrama de blocos MDFM/MFI para pontos de Γ não coincidentes.	46
Figura 11 – Gráfico da função $\phi_4(r)$	48
Figura 12 – Esquema do plano π com $\widehat{u}_i$ e k_i	51
Figura 13 – Esquema do plano π em relação aos termos das equações de <i>Momentum Linear</i> . .	52
Figura 14 – Esquema de paralelização do tipo memória compartilhada.	55
Figura 15 – Esquema de paralelização do tipo passagem de memória.	56
Figura 16 – Esquema de paralelização heterogêneo CPU/GPU.	57
Figura 17 – Função amortecedora do método <i>buffer</i>	61
Figura 18 – Impacto da zona de forçagem na exatidão da modelagem IMERSPEC.	62
Figura 19 – Esquema de <i>aliasing</i>	63
Figura 20 – Fluxograma simplificado da metodologia IMERSPEC.	68
Figura 21 – Condições iniciais para a verificação numérica.	71
Figura 22 – Curva de L_2 para u , v , w e p na verificação de MPF.	72
Figura 23 – <i>Speedup</i> de diferentes versões da FFTE para a Verificação do MPF.	73
Figura 24 – Esquema de implementação.	74
Figura 25 – Verificação da solução numérica com malha de 16^3 pontos de colocação. . .	75
Figura 26 – <i>Speedup</i> para verificação do MPF com MDFM/MFI coincidente.	75
Figura 27 – Esquema de implementação.	76
Figura 28 – Verificação da solução numérica com malha de 16^3 pontos de colocação. . .	77
Figura 29 – Análise da Convergência do IMERSPEC MPI para Γ não coincidente.	78
Figura 30 – <i>Speedup</i> para Verificação do MPF com MDFM/MFI não coincidente.	78
Figura 31 – Distribuição de Γ pontos de acordo com 4,8 e 32 processos.	79
Figura 32 – Esquema de implementação.	81

Figura 33 – Plano yz do domínio físico para a componente w e TF com discretização divergente em $t_a = 100$	82
Figura 34 – Perfis em $14h$ e $t_a = 100$ para TF antissimétrico com $4 \times 32 \times 1024$ pontos de colocação e para 3 malhas diferentes com TF divergente.	82
Figura 35 – Linhas de corrente no plano yz para $t_a = 500$ para $4 \times 32 \times 1024$ pontos de colocação.	84
Figura 36 – Perfis de velocidade w/U_∞	85
Figura 37 – Comparação da distribuição de C_f para a simulação 3D e a pseudo-3D.	86
Figura 38 – Curva de L_2 para u, v, w e p para a verificação de MPF em IMERSPEC CUDA.	89
Figura 39 – Custo computacional por número de <i>threads</i> em cada bloco.	90
Figura 40 – Custo computacional por número de operações em cada thread.	91
Figura 41 – Análise de parâmetros de performance do IMERSPEC CUDA na placa GPU A100.	93
Figura 42 – Normas L_2 referentes a verificação com fronteira imersa coincidente de IMERSPEC CUDA.	95
Figura 43 – Comparação de curvas L_2 de IMERSPEC MPI e IMERSPEC CUDA	96
Figura 44 – Comparação de curvas de convergência de IMERSPEC MPI e IMERSPEC CUDA.	97
Figura 45 – Resultados Computacionais para MIED com 16^3 pontos de colocação.	99
Figura 46 – Comparação de curvas de convergência de MDFM e MIED	100
Figura 47 – Comparação da exatidão dos códigos IMERSPEC MPI e CUDA	101
Figura 48 – Esquema do Experimento Material.	104
Figura 49 – Fluxograma atualizado da metodologia IMERSPEC para modelagem de escoamentos turbulentos.	106
Figura 50 – Esquema de modelagem do Escoamento Turbulento.	107
Figura 51 – $E(x_i, f)$ para os distintos processos de <i>dealising</i>	109
Figura 52 – Perfis Verticais em $x = 2H$ e $z = 6H$	111
Figura 53 – Critério Q (2,75) em $t_a = 900$ para malha de $64 \times 256 \times 512$ pontos de colocação com o modelo submalha de Smagorinsky com Van Driest.	113
Figura 54 – Vista lateral dos campos de $\ \vec{\omega}\ $ (a) e ν_{eff}/ν (b) relativos ao modelo de fechamento submalha de Smagorinsky com Funções de amortecimento de Van Driest em $t_a = 900$ com $64 \times 256 \times 512$ pontos de colocação.	114
Figura 55 – Perfis verticais de $\langle \bar{w}/U_\infty \rangle$ para o modelo de Smagorinsky com funções de Van Driest.	115
Figura 56 – Perfis Verticais de $\langle \bar{v}/U_\infty \rangle$ para o modelo de Smagorinsky com funções de Van Driest.	116
Figura 57 – Componente $\langle \bar{w}^2/U_\infty^2 \rangle$ do tensor de Boussinesq-Reynolds para o modelo de Smagorinsky com funções de Van Driest.	116

Figura 58 – Componente $\langle \bar{v}'^2/U_\infty^2 \rangle$ do tensor de Boussinesq-Reynolds para o modelo de Smagorinsky com funções de Van Driest.	117
Figura 59 – $\langle C_f \rangle$ para o modelo de Smagorinsky com funções de Van Driest.	118
Figura 60 – Espectros com Modelo Clássico de Smagorinsky.	119
Figura 61 – Vista lateral de $\ \vec{\omega}\ $ (a) e ν_{eff}/ν (b) relativos ao modelo de fechamento submalha WALE em $t_a = 900$ com $64 \times 256 \times 512$ pontos de colocação.	120
Figura 62 – Perfis Verticais de $\langle \bar{w}/U_\infty \rangle$ para o modelo WALE.	121
Figura 63 – Perfis Verticais de $\langle \bar{v}/U_\infty \rangle$ para o modelo WALE.	121
Figura 64 – Perfis verticais de $\langle \bar{w}'^2/U_\infty^2 \rangle$ para o modelo WALE.	122
Figura 65 – Perfis verticais de $\langle \bar{v}'^2/U_\infty^2 \rangle$ para o modelo WALE.	123
Figura 66 – Distribuições de $\langle C_f \rangle$ para o modelo WALE.	123
Figura 67 – $E(x_i, f)$ para as diferentes malhas computacionais com modelo WALE.	124
Figura 68 – Vista lateral de $\ \vec{\omega}\ $ (a), C_d (b) e ν_{eff}/ν (c) relativos ao modelo de fechamento submalha de Smagorinsky dinâmico em $t_a = 900$ com $64 \times 256 \times 512$ pontos de colocação.	127
Figura 69 – Perfis Verticais de $\langle \bar{w}/U_\infty \rangle$ para o modelo dinâmico de Smagorinsky.	128
Figura 70 – Perfis Verticais de $\langle \bar{v}/U_\infty \rangle$ para o modelo dinâmico de Smagorinsky.	128
Figura 71 – Perfis verticais de $\langle \bar{w}'^2/U_\infty^2 \rangle$ para o modelo dinâmico de Smagorinsky.	129
Figura 72 – Perfis verticais de $\langle \bar{v}'^2/U_\infty^2 \rangle$ para o modelo dinâmico de Smagorinsky.	129
Figura 73 – Distribuições de $\langle C_f \rangle$ para o modelo de Smagorinsky dinâmico.	130
Figura 74 – $E(x_i, f)$ para as diferentes malhas computacionais com Modelagem Dinâmica de Smagorinsky.	131
Figura 75 – Comparação de $E(x_i, f)$ utilizando a malha com $64 \times 256 \times 512$ pontos de colocação para os modelos submalha implementados.	132
Figura 76 – Critério Q (2,75) em $t_a = 900$ de dois modelos submalha com a malha de $64 \times 256 \times 512$ pontos de colocação.	133

LISTA DE TABELAS

Tabela 1 – Síntese comparativa de trabalhos com paralelização GPGPU abordados.	24
Tabela 2 – Parâmetros de MRK46.	67
Tabela 3 – Propriedades da Simulação Relacionada a Verificação MPF.	71
Tabela 4 – Propriedades das simulações do escoamento sobre degrau.	81
Tabela 5 – Coordenadas de interesse e custo computacional das simulações.	84
Tabela 6 – Propriedades de simulação relacionadas a verificação IMERSPEC CUDA.	89
Tabela 7 – Custo computacional, em segundos, da verificação de MPF para IMERSPEC MPI e IMERSPEC CUDA.	94
Tabela 8 – Custo computacional, em segundos, da verificação com fronteira imersa coincidente para IMERSPEC MPI e IMERSPEC CUDA.	95
Tabela 9 – Custo computacional, em segundos, da verificação com fronteira imersa não coincidente para IMERSPEC MPI e IMERSPEC CUDA.	98
Tabela 10 – Comparativo de MDFM e MIED em relação aos seus custos computacionais, em segundos.	100
Tabela 11 – Propriedades das simulações do escoamento sobre degrau.	108
Tabela 12 – Possíveis condições de contorno no topo do domínio físico.	110

LISTA DE ABREVIATURAS E SIGLAS

AMD	Modelo de dissipação mínima anisotrópica, do inglês <i>Anisotropic Minimum-Dissipation</i>
API	Interface de programação de aplicações, do inglês <i>Application Programming Interface</i>
BFS	Escoamento sobre o degrau retroativo, do inglês <i>Backward-Facing Step</i>
CFL	Condição de estabilidade de Courant-Friedrichs-Lewy
CFD	Dinâmica dos Fluidos Computacional, do inglês <i>computational fluid dynamics</i>
CPU	Unidade central de processamento, do inglês <i>Central Processing Unit</i>
CUDA	Arquitetura de dispositivo de computação unificada, do inglês <i>Compute Unified Device Architecture</i>
cuFFT	Biblioteca de transformada rápida de Fourier para GPU da NVIDIA, do inglês <i>CUDA Fast Fourier Transform</i>
DFM	Método de imposição direta, do inglês <i>Direct Forcing Method</i>
DFT	Transformada discreta de Fourier, do inglês <i>Discrete Fourier Transform</i>
DNS	Simulação computacional direta, do inglês <i>Direct Numerical Simulation</i>
FFT	Transformada rápida de Fourier, do inglês <i>Fast Fourier Transform</i>
FFTE	Biblioteca de transformada rápida de Fourier em Fortran, do inglês <i>Fast Fourier Transform subroutines written in Fortran</i>
FFTW	Biblioteca de transformada rápida de Fourier, do inglês <i>Fastest Fourier Transform in the West</i>
GPGPU	Unidade de processamento gráfico de propósito geral, do inglês <i>General Purpose Graphics Processing Unit</i>
GPU	Unidade de processamento gráfico, do inglês <i>Graphical Processing Unit</i>
IBM	Método da fronteira imersa, do inglês <i>Immersed Boundary Method</i>
IDFT	Transformada discreta inversa de Fourier, do inglês <i>Inverse Discrete Fourier Transform</i>

IMERSPEC	Metodologia híbrida da junção do Método Pseudoespectral de Fourier e do Método da Fronteira Imersa
LaMCAD	Laboratório de Modelagem e Computação Científica Avançada
LDV	Velocimetria a laser Doppler, do inglês <i>Laser Doppler Velocimetry</i>
LES	Simulação de Grandes Estruturas de Turbulência, do inglês <i>Large Eddy Simulation</i>
MDFM	Método de múltiplas imposições diretas, do inglês <i>Multi Direct Forcing Method</i>
MFI	Método da fronteira imersa
MFLab	Laboratório de Mecânica dos Fluidos da Universidade Federal de Uberlândia
MIED	Método de imposição espectral direta
MPI	Interface de passagem de mensagens, do inglês <i>Message Passing Interface</i>
MPF	Método pseudoespectral de Fourier
MSM	Método de soluções manufaturadas
NUFFT	Transformada de Fourier não uniforme, do inglês <i>Non-Uniform Fast Fourier Transform</i>
OpenMP	Interface de programação para paralelização com memória compartilhada, do inglês <i>Open Multi-Processing</i>
RANS	Equações de Navier-Stokes com média de Reynolds, do inglês <i>Reynolds Averaged Navier-Stokes</i>
RFG	Geração de escoamento aleatório, do inglês <i>Random Flow Generation</i>
RHS	Lado direito da equação, do inglês <i>Right Hand Side</i>
RK46	Método de Runge-Kutta de quarta ordem com seis passos
SGS	Submalha, do inglês <i>Sub-Grid Scale</i>
SHJAR	Dispositivo acústico para jatos quentes pequenos, do inglês <i>Small Hot Jet Acoustic Rig</i>
SI	Sistema Internacional de Unidades
SIMD	Instrução única para múltiplos dados, do inglês <i>Single-Instruction, Multiple-Data</i>

SM	Multiprocessador de GPU, do inglês <i>Streaming Multiprocessor</i>
SPIRAL	Sistema de geração de código otimizado para processamento de sinais, do inglês <i>Signal Processing Implementation Research for Adaptable Libraries</i>
UFG	Universidade Federal de Goiás
UFU	Universidade Federal de Uberlândia
URANS	Equações de Navier-Stokes com média de Reynolds transiente, do inglês <i>Unsteady Reynolds Averaged Navier-Stokes</i>
WALE	Modelo submalha de viscosidade adaptativa à parede, do inglês <i>Wall-Adapting Local Eddy-viscosity</i>
WMLES	Simulação de grandes estruturas com modelagem de parede, do inglês <i>Wall-Modelled Large-Eddy Simulation</i>

LISTA DE SÍMBOLOS

Sobrescritos

*	Parâmetro temporário
**	Parâmetro adimensional
'	Flutuação da informação em relação ao sinal filtrado
—	Filtragem do sinal (ver também Operadores)
^	Propriedade no espaço espectral de Fourier (ver também Operadores)
~	Filtragem pelo filtro teste do modelo dinâmico de Smagorinsky (ver também Operadores)
<i>SGS</i>	Relativo à submalha
+	Normalização com a dimensão característica de parede y_w
t	Relativo ao instante temporal t
$t + \Delta t$	Relativo ao próximo instante temporal
it	Número de iteração do MDFM
l	Passo do método de avanço temporal RK46
n	Relativo à solução numérico-computacional
c	Relativo à solução contínua ou manufaturada
IM	Relativo à interface imersa
$Total$	Força total sobre o ponto lagrangiano
d	Tensor deviatórico (e.g. $\bar{G}_{ij}^d$)

Letras Gregas

ρ	Massa específica do fluido
ν	Viscosidade cinemática molecular

ν_{SGS}	Viscosidade cinemática de submalha
Δ	Comprimento característico das estruturas submalha; incremento espacial da malha computacional
Δt	Incremento temporal
Δs	Incremento lagrangiano geral
δ	Delta de Kronecker
δ^*	Espessura de deslocamento da camada limite
ϕ	Informação genérica
ϕ_{ij}	Tensor de projeção espectral
λ	Microcomprimento de Taylor
ψ	Informação genérica
Ψ	Função de amortecimento do método buffer
τ	Tensor global de submalha ou tensão viscosa
τ_w	Tensão de cisalhamento na parede
ω	Vorticidade
Ω	Domínio de cálculo euleriano
Γ	Domínio de cálculo lagrangiano (interface imersa)
α_l, β_l	Constantes do método de avanço temporal RK46
ϵ	Critério de parada do MDFM
σ	Fator de superamostragem (<i>upsampling</i>) da NUFFT
v	Número de pontos de suporte compacto da função ψ na NUFFT
π	Plano hipotético perpendicular ao vetor número de onda k_i

Letras Latinas

a	Constante qualquer
u	Velocidade euleriana na direção x
v	Velocidade euleriana na direção y
w	Velocidade euleriana na direção z
$\mathbf{V}$	Vetor velocidade; $\ \mathbf{V}\ $ denota sua magnitude
p	Pressão modificada do escoamento ($p = p_e/\rho$)
p_e	Pressão estática
f_i	Termo fonte nas equações de momentum linear (forças de campo ou fronteira imersa)
F_i	Termo fonte lagrangiano
k	Número de onda escalar no domínio espectral; também limite de Nyquist quando $k = K$
k_i	Vetor número de onda na direção i
K	Energia cinética turbulenta específica; também limite de Nyquist $K = \pi/h$
k_c	Número de onda de corte da abordagem LES ($k_c = \pi/\Delta_i$)
f_c	Frequência de corte espectral
S_{ij}	Tensor taxa de deformação; $\bar{S}_{ij}$ denota sua versão filtrada
W_{ij}	Tensor taxa de rotação filtrado
$\bar{G}_{ij}^d$	Tensor deviatórico do quadrado do gradiente da velocidade filtrada (modelo WALE)
$\bar{g}_{ij}$	Tensor gradiente da velocidade filtrada ($\bar{g}_{ij} = \partial \bar{u}_i / \partial x_j$)
x	Posição geométrica
t	Tempo físico
t_a	Tempo adimensional ($t_a = tU_\infty/h$)
t_s	Tempo adimensional de início do cálculo de estatísticas
E	Distribuição espectral de energia cinética específica turbulenta

T	Tensor das tensões relativas ao segundo filtro (tensor subteste do modelo dinâmico de Smagorinsky)
L	Tensor de Leonard global
$G, \hat{G}$	Função filtro no domínio físico e espectral, respectivamente
C_s	Constante ad-hoc do modelo de Smagorinsky clássico
C_{WA}	Constante ad-hoc do modelo WALE
C_d	Constante dinâmica do modelo de Smagorinsky dinâmico (função de x_i e t)
M_{ij}	Tensor auxiliar do procedimento de Germano-Lilly
N	Número de pontos discretos (nós de colocação)
N_b	Número total de pontos lagrangianos
N_L	Número de iterações do MDFM
N_e	Número de modos de Fourier do ruído sintético RFG
L_i	Comprimento do domínio computacional na direção i
h	Altura do degrau; também incremento euleriano geral
H	Altura do degrau nas simulações turbulentas
Re	Número de Reynolds
Re_τ	Número de Reynolds de atrito ($Re_\tau = u_\tau H / \nu$)
E_r	Razão de expansão do degrau
U_∞	Velocidade característica do escoamento livre
u_τ	Velocidade de atrito (velocidade característica de regiões próximas às paredes)
y^+	Distância adimensional de parede ($y^+ = u_\tau y / \nu$)
C_f	Coefficiente de atrito de parede
Q	Critério Q para identificação de estruturas turbilhonares
sp	Speedup de paralelização
AUX	Variável auxiliar filtrada do método RK46
r	Coordenada geral da função de interpolação/distribuição φ_4

ς	Variável genérica do escoamento no domínio físico
c_j	Coeficientes da NUFFT
ω_j	Pesos de quadratura da NUFFT
I_{turb}	Intensidade turbulenta
z_r	Coordenada do ponto de recolamento da camada limite
z_s	Coordenada do ponto de descolamento da camada limite
z_{rs}	Coordenada do ponto de recolamento secundário da camada limite

Operadores

$\int$	Integral
d	Derivada total
∂	Derivada parcial
$\sum$	Somatório
$-$	Filtragem espacial do sinal (filtro em nível de malha)
$\wedge$	Transformada de Fourier de uma propriedade (domínio espectral)
$\sim$	Filtragem pelo filtro teste (modelo dinâmico de Smagorinsky)
$\ \, \ $	Norma de um tensor ou vetor
$\langle \, \rangle$	Média temporal ou espacial de uma propriedade
$*$	Produto de convolução no domínio espectral
TNL	Termo não linear das equações de momentum linear no domínio espectral

Subscritos

i, j	Índices de notação tensorial
w	Propriedade na parede
e	Propriedade estática
c	Propriedade de corte no domínio espectral

s	Relativo ao modelo de Smagorinsky clássico; também ponto de descolamento
d	Relativo ao modelo de Smagorinsky dinâmico
τ	Relativo à grandeza característica de regiões próximas às paredes
SGS	Relativo à submalha
eff	Relativo à viscosidade efetiva ($\nu_{eff} = \nu + \nu_{SGS}$)
WA	Relativo ao modelo WALE (<i>Wall-Adapting</i>)
x, y, z	Propriedade na direção x , y ou z , respectivamente
∞	Relativo à corrente livre
H	Relativo ao tamanho do degrau H
0	Condição inicial ou de referência
m	Relativo à pressão modificada
max	Valor máximo
min	Valor mínimo
A	Alias de um número de onda (k_A)
n	Relativo ao nó de colocação n
r	Relativo ao ponto de recolamento
rs	Relativo ao ponto de recolamento secundário

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Justificativa do trabalho	4
1.2	Objetivos	5
1.3	Escopo do Trabalho	6
2	REVISÃO BIBLIOGRÁFICA	8
2.1	Revisão de Modelagens LES	8
2.2	Revisão de Modelagens Relativas ao Escoamento sobre o Degrau Descendente	11
2.3	Revisão de Metodologias Relativas à Fronteira Imersa	14
2.4	IMERSPEC	17
2.5	Revisão do Fenômeno de <i>aliasing</i>	20
2.6	Revisão de Paralelizações GPGPU	22
3	MODELAGEM MATEMÁTICA	26
3.1	Modelagem de Escoamentos Incompressíveis	26
3.2	Simulação de Grandes Estruturas (LES)	27
3.2.1	Filtragem Espacial	27
3.2.2	Equações de Balanço de Massa e de <i>Momentum</i> Linear Filtradas	29
3.2.3	Problema de Fechamento e Hipótese de Boussinesq	31
3.3	Modelos de Fechamento Funcionais Submalha	32
3.3.1	Modelo Smagorinsky Clássico	33
3.3.2	Modelo WALE	35
3.3.3	Modelo Dinâmico Germano-Lilly	37
4	METODOLOGIA IMERSPEC	41
4.1	Método da Fronteira Imersa	41
4.1.1	Múltipla Imposição com Domínios Ω e Γ Coincidentes	42
4.1.2	Múltipla Imposição com Domínios Ω e Γ Não Coincidentes	45
4.1.3	Imposição Espectral Direta com Domínios Ω e Γ Não Coincidentes	47
4.2	Método Pseudoespectral de Fourier	50
4.2.1	Metodologia Pseudoespectral	50
4.2.2	Procedimento para o MPF	51
4.2.3	Recuperação da Pressão	54
4.3	Aspectos Numérico-Computacionais	55
4.3.1	Paralelismos Implementados	55

4.3.2	Transformadas Rápidas de Fourier	58
4.3.3	Periodização de Escoamentos Não Periódicos	60
4.3.4	O problema de <i>aliasing</i>	62
4.3.5	Formas de Discretização do Termo Difusivo	65
4.3.6	Método de Avanço Temporal de Alta Ordem	66
4.3.7	Estrutura Geral IMERSPEC	67
5	RESULTADOS: VERIFICAÇÃO E VALIDAÇÃO	69
5.1	Paralelização MPI	69
5.1.1	Verificação MPF	69
5.1.2	Verificação MFI Coincidente	73
5.1.3	Verificação MFI Não Coincidente	76
5.1.4	Escoamento Laminar sobre um Degrau	80
5.2	Paralelização CPU/GPU	87
5.2.1	Verificação MPF	88
5.2.2	Verificação MFI Coincidente	94
5.2.3	Verificação MFI Não Coincidente	96
5.2.4	Verificação MIED/MFI	98
5.2.5	Escoamento Laminar sobre um Degrau	101
6	RESULTADOS: MODELAGEM DE ESCOAMENTOS TURBULENTOS	103
6.1	Trabalhos de Referência	103
6.2	Modelagem Turbulenta e a Metodologia IMERSPEC	105
6.3	Condições de Topo	110
6.4	Modelo Submalha de Smagorinsky Clássico com Van Driest	112
6.5	Modelo Submalha WALE	118
6.6	Modelo Submalha Dinâmico de Germano-Lilly	124
7	CONCLUSÕES	135
7.1	Futuros Trabalhos	138
	REFERÊNCIAS	140

1 INTRODUÇÃO

A turbulência em escoamentos de fluidos é um dos fenômenos físicos mais complexos e relevantes, presente em escoamentos atmosféricos, industriais e geofísicos. Em projetos de turbinas eólicas, a compreensão da turbulência no escoamento que o equipamento estará submetido e a correta modelagem da camada limite nas pás é essencial para a sua performance; em motores de combustão interna, uma turbulência expressiva promove uma melhor mistura entre o combustível e o ar, promovendo assim maior eficiência e reduzindo emissões de poluentes; em projetos de aviões, atrasando o desenvolvimento da camada limite turbulenta nas asas pode reduzir o consumo de combustível em voos.

Esses são apenas alguns de muitos exemplos que demonstram a importância da compreensão desse fenômeno na engenharia e justifica o interesse contínuo da comunidade científica ao longo das décadas. Entretanto, mesmo depois de mais de um século desde a primeira descoberta de escoamentos turbulentos por Osborne Reynolds³, a correta compreensão do fenômeno se mantém como um grande desafio da ciência.

Quando se analisa o processo de construção de conhecimento sobre o fenômeno da turbulência, nota-se que a compreensão dos escoamentos turbulentos foi por muito tempo ainda mais limitada do que hoje ou, por vezes, mal entendida⁴. Assim, como consequência, a capacidade de controlar escoamentos turbulentos, um tópico de grande interesse da engenharia mecânica, ainda não é factível.

Uma forma de se analisar esse fenômeno, para um avanço em seu entendimento, é através da estatística. No entanto, a coexistência de processos determinísticos e estocásticos na turbulência impossibilita uma abordagem puramente estatística para a sua modelagem⁴. Portanto, as formas de modelagem de escoamentos turbulentos se resumem necessariamente em três opções: puramente teórica, a material e a computacional. Através dessas formas, a disciplina de modelagem de turbulência avançou ao longo dos anos como uma combinação de intuição, teorias que tangenciam o fenômeno e empirismo, enquanto, no contexto de modelagens computacionais, limitados por motivos práticos relacionados a estabilidade numérica e eficiência computacional⁵.

Em um contexto teórico, tem-se que as clássicas equações de Navier-Stokes para fluidos newtonianos, servem para modelar mesmo escoamentos turbulentos complexos. Ao mesmo tempo, essas equações são não-lineares, e assim apenas escoamentos triviais são possíveis de solução contínua. Não se pode argumentar com a existência do termo não linear presente nas equações de Navier-Stokes visto que é esse exato termo que possibilita a modelagem de um fenômeno não linear com características tão complexas quanto a turbulência. Assim, restam a modelagem computacional ou material a representação de escoamentos turbulentos complexos e para a contribuição da compreensão de fenômeno da turbulência em escoamentos.

Na presente tese foca-se na modelagem computacional mas reconhece-se que modelagens através de experimentos materiais tiveram e tem significativa contribuição no campo de estudos da turbulência. Por vezes, ambas as modelagens se complementam para um entendimento mais profundo do fenômeno físico.

A modelagem computacional da turbulência é feita por simulações computacionais ou experimentos virtuais de escoamentos turbulentos e podem ser realizadas em diferentes níveis de fidelidade física e custo computacional. A simulação computacional direta (DNS) resolve todas as estruturas turbilhonares do escoamento, sem modelagem de fechamento, sendo restrita a baixos valores de número de Reynolds (Re) e à significativa capacidade computacional. Por outro lado, tem-se as abordagens médias das equações de Navier Stokes RANS/URANS, do inglês *Reynolds Averaged Navier Stokes Equations* ou *Unsteady RANS*, para modelar integralmente os efeitos da turbulência, com custo computacional menor, mas limitado ao comportamento médio do escoamento. As Simulações de Grandes Estruturas (LES), do inglês *Large Eddy Simulations*, representam o compromisso entre custo computacional e exatidão, para a solução das estruturas turbilhonares dominantes, mediante a modelagem das estruturas e interações submalha não resolvidas, tornando-se a abordagem de referência para escoamentos turbulentos complexos de interesse científico e de aplicações de engenharia.

Tanto as abordagens RANS/URANS e LES necessitam de modelos de fechamento da turbulência já que parte ou a totalidade das interações não lineares são modeladas. Esses modelos possuem uma história notável. Autores como J. V. Boussinesq, G. I. Taylor, Ludwig Prandtl, A. N. Kolmogorov, Theodore von Kármán, recorreram a analogias com a teoria cinética e o movimento browniano, bem como a conceitos de universalidade e comprimentos característicos, para estabelecer fundamentos base⁶. Ainda hoje, a modelagem do transporte turbulento ocorre a partir da noção de uma viscosidade turbulenta ou viscosidade submalha, em inglês *eddy viscosity*, sendo necessária à maioria dos modelos de fechamento de turbulência. Ainda sim, as bases desses modelos primordiais: viscosidade turbulenta, covariâncias de velocidades e tempos característicos de correlações também são as bases dos modelos de fechamento recentes. A modelagem de fechamento tem sido uma combinação de análise, empirismo e analogia, voltada para objetivos bastante práticos⁶.

Em relação às abordagens alternativas àquelas puramente teóricas, por vezes já foi sugerido que as abordagens RANS/URANS caíam em desuso e seriam substituídas inteiramente por LES como consequência do rápido poder computacional⁶. Certamente, a abordagem LES possibilita resultados computacionais mais exatos, entretanto, a previsão feita ainda não se tornou realidade. A dependência de maior custo computacional, horas humanas, malhas computacionais mais refinadas e/ou algoritmos mais exatos torna o extensivo uso dessa abordagem ainda inviável mesmo no contexto atual de programação em GPUs, do inglês *Graphical Processing Unit*.

Considerando a necessidade de algoritmos computacionais mais exatos para a abordagem LES que será explorada na presente tese, tem-se os métodos espectrais, os quais foram pioneiros

em modelagens dentro da área de dinâmica de fluidos computacional (CFD). Esses métodos atingem significativa acurácia com malhas grosseiras, apresentando assim uma grande vantagem sobre as demais metodologias em um contexto histórico onde os recursos computacionais eram severamente limitados. Kim; Leonard⁴ afirmam que a ordem de exatidão de métodos espectrais é usualmente classificada como infinita, *i.e.*, a exatidão desses métodos excede qualquer ordem algébrica (primeira, segunda, terceira, etc) desde que certas condições específicas sejam cumpridas.

Por exemplo, Orszag⁷ estimou que uma simulação computacional espectral com N^3 pontos colocantes apresenta a mesma exatidão de um experimento virtual que utiliza o método de diferenças finitas com $(4N)^3$ mas com o mesmo custo computacional de uma simulação que utiliza diferenças finitas com $(2N)^3$. Essa afirmação demonstra que métodos espectrais possuem maior exatidão com menor quantidade de pontos colocantes, entretanto, o seu custo computacional é, comparativamente, mais alto do que métodos de diferenças finitas.

Ainda hoje abordagens espectrais são utilizadas em algumas aplicações como modelagens relacionadas ao oceano e atmosfera, de alta resolução para camada limite atmosférica e em turbulência⁸, embora o seu custo computacional continue significativo em relação às técnicas numéricas similares.

Dentre as formulações espectrais disponíveis para a resolução computacional das equações de Navier-Stokes, como os métodos de Galerkin e de Chebyshev, o método de Fourier se destaca por sua eficiência computacional aliada à implementação via FFT, transformada rápida de Fourier do inglês *Fast Fourier Transform*, tornando-o uma alternativa no contexto da presente tese.

Assim, na modelagem apresentada neste trabalho utilizou-se o método de Fourier, em específico, o método pseudoespectral de Fourier (MPF). Com este método solucionam-se parcialmente as equações de balanço relacionadas ao escoamento modelado no domínio espectral enquanto produtos entre duas variáveis são resolvidos no domínio físico a fim de se evitar a convolução, operação computacional de alto custo.

Outro significativo desafio em formulações como a pseudoespectral de Fourier é a imposição de condições de contorno junto ao escoamento modelado. Por parte das operações serem realizadas no domínio espectral, condições de contorno também devem ser implementadas nesse domínio. Operações dentro do domínio de Fourier são globais e por isso a aplicação de condições de contorno não são diretas ou facilmente definidas.

Como alternativa a implementação dessas condições através do MPF, tem-se o Método da Fronteira Imersa (MFI) desenvolvido originalmente por Peskin⁹ para a modelagem numérico-computacional do escoamento em válvulas cardíacas. Nessa metodologia, modela-se as condições de contorno através de termos fontes inseridos nas equações de Navier-Stokes. A ideia básica por trás do método, por exemplo, é “forçar” o fluido em determinada região modelando seu

comportamento caso estivesse em contato com uma parede fixa.

A partir da junção do MFI com o MPF foi criada a metodologia IMERSPEC. Essa abordagem foi desenvolvida no laboratório de Mecânica dos Fluidos (MFLab) da Universidade Federal de Uberlândia (UFU) por Mariano¹⁰, Moreira¹¹, Villela¹², Kinoshita *et al.*¹³ e Nascimento¹⁴.

O IMERSPEC é uma metodologia espectral, que é caracterizada por exatidão expressiva mas também por significativo custo computacional. No contexto da presente tese, essa metodologia é empregada junto a abordagem LES que requisita malha computacional refinada e doravante o custo e poder computacional se tornam ainda mais relevantes.

Como a metodologia IMERSPEC aliada a abordagem LES apresenta significativo custo computacional, técnicas de paralelização que aumentam a performance da modelagem devem ser empregadas.

Historicamente, o aumento do poder computacional foi o principal fator na modelagem de turbulência. Afinal, Durbin⁶ afirmam que a modelagem computacional do fenômeno só se desenvolveu depois da conferência de 1968 sobre camadas limites turbulentas em Stanford.

Na última década, o aumento do poder computacional teve novamente papel protagonista ao introduzir técnicas de paralelismo heterogêneo entre GPU e CPU (Central Processing Unit), por meio da API CUDA, uma plataforma de computação paralela da NVIDIA que permite utilizar a GPU para processamento de propósito geral. Nessa abordagem desenvolvem-se códigos computacionais através de uma unidade de processamento gráfico de Propósito Geral, do inglês *General Purpose Graphics Processing Unit*, (GPGPU), os quais utilizam GPU/CPU para a modelagem de fenômenos físicos. No entanto, também deve-se considerar técnicas de paralelização consagradas pela comunidade científica como, por exemplo, a paralelização por MPI, do inglês *Message Passing Interface*.

As diferentes formas de paralelização devem ser avaliadas para a abordagem IMERSPEC, principalmente pela sua dependência pelo algoritmo de transformada rápida de Fourier (FFT), do inglês *Fast Fourier Transform*, o qual introduz dificuldade de paralelização devido a necessidade de operações de comunicação entre todos os pontos da malha computacional.

Assim, no presente trabalho utilizou-se a metodologia IMERSPEC, em um primeiro momento, explorando a biblioteca MPI e, em um segundo momento, a API CUDA em dois diferentes códigos desenvolvidos, respectivamente, IMERSPEC MPI e IMERSPEC CUDA.

1.1 Justificativa do trabalho

Embora significativas contribuições tenham sido feitas à abordagem IMERSPEC ao longo dos anos, a maioria destas foram restritas a modelagens bidimensionais e modelagens tridimensionais de escoamentos livres. Destaca-se que desde o primeiro trabalho sobre a metodologia em 2007, teve-se apenas três trabalhos sobre escoamentos tridimensionais em que estes

trabalhos são restritos a escoamentos cisalhantes livre. Sendo assim, a presente tese é o primeiro trabalho a modelar um escoamento tridimensional com a presença de paredes e ainda vai além ao modelar o mesmo escoamento em regime turbulento.

Destaca-se que em tentativas anteriores de modelagem de escoamentos tridimensionais com presença de parede pelo IMERSPEC, observou-se o surgimento de oscilações numéricas indesejáveis na direção perpendicular à parede. Esse problema foi corrigido na presente tese, e o escoamento sobre o degrau com expansão brusca, em qualquer regime, foi simulado usando a metodologia IMERSPEC.

A escolha desse escoamento de referência se deve à sua relevância tanto teórica quanto prática em engenharia. Trata-se de um escoamento complexo, caracterizado pela separação da camada limite, pela evolução de estruturas turbulentas e pelo recolamento do escoamento à parede à jusante do degrau. Exemplos práticos desse tipo de escoamento são encontrados em aerofólios com grandes ângulos de ataque, escoamentos em torno de defletores e de veículos, em entradas de dutos de motores, em condensadores, em válvulas de retenção e em escoamentos ao redor de edificações¹⁵. Além disso, o escoamento sobre o degrau é um problema clássico de referência na literatura, com diversas contribuições computacionais e experimentais disponíveis para comparação^{16–22}.

Outra novidade, que é apresentada no presente trabalho, é a introdução de novas técnicas de paralelização junto ao IMERSPEC, tornando-o mais amigável a modelagens complexas. Salienta-se que a paralelização GPGPU é uma novidade junto ao IMERSPEC enquanto a paralelização em MPI que ocorre em duas das três direções do escoamento é um avanço em relação à paralelização introduzida por Moreira²³ que paralelizava apenas uma única direção do escoamento. Como foram testadas técnicas de paralelização distintas, escreveu-se dois códigos computacionais diferentes: IMERSPEC MPI e IMERSPEC CUDA.

Adicionalmente, ao longo do desenvolvimento, duas abordagens foram investigadas além dos objetivos inicialmente propostos. A primeira é a discretização antissimétrica do termo difusivo, proposta como alternativa para suprimir as oscilações numéricas indesejáveis identificadas na modelagem de escoamentos com parede e minimizar os efeitos de *aliasing* inerentes a métodos espectrais. A segunda é o Método de Imposição Espectral Direta (MIED), desenvolvido como uma tentativa de tornar integralmente espectral o processo de transferência de informação entre os domínios euleriano e lagrangiano do IMERSPEC. Embora ambas tenham sido descartadas ao longo da pesquisa, sua investigação gerou contribuições metodológicas relevantes, discutidas nos capítulos 5 e 6.

1.2 Objetivos

Com base na justificativa da tese, tem-se como objetivo geral o desenvolvimento de uma ferramenta computacional com metodologia espectral de Fourier com alta ordem e capaz de

uma modelagem de escoamentos turbulentos com a presença de paredes segundo a abordagem LES com paralelização eficiente. O objetivo geral, portanto, é necessário para responder o questionamento: o IMERSPEC consegue modelar escoamentos turbulentos, tridimensionais com a presença de paredes ou sua aplicação deve ser restrita a modelagens de escoamentos cisalhantes livres?

Para cumprir com o objetivos geral, os seguintes objetivos específicos são definidos:

- Tridimensionalizar a metodologia IMERSPEC com diferentes formas de paralelização;
- Verificar os códigos computacionais desenvolvidos, IMERSPEC MPI e IMERSPEC CUDA, para os diferentes tipos de paralelização explorados;
- Validar os códigos computacionais através da modelagem do escoamento laminar clássico sobre o degrau;
- Avaliar o ganho da paralelização GPGPU frente à paralelização MPI;
- Adaptar a metodologia IMERSPEC para a abordagem LES no escoamento turbulento sobre degrau;
- Analisar a performance dos modelos de fechamento submalha: Smagorinsky clássico, WALE e Dinâmico de Germano-Lilly em relação ao escoamento modelado.

1.3 Escopo do Trabalho

A fim de atingir todos os objetivos da presente tese e a partir desta introdução, o trabalho está organizado de acordo com os seguintes capítulos:

- O capítulo 2: apresenta-se uma revisão bibliográfica sobre os principais tópicos abordados na tese, indicando onde se insere a presente pesquisa no contexto mundial;
- Capítulo 3: aborda-se a formulação matemática envolvida na modelagem de escoamentos laminares ou turbulentos focando na abordagem LES e os modelos funcionais de fechamento submalha implementados;
- Capítulo 4: explora-se a metodologia IMERSPEC, desde a sua formulação matemática em que se discute os MPF e MFI até os aspectos numérico computacionais envolvidos na implementação computacional dos códigos desenvolvidos;
- Capítulo 5: os processos de verificação e validação dos códigos IMERSPEC MPI e IMERSPEC CUDA são apresentados e discutidos. No final do capítulo, constata-se qual o melhor código para a modelagem de fenômenos turbulentos tridimensionais;

- Capítulo 6: mudanças na metodologia IMERSPEC para a correta reprodutibilidade de fenômenos físicos turbulentos são apresentados junto aos trabalhos de experimento material e numérico de referência, utilizados ao longo do capítulo. Em seguida, é feita a justificativa e análise de escolhas de modelagem do caso de referência junto ao IMERSPEC. Por fim, a discussão de cada um dos modelos de fechamento de turbulência submalha implementados é apresentada. Destaca-se que ao final do capítulo, para a modelagem do escoamento turbulento sobre o degrau, também é constatado um modelo submalha que apresentou melhor custo benefício dentre os resultados e análises realizados.
- Capítulo 7: apresenta-se às conclusões da tese, assim como perspectivas de futuros trabalhos relacionados aos tópicos abordados na presente pesquisa.

2 REVISÃO BIBLIOGRÁFICA

No presente capítulo busca-se contextualizar o estado da arte em seis diferentes tópicos: modelos de fechamento da modelagem LES, o escoamento sobre um degrau descendente, metodologias relacionadas ao MFI, metodologia IMERSPEC explorada no presente trabalho, fenômeno de *aliasing* em modelagens espectrais de Fourier e a paralelização GPGPU.

2.1 Revisão de Modelagens LES

A Simulação de Grandes Estruturas (LES) é uma abordagem consagrada para a modelagem de escoamentos turbulentos complexos por representar um equilíbrio ótimo entre custo computacional e fidelidade à física. Diferentemente da Simulação Computacional Direta (DNS), que resolve todas as estruturas turbilhonares, e das abordagens RANS/URANS, que modelam integralmente todas as interações não lineares, o LES baseia-se na aplicação de um filtro espacial nas equações de balanço da fluidodinâmica, conceito formalizado originalmente por Léonard²⁴. Sob essa ótica, as estruturas turbilhonares maiores e energeticamente dominantes são resolvidas diretamente, surgindo a necessidade de modelar o efeito das estruturas não resolvidas por meio do tensor de tensões submalha, τ_{ij}^{SGS} .

Embora as primeiras decomposições do termo não linear filtrado tenham sido propostas por Léonard²⁴, a definição moderna do tensor submalha, adotada na presente tese, como $\tau_{ij}^{SGS} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j$ foi estabelecida por Germano²⁵. Essa redefinição foi fundamental para garantir a invariância Galileana do tensor submalha, superando as limitações das decomposições anteriores, cujos componentes individuais dependiam do sistema de referência adotado. Outro destaque é também a maior facilidade de modelagem de um tensor, já que antes os efeitos do tensor de Leonard e o cruzado também deveriam ser modelados. Salienta-se ainda que a identidade proposta de Germano²⁵ engloba todos os tensores previamente modelados separadamente, sem prejuízo da modelagem turbulenta²⁶.

Portanto, a escolha do modelo de fechamento para o tensor submalha é um dos principais fatores determinantes para a correta representividade do fenômeno físico turbulento, especialmente em escoamentos caracterizados por fortes gradientes de pressão e descolamento de camada limite.

Historicamente, o modelo de fechamento de turbulência submalha mais clássico, o primeiro proposto, fundamenta-se na hipótese de viscosidade submalha definida a partir da hipótese de Boussinesq-Reynolds, foi o modelo de Smagorinsky²⁷. Este modelo foi criado para a modelagem de escoamentos meteorológicos e determina o tensor submalha, τ_{ij}^{SGS} a partir do tensor de taxa de deformação resolvido junto ao comprimento característico da malha

computacional Δ e uma constante adimensional *ad-hoc* C_s . Apesar de sua simplicidade e baixo custo computacional, o modelo clássico falha ao superestimar a difusão viscosa de informações em regiões de cisalhamento laminar e próximo a paredes sólidas²⁸. Para mitigar essa deficiência, abordagens tradicionais introduzem funções de amortecimento fenomenológicas, destacando-se a formulação empírica de Van Driest²⁹, que força o decaimento correto da viscosidade turbulenta, por amortecimento, à medida que a distância normalizada à parede, y^+ tende a zero.

Como evolução natural para contornar o uso de ajustes empíricos fixos, surgiram modelos submalha que já modelam corretamente o comportamento do escoamento próximo à parede. O modelo *Wall-Adapting Local Eddy-viscosity* (WALE), proposto por Nicoud; Ducros³⁰, redefine a viscosidade submalha com base no quadrado do tensor de gradiente de velocidade. Essa formulação garante intrinsecamente o comportamento assintótico correto perto de paredes sólidas (y^{+3}) e anula naturalmente em escoamentos laminares, isto é, a viscosidade submalha só é ativa em estruturas turbilhonares tridimensionais.

Alternativamente, outra proposta para melhorar o modelo de Smagorinsky, foi a determinação local e dinâmica do coeficiente estático *ad-hoc*, o C_s . Assim, este coeficiente tornou-se variável com o modelo dinâmico de Germano-Lilly^{1,2}. Baseado em uma identidade algébrica entre dois níveis de filtragem distintos de mesma natureza (filtro base e filtro-teste), o coeficiente do modelo, $C_d = C_s^2$, é computado dinamicamente à medida que a simulação progride, zerando em regimes laminares e junto a paredes. Para sanar instabilidades numéricas e singularidades matemáticas decorrentes de valores locais negativos, Lilly² introduziu uma formulação baseada no método dos mínimos quadrados, consolidando o amplamente conhecido modelo submalha de Germano-Lilly^{1,2}.

A fim de estabelecer o estado da arte em modelos submalha, a literatura tem formulações que revolucionam, estendem ou refinam os modelos de fechamento apresentados. No âmbito dos modelos dinâmicos locais, Meneveau; Lund; Cabot³¹ propuseram o modelo dinâmico lagrangiano, que realiza a média dos coeficientes submalha ao longo de *pathlines*, conferindo estabilidade numérica em geometrias complexas e totalmente heterogêneas onde médias espaciais homogêneas são inaplicáveis. Os autores ainda determinaram que o modelo desenvolvido apresenta o mesmo resultado computacional em escoamentos homogêneos que o modelo dinâmico de Germano-Lilly mas em, por exemplo escoamentos ao longo de um canal, a nova formulação apresenta resultados ligeiramente superiores.

Em relação à viscosidade submalha baseada em invariantes algébricos modernos, destacam-se o modelo de Vreman³² e o modelo σ de Nicoud *et al.*³³, este último utilizando valores singulares do gradiente de velocidade para aprimorar o comportamento em escoamentos tridimensionais complexos. O modelo submalha de Vreman foi idealizado para aplicações de engenharia por ser de relativa simplicidade matemática e de implementação. Destaca-se que, em seu artigo de apresentação, o modelo apresentou exatidão tão boa quanto o modelo dinâmico de Germano-Lilly^{1,2}.

Em contrapartida, o modelo de fechamento σ , desenvolvido por parte dos mesmos autores que propuseram o modelo WALE, pode ser considerado uma evolução direta capaz de superar as falhas de consistência física intrínsecas às formulações anteriores em escoamentos complexos. A deficiência clássica do modelo de Smagorinsky reside na sua incapacidade de se anular em escoamentos de cisalhamento puro, além de não decair naturalmente na proximidade de paredes sólidas sem o uso de funções de amortecimento. Para mitigar as limitações de parede, o modelo WALE introduziu com sucesso o comportamento assintótico cúbico correto, $O(y^{+3})$, e zerou no cisalhamento, mas manifestou uma falha conceitual oposta: ele produz viscosidade submalha espúria em regiões de rotação sólida pura, além de não se anular em movimentos de contração ou expansão axissimétrica e isotrópica.

O modelo σ elimina esse impasse cruzado ao fundamentar sua formulação nos valores singulares do tensor de gradiente de velocidade resolvido. Por meio do menor valor singular (σ_3), que atua como um marcador matemático exato para escoamentos locais bidimensionais, o modelo σ zera a viscosidade submalha simultaneamente em cenários de cisalhamento e de rotação rígida, estendendo esse desligamento ideal também para expansões e contrações puras. Dessa forma, ao erradicar as falhas de difusão artificial que comprometem tanto o modelo de Smagorinsky quanto o WALE, o modelo σ estabelece uma consistência matemática e física superior para simulações tridimensionais altamente heterogêneas.

Adicionalmente, visando romper com as limitações de alinhamento impostas pela hipótese clássica de Boussinesq-Reynolds, os modelos de similaridade entre grandezas características, documentados por Liu; Meneveau; Katz³⁴, estimam as tensões submalha diretamente a partir das menores estruturas resolvidas da malha através do tensor de tensões resolvidas $L_{ij} = \widetilde{\bar{u}_i \bar{u}_j} - \widetilde{\bar{u}_i} \widetilde{\bar{u}_j}$. Testes *a priori* experimentais em escoamentos com alto número de Reynolds, $Re_\lambda = 430$ em que este adimensional é definido a partir do microcomprimento de Taylor (λ), revelam que essa abordagem atinge coeficientes de correlação substancialmente superiores em relação a modelos submalha que utilizam viscosidade cinemática submalha, ν_{SGS} , por capturar uma coerência relativa que possui tanto uma natureza cinemática espectral quanto uma assinatura dinâmica decorrente da forte intermitência espacial da turbulência.

Essa formulação permite representar localmente o fenômeno de *backscatter*, o retorno de energia das pequenas para as grandes estruturas turbilhonares, efeito este cronicamente ignorado pelos modelos discutidos até aqui por considerações da criação dos próprios modelos ou por estabilidade numérica-computacional. Todavia, os autores pontuam que essa assinatura física exige estritamente o emprego de filtros bem localizados no espaço físico (como os filtros *top-hat* ou Gaussiano). Essa escolha deve ser feita em virtude dos filtros de corte espectral que destroem a correlação das estruturas locais devido à oscilações espúrias. A natureza pouco dissipativa do modelo idealizado requer o uso associado de modelos de fechamentos mistos ou de funções de controle de transferência energética para mitigar severas instabilidades numéricas. Destaca-se que essa recomendação do filtro dos Liu; Meneveau; Katz³⁴ também foi percebida no presente

trabalho na implementação do modelo dinâmico de Germano-Lilly.

Paralelamente, a busca por consistência matemática estrita levou ao desenvolvimento de algoritmos modernos de avaliação de consistência física para modelos submalha, como o proposto por Silvis; Remmerswaal; Verstappen³⁵, que mapeia às simetrias das equações de Navier-Stokes e às propriedades da difusão viscosa. Nessa linha de controle rigoroso da energia, situam-se as discussões de Verstappen³⁶ sobre a quantidade mínima de difusão viscosa necessária para amortecer as estruturas submalha sem degradar a física resolvida, princípio que norteou a concepção dos modelos de difusão mínima, como o modelo QR, proposto por Verstappen³⁶ e o modelo AMD, do inglês *Anisotropic Minimum-Dissipation*, proposto por Rozema *et al.*³⁷. O modelo AMD generaliza as propriedades ideais de desligamento em regimes laminares para malhas fortemente anisotrópicas sem necessitar de aproximações grosseiras para a largura do filtro da malha base (Δ), *i.e.*, o comprimento característico da malha base. O modelo original QR tinha sua exatidão comprometida em malhas anisotrópicas visto que esta é diretamente relacionada ao incremento mais grosseiro.

Embora abordagens focadas em consistência física estrita³⁵ ou difusão mínima anisotrópica³⁷ representem avanços expressivos na modelagem LES, os modelos de fechamento de turbulência utilizados no presente trabalho foram restritos a modelos clássicos simples.

Na presente tese, se implementou o modelo clássico Smagorinsky com amortecimento das funções Van Driest²⁹, o modelo WALE³⁰ e o modelo dinâmico de Germano-Lilly^{1,2}. Conforme explicitado no capítulo 1, a presente tese é o trabalho pioneiro em modelagem de escoamento turbulentos com a presença de paredes usando a metodologia IMERSPEC. Sendo assim, optou-se por métodos históricos de grande significância e popularidade ainda hoje.

2.2 Revisão de Modelagens Relativas ao Escoamento sobre o Degrau Descendente

Além das escolhas dos modelos de fechamento de turbulência, foi escolhido um escoamento clássico que apresentasse a interação do fluido com paredes como o escoamento sobre o degrau, do inglês *backward-facing step*, um *benchmark* clássico. Esse escoamento oferece um cenário desafiador para os modelos submalha e a metodologia IMERSPEC já que apresenta fenômenos severos de separação e recolamento da camada limite. Este escoamento específico é caracterizado pela separação abrupta na aresta do degrau, geração de uma camada de cisalhamento com evolução de estruturas turbilhonares e o subsequente recolamento do escoamento à parede, a jusante do degrau.

Jovic; Driver¹⁶ modelaram esse escoamento através de experimentos materiais para $Re_H = 5000$ em que H é a altura do degrau e uma taxa de expansão igual a 1,2. Esses experimentos materiais foram realizados para validar uma simulação numérico-computacional

DNS de $Re_H = 5100$. Salienta-se que na época da realização do experimento material, a limitação do poder computacional dos *hardwares* disponíveis foi o fator limitante na de definição do número de Reynolds dos experimentos virtuais e materiais.

A partir de velocimetria a *laser Doppler* de dois feixes com dois componentes de fibra óptica, do inglês *Laser Doppler Velocimetry* (LDV), mediu-se os componentes médios da velocidade do escoamento e três componentes do tensor médio de Boussinesq-Reynolds ($\overline{u'^2}$, $\overline{v'^2}$ e $-\overline{u'v'}$) enquanto interferômetro a laser de escoamento de óleo foi utilizado para determinar a distribuição média do coeficiente de atrito e pressão. Um ponto de recolamento igual a $6,0H \pm 0,15H$ para esse Re_H .

Outro trabalho que também será usado para comparação com os resultados referentes a modelagem de escoamentos turbulentos na presente tese é o trabalho numérico-computacional desenvolvido por Le; Moin; Kim¹⁷ em que os autores afirmam ser DNS. Os resultados obtidos por simulação numérico-computacional reduziram a região de modelagem do escoamento sobre o degrau através do uso de condição de simetria no topo do domínio computacional. Os experimentos virtuais foram executados a partir de $Re_H = 5100$ e com a mesma taxa de expansão do trabalho de Jovic; Driver¹⁶ e destaca-se que não há concordância completa entre os perfis médios de $\bar{v}$ entre os autores. Maior detalhamento das condições dos experimentos virtuais e materiais serão abordados no capítulo 6.

Por outro lado, Lee; Mateescu¹⁸ conduziram experimentos virtuais e materiais para o mesmo benchmark clássico mas restrito a modelagens/medições bidimensionais para $Re_h \leq 3000$ para razões de expansões iguais a 1,7 e 2,0. Como a turbulência é um fenômeno tridimensional, considerou-se apenas os resultados desses autores para modelagens laminares no presente trabalho (Capítulo 5). Salienta-se que as medições materiais de Lee; Mateescu¹⁸ foram realizadas a partir de sensores de fio quente que representa um processo de medição não intrusiva.

Em relação a modelagem numérico-computacional do *benchmark*, Aider; Danet¹⁹ demonstram a importância da correta imposição das condições de entrada nessa classe de escoamento. Os autores testam duas diferentes condições: velocidade média turbulenta da camada limite perturbada por ruído branco e um perfil de entrada determinado por uma simulação precursora através da modelagem de uma camada limite quase-temporal segundo uma modelagem LES com o modelo de fechamento submalha do tipo função estrutura de velocidade de ordem dois. Os melhores resultados acerca da modelagem realizada foram aqueles onde se implementou a condição de entrada a partir do cálculo de uma simulação precursora. Para este caso, notou-se que as estruturas turbulentas provenientes do escoamento a montante influenciam significativamente o desenvolvimento do escoamento a jusante da expansão do degrau. Em particular, os streaks de velocidade e os vórtices quase-longitudinais desencadeiam uma rápida desestabilização da camada de cisalhamento, resultando em um encurtamento do comprimento de recolamento do escoamento. Nota-se, entretanto, que o *inlet* a partir de uma simulação precursora foi comparado com um perfil perturbado por ruído branco que não corresponde a uma perturbação física. Adici-

onalmente, destaca-se que o uso de uma simulação precursora para a condição de entrada na modelagem de interesse exige maiores recursos computacionais.

Considerando ainda o efeito da condição de entrada do escoamento no domínio computacional em modelagens do escoamento sobre o degrau, Jürgens; Kaltenbach²¹ analisam o ponto de recolamento para um escoamento em transição a partir da indução de ondas de instabilidade formadas através da sopragem ou sugamento periódico do fluido por uma fenda na borda do degrau. As simulações numérico-computacionais foram realizadas seguindo uma abordagem LES junto ao modelo de fechamento submalha dinâmico de Germano-Lilly^{1,2}. Destaca-se que os autores através do forçamento da onda, obtiveram uma redução relativa no valor do ponto de recolamento de até 29% para ângulos de enviesamento de até 40°, com uma intensidade de ruído de 1% da velocidade do escoamento livre. De forma geral, a partir dos resultados, observa-se que um aumento do ângulo de enviesamento provoca um encurtamento do valor do ponto de recolamento do escoamento modelado.

Ainda destacando o impacto das condições de entrada no escoamento de interesse para uma modelagem espectral, destaca-se o trabalho de Kanchi; Sengupta; Mashayek²². Os autores realizaram novamente um estudo sobre o impacto das condições *inlet* na modelagem do fenômeno físico, entretanto o diferencial do trabalho se deve ao fato de que as simulações foram realizadas através de uma metodologia espectral, em específico, através do método de elementos espectrais de Legendre, e utilizando uma abordagem LES implícita (um filtro passa-baixa é aplicado diretamente à solução numérico-computacional, sem a implementação de modelos de fechamento ou o uso da hipótese de Boussinesq). A turbulência gerada para a condição de entrada é determinada por processos estocásticos. Destaca-se aqui que a abordagem espectral utilizada pelos autores apresentou resultados significativos em comparação com as respectivas referências utilizadas, sem a necessidade de simulações precursoras conforme utilizadas por Aider; Danet¹⁹. Essa estratégia também foi adotada no presente trabalho ao utilizar perfis médios de velocidade perturbados pelo ruído de Smirnov; Shi; Celik³⁸, que modela um comportamento físico, ao contrário do ruído branco²³.

Nota-se a partir dos trabalhos de Aider; Danet¹⁹, Jürgens; Kaltenbach²¹ e Kanchi; Sengupta; Mashayek²² o significativo impacto das condições de entrada do domínio computacional sobre a modelagem do escoamento sobre um degrau descendente, em especial sobre o parâmetro do ponto de recolamento. Algo a ser considerado ao analisar os resultados do capítulo 6 na presente tese. Para mais detalhes acerca dos mecanismos de escoamento, transferência de calor e estratégias de controle relacionados ao *benchmark* clássico do escoamento sobre o degrau descendente, pode-se consultar a revisão bibliográfica realizada por Chen *et al.*¹⁵.

2.3 Revisão de Metodologias Relativas à Fronteira Imersa

Abordada a metodologia utilizada na modelagem de escoamentos turbulentos no presente trabalho e o *benchmark* clássico escolhido, é discutida a metodologia usada para a imposição de condições de contorno nos escoamentos simulados na presente seção. O método da Fronteira Imersa, MFI, foi desenvolvido por Peskin⁹ a fim de modelar válvulas cardíacas. A essência dessa metodologia é a utilização de termos fontes nas equações de Navier-Stokes para a modelagem de superfícies sólidas no escoamento. Em um trabalho posterior, Peskin³⁹ formalizou a metodologia matemática do MFI definindo muitas características do método ainda usadas. Por exemplo, definiu-se: uma descrição euleriana para a modelagem do fluido e uma lagrangiana para a modelagem da geometria sólida imersa; a dedução do MFI realizada a partir do princípio da menor ação; e requisitos mínimos para funções de interpolação/distribuição que são aproximações numéricas da função delta de Dirac.

Considerando a aproximação numérica do delta de Dirac, o trabalho de Tornberg; Engquist⁴⁰ realizou um estudo da exatidão em suavizações do delta de Dirac com suporte compacto, analisando três suavizações com suporte de 1 ou 2 pontos. A melhor função de suavização dentre as analisadas pelos pesquisadores foi a função cúbica, utilizada no IMERSPEC, mesmo que ela não siga completamente as sugestões de uma boa função aproximadora determinadas previamente por Peskin³⁹. Griffith; Peskin⁴¹, ao implementar essa função, identifica um custo computacional menor do que alternativas como a função delta com 6 pontos para a mesma exatidão numérica, fato também atestado por Mariano⁴², mesmo que também notando oscilações numéricas para modelagens com uma interface verdadeira.

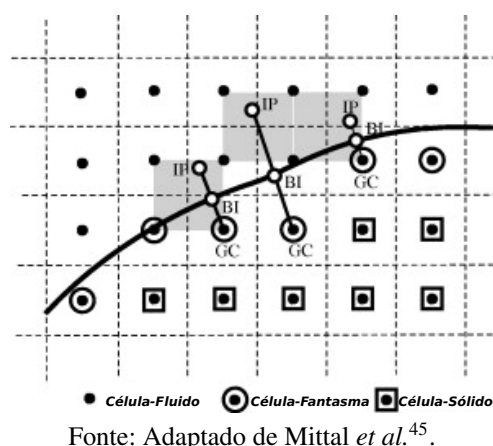
Uhlmann⁴³ propôs uma nova metodologia dentro do escopo do MFI. A nova metodologia, denominada método de imposição direta, DFM, incorporou as aproximações do delta de Dirac em uma abordagem direta entre os domínios eulerianos e lagrangianos. Essa metodologia foi implementada junto a um método espacial de diferenças finitas e permitiu avanços temporais menos restritivos. Uhlmann⁴³ ainda define uma proporção de, no máximo, 1,2 entre as malhas lagrangianas e eulerianas para uma integração estável entre os domínios.

A fim de aumentar a exatidão do DFM, Wang; Fan; Luo⁴⁴ desenvolveram o MDFM, que torna o DFM um processo iterativo. A principal motivação dos autores no aperfeiçoamento desse esquema do MFI foi a modelagem de um escoamento com milhares de partículas de tamanho finito e se movendo, usando soluções computacionais em grandezas reais. Para isso, tornou-se o método proposto por Uhlmann⁴³ em processo iterativo e, assim, conseguiu-se reproduzir as condições de não deslizamento na superfície das inúmeras partículas presentes no escoamento de forma acurada.

Até então, as metodologias referenciadas nesta seção implementam a geometria imersa do MFI seguindo uma proposta de interface difusa, *i.e.*, a geometria lagrangiana é implementada através do termo fonte das equações de balanço da quantidade de movimento linear do

escoamento modelado. Há, entretanto, outra classe de esquemas de implementação do MFI denominados métodos de interface nítida, em que a fronteira imersa é implementada diretamente sem termos fontes. Mittal *et al.*⁴⁵ desenvolveram um esquema de MFI no qual as velocidades da fronteira imersa são implementadas a partir de células fantasmas, do inglês *ghost-cell* (GC), diretamente dispostas na vizinhança do ponto de interesse. Na Figura 1 é apresentado um esquema de implementação onde IP representa um ponto imagem, do inglês *image-point*, e BI representa a intersecção de fronteira, do inglês *boundary intercept*.

Figura 1 – Esquema de implementação de MFI com interface nítida.



Fonte: Adaptado de Mittal *et al.*⁴⁵.

Destaca-se que esse método de interface nítida modela o fluido com a presença do sólido imerso apenas em relação ao fluido externo à superfície. Caso esse tipo de esquema fosse implementado junto a uma metodologia espectral de Fourier, em toda a superfície sólida imersa aconteceria um salto de propriedades, *i.e.*, uma descontinuidade. Por conta das diversas descontinuidades, o escoamento apresentaria o fenômeno de Gibbs e qualquer modelagem do fenômeno físico seria comprometida. Assim, especula-se que esse tipo de esquema não é viável para ser empregado junto ao IMERSPEC, admitindo-se que essa hipótese deve ser testada em futuros trabalhos. Ainda assim, esse tipo de esquema tem sido explorado dentro do MFLab no código MFSim⁴⁶.

O escoamento laminar e turbulento sobre colinas periódicas também foi modelado por Chang *et al.*⁴⁷ em uma abordagem híbrida onde se utilizou parcialmente transformadas de Fourier na resolução computacional em duas das três direções do escoamento. O avanço temporal é feito de forma híbrida utilizando-se o método de Adams-Bashforth para os termos advectivos e o de Crank-Nicolson para os difusivos. As paredes do escoamento sobre colinas periódicas confinadas foram modeladas a partir do MFI com um esquema difuso em que são realizados processos de interpolação e distribuição para a implementação das condições de contorno de forma análoga ao MDFM explorado no presente trabalho. Destaca-se que uma avaliação do impacto da modelagem de funções de parede em simulações de escoamentos turbulentos foi realizada e constatou-se que elas são necessárias.

Nota-se ao longo do tempo um esforço da comunidade científica em aumentar a exatidão

do MFI. Em interfaces difusas, o maior ponto a ser aperfeiçoado são as funções de interpolação/distribuição. Com base nesse aspecto, Bao; Kaye; Peskin⁴⁸ desenvolveram uma nova função de interpolação/distribuição que aperfeiçoou a invariância translacional quando comparada a outras funções clássicas usadas no MFI. Em um esforço de redução do erro referente a manutenção do posicionamento lagrangiano em modelagens de escoamentos com grande diferença de pressão entre superfícies sólidas finas pelo MFI, Bao *et al.*⁴⁹ desenvolveram esquemas de implementação do MFI livre de divergência em sua interpolação e distribuição, em que a velocidade interpolada do método seja, necessariamente, $\mathbb{C}^1$. Destaca-se a utilização de funções *B-Spline* como funções de interpolação/distribuição. Os novos esquemas do MFI resultaram em forças lagrangianas mais suaves e reduziram o erro da manutenção do posicionamento lagrangiano inerente ao MFI, entretanto, tais resultados foram alcançados com um custo computacional maior e são restritos a escoamentos periódicos.

Chen; Peskin⁵⁰ introduziram um novo tipo de fronteira imersa espectral, denominado fronteira imersa espectral Fourier, através do uso da NUFFT. Os autores propuseram essa nova metodologia substituindo processos de interpolação e distribuição entre os domínios lagrangianos e eulerianos por transformadas da NUFFT, enquanto a modelagem da interação entre os escoamentos e as estruturas é realizada considerando uma rigidez e deslocamento em cada ponto lagrangiano. A fronteira imersa espectral é considerada uma metodologia sem malha por Chen; Peskin⁵⁰, entretanto, a nova metodologia continua dependendo dos números de onda em seus cálculos, *i.e.*, ainda há dependência de uma malha.

Pelo uso da NUFFT no algoritmo são preservadas propriedades básicas ao IBM como a manutenção do posicionamento lagrangiano, invariância translacional exata, processos de interpolação/distribuição exatas. Porém, o método ainda apresenta convergência de segunda ordem e os testes foram realizados em um algoritmo pseudoespectral de Fourier. Nota-se que, em relação à metodologia pseudoespectral de Fourier e aos seus possíveis problemas, seção 2.5, que o termo advectivo foi discretizado através de sua forma não divergente e resolvido de forma pseudoespectral, sendo assim, os autores foram forçados a empregar um processo de *dealiasing*, a regra clássica de 2/3, diminuindo a principal vantagem de qualquer formulação espectral: sua exatidão.

Gruninger; Griffith⁵¹, em um esforço para atingir uma maior manutenção do posicionamento lagrangiano no MFI, desenvolveram novas funções de interpolação/distribuição derivadas a partir da composição de *B-Splines*. Conforme também visto no trabalho de Bao *et al.*⁴⁹, funções da família *B-Splines* possibilitam interpolação de velocidade localmente livre de divergência e operadores de distribuição de força lagrangiana efetivos. Funções de interpolação/distribuição localmente livres de divergência oferecem uma manutenção do posicionamento lagrangiano melhor em relação à clássica implementação do MFI, principalmente em modelagens de escoamentos pressurizados com membranas fechadas.

O trabalho de Gruninger; Griffith⁵¹ é diretamente relacionado ao de Bao *et al.*⁴⁹, apresen-

tando o mesmo comportamento para modelagens dinâmicas de membranas excitadas desde que se utilize funções de interpolação de classe C^1 ou acima. Entretanto, as novas funções propostas oferecem alta exatidão sem o custo computacional dos esquemas previamente desenvolvidos, sendo preferíveis em simulações com significativas dimensões tridimensionais.

Nota-se que, apesar do esforço de uma maior exatidão dos recentes trabalhos de MFI, nenhum desses casos atingiu uma exatidão igual ou maior do que a metodologia IMERSPEC explorada no presente trabalho para os casos de verificação onde foram utilizados processos de interpolação e distribuição, ainda que se admita que, até então, não foram realizadas modelagens de casos extremos como um escoamento altamente pressurizado próximo a uma membrana delgada excitada⁵¹.

2.4 IMERSPEC

A partir do MFI tornou-se possível o acoplamento dos métodos pseudoespectral de Fourier e fronteira imersa, denominado IMERSPEC. Essa metodologia foi desenvolvida em 2007 dentro do MFLab da Universidade Federal de Uberlândia. Mariano¹⁰ desenvolveu o primeiro trabalho que utilizou essa ferramenta computacional para a modelagem do escoamento em uma cavidade bidirecional com tampa deslizante a baixos números de Reynolds. Neste trabalho foram apresentadas as vantagens do IMERSPEC, como a acurácia significativa para modelar detalhes do escoamento com $Re = 400$ até então não observados em outras publicações à época. A fronteira imersa, neste trabalho, foi implementada através do MFV, com o qual apresentaram-se resultados classificados como satisfatórios, entretanto, esse método apenas era considerado para malhas lagrangianas e eulerianas coincidentes.

A metodologia IMERSPEC foi expandida para a modelagem de escoamentos tridimensionais através do trabalho de Moreira¹¹, o qual modelou jatos turbulentos livres circulares em desenvolvimento temporal utilizando LES junto ao modelo submalha de Smagorinsky. Os resultados obtidos foram consistentes com análises físicas em que se verificou a transição do escoamento à turbulência e as respectivas estruturas turbilhonares típicas relacionadas ao fenômeno modelado. Em especial, destaca-se que na modelagem de escoamentos em desenvolvimento temporal, as condições de periodicidade inerentes ao uso da FFT são obrigatórias.

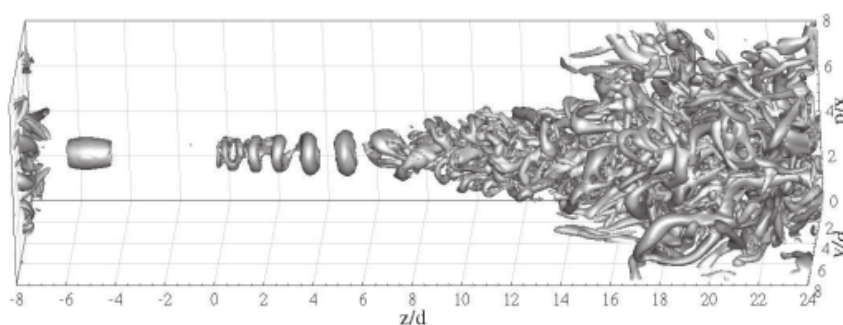
Mariano⁴² aperfeiçoou o IMERSPEC substituindo o MFV pelo MDFM, onde os processos de validação e verificação indicaram quarta ordem de convergência. Esse foi o primeiro trabalho onde modelagens de escoamentos em desenvolvimento espacial foram implementadas com o IMERSPEC através do uso do método de *Buffer* junto a uma região de forçagem do MDFM, responsáveis pela periodização na direção de desenvolvimento do escoamento e condição de entrada no domínio útil. Mariano⁴² modelou escoamentos clássicos bidimensional e tridimensional sobre degrau, além de escoamentos sobre cilindros e na queda de corpos rígidos. Seus resultados mostraram-se de acordo com a literatura e parte deles foram publicados

posteriormente⁵².

Escoamentos bifásicos bidimensionais foram modelados no trabalho de Villela¹², que adaptou o IMERSPEC para a representação de interfaces móveis e deformáveis típicas do escoamento de interesse através do método híbrido *Front-Tracking/Front-Capturing*. O escoamento modelado foi a ascensão de uma bolha em um escoamento de formato cilíndrico, onde se notou o fenômeno de Gibbs para grandes saltos de propriedades físicas entre as suas fases. As simulações mais sofisticadas envolveram a remalhagem de pontos lagrangianos durante a ascensão da bolha enquanto esta se deforma.

Moreira²³ modelou tridimensionalmente jatos livres circulares turbulentos em desenvolvimento espacial, utilizando a versão do IMERSPEC aprimorada de Mariano⁴² junto à modelagem LES com modelos submalha de Smagorinsky e Smagorinsky Dinâmico. Como os jatos foram modelados em desenvolvimento espacial, foi necessária a implementação do método de *Buffer* junto à zona de forçagem com MDFM para garantir o amortecimento da condição do escoamento de saída do domínio útil e a reformulação da condição de entrada do domínio útil. Essa região formada pelo agrupamento do método de Buffer junto à zona de forçagem com MDFM recebeu o nome de domínio complementar. O processo de amortecimento de estruturas turbilhonares para esse escoamento é apresentado na Fig. 2 em que o domínio complementar é definido em $-8 \leq z/d \leq 0$ para d igual ao diâmetro do orifício de saída do jato. Os resultados foram validados comparando-se a resultados provenientes de um experimento material para o escoamento com $Re = 1050$. Destaca-se que a inserção de um ruído numérico para esse Re na entrada do domínio útil é necessária, visto que a exatidão de erro de máquina do IMERSPEC inviabiliza perturbações numéricas inerentes a métodos de baixa ordem.

Figura 2 – Estruturas turbilhonares sendo amortecidas pelo domínio complementar em modelagem de jato livre circular turbulento com o IMERSPEC.



Fonte: Moreira²³.

Até então apenas escoamentos isotérmicos haviam sido modelados com o IMERSPEC. Kinoshita⁵³ estendeu a metodologia para a modelagem de escoamentos com efeitos térmicos, explorando condições de contorno de primeira, segunda e terceira espécies. Foi verificado que para problemas físicos não isotérmicos, a metodologia apresenta uma taxa de convergência de

segunda ordem. Os resultados obtidos foram publicados posteriormente¹³.

Jacob⁵⁴ modelou escoamentos de esteiras tridimensionais em transição e turbulentas em desenvolvimento temporal utilizando abordagens LES e DNS. A necessidade de perturbações randômicas de diferentes amplitudes mostrou-se necessária para a correta modelagem desse tipo de escoamento cisalhante livre, tal qual observado por Moreira¹¹²³ para os jatos livres circulares, devido à alta acurácia do IMERSPEC. Os resultados quantitativos foram validados em termos do espectro de energia cinética turbulenta em diferentes instantes temporais.

A modelagem de escoamentos bifásicos bidimensionais dentro do IMERSPEC foi aperfeiçoada por Villela⁵⁵, que acoplou ao IMERSPEC dois métodos para a representação da interface móvel e deformável: *Front-Tracking* (FT) e *Volume of Fluid* (VOF). Os resultados foram publicados posteriormente⁵⁶.

O IMERSPEC também foi utilizado por Nascimento¹⁴ em problemas que envolvem interação fluido-estrutura, como a modelagem bidimensional simplificada de perfurações de poços de petróleo e gás, onde foi possível observar padrões de escoamentos nas tubulações de extração para diferentes razões de aspecto e números de Reynolds. Os resultados foram publicados posteriormente⁵⁷.

A metodologia IMERSPEC continua sendo aprimorada em diferentes instituições. No LATEF da Universidade Federal de Goiás, Monteiro; Mariano⁵⁸ utilizaram esta metodologia na modelagem de escoamentos bidimensionais sobre aerofólios e em turbinas eólicas verticais, obtendo resultados comparáveis à literatura. Ainda no contexto eólico, estudos relacionados a interação fluido-estrutura em turbinas eólicas verticais do tipo savonius foram realizados por Silva; Nascimento⁵⁹ utilizando a metodologia bidimensional IMERSPEC.

No Centro Acadêmico do Agreste da Universidade Federal de Pernambuco (CAA), Albuquerque; Villela; Mariano⁶⁰ investigaram o uso de composições híbridas de funções de interpolação/distribuição para a modelagem de escoamentos sobre aerofólios esguios. Ainda no CAA, Albuquerque; Villela; Mariano⁶¹ adaptaram o modelo URANS de fechamento de turbulência Spalart-Allmaras junto ao IMERSPEC para simulações de escoamentos turbulentos em canal plano, com resultados validados contra DNS para Re_τ igual a 180, 550 e 1000.

Recentemente, no MFLab, iniciou-se a modelagem de escoamentos parietais bidimensionais utilizando o IMERSPEC⁶². A modelagem clássica do escoamento laminar e bidimensional ao longo de uma placa plana, com $Re = 10^4$, foi realizada com essa metodologia de alta ordem, alcançando-se exatidão computacional integral no componente u e na quarta casa decimal para o componente v da velocidade.

Desde sua apresentação, o IMERSPEC foi utilizado nas mais diversas modelagens, todavia, suas simulações tridimensionais foram realizadas apenas para escoamentos cisalhantes livres. Com base nisso, o principal avanço do presente trabalho é o desenvolvimento dessa metodologia tridimensionalmente para modelagens de escoamentos parietais, expandindo o

escopo de aplicação do IMERSPEC para uma classe de escoamentos até então não explorada com essa metodologia.

2.5 Revisão do Fenômeno de *aliasing*

Apesar da hibridização, o IMERSPEC continua sendo uma metodologia espectral de Fourier e assim pode apresentar fenômenos numérico-computacionais inerentes a essa classe de métodos. Um deles é o fenômeno de *aliasing*. Conforme identificado por Boyd⁶³, o primeiro trabalho a identificar esse fenômeno em modelagens espectrais foi escrito em 1959 por Norman Phillips. Metodologias espectrais apresentam alta exatidão, entretanto, um fenômeno numérico que pode comprometer a correlação entre a modelagem e o fenômeno físico de interesse é o *aliasing*. Blaisdell; Spyropoulos; Qin⁶⁴ analisaram espectralmente as diferentes discretizações do termo advectivo em modelagens unidimensionais da equação de Burgers e em simulação de grandes estruturas da turbulência, LES, modelando o decaimento de turbulência isotrópica em escoamentos compressíveis. As análises indicaram que tanto as formas divergentes e não-divergentes do termo advectivo promovem interações entre os seus números de onda que resultam em um efeito de blocagem espectral que origina o fenômeno de *aliasing*. Segundo os resultados de Blaisdell; Spyropoulos; Qin⁶⁴, apenas a forma antissimétrica inibe a formação desse fenômeno numérico sem a necessidade de qualquer processo de filtragem ou de *dealiasing*.

Embora as análises feitas por Blaisdell; Spyropoulos; Qin⁶⁴ tenham sido realizadas para o termo advectivo, muitas de suas conclusões contribuíram significativamente para o presente trabalho em relação à discretização do termo difusivo das equações de Navier-Stokes, conforme discutido no capítulo 5.

Ainda considerando o problema de *aliasing* em métodos pseudoespectrais de Fourier, Hou; Li⁶⁵ realizaram um estudo comparando a clássica regra de 2/3 de *dealiasing* com uma nova metodologia, proposta por eles, em que há a filtragem espectral dos altos números de onda. Todas as duas metodologias foram realizadas a partir de um código pseudoespectral modelando fenômenos periódicos. Neste estudo realizou-se comparações unidimensionais através da equação de Burgers e tridimensionais considerando as equações de Euler e, também, as equações de Navier-Stokes. Os autores identificaram a nova metodologia de filtragem espectral proposta como a melhor alternativa para os casos simulados, retraindo cerca de 12% a 15% dos modos de Fourier para cada dimensão a mais do que a regra clássica de *dealiasing*. Destaca-se que todas as simulações realizadas no trabalho utilizaram condições de contorno periódicas e, ao contrário do que se espera ao utilizar um processo de *dealiasing*, foi identificado o fenômeno de Gibbs em modelagens de soluções computacionais singulares quando utilizada a regra clássica de 2/3 de *dealiasing*, enquanto a filtragem espectral não apresentou tal fenômeno.

Hou; Li⁶⁵ afirmam em seu estudo que não há um gasto computacional maior no processo de *dealiasing* proposto em relação à regra clássica, entretanto, deve-se lembrar que esse processo

numérico-computacional em si apresenta significativo gasto computacional e, por mais otimizado que seja, há ainda perda de exatidão computacional. Esse último aspecto é a característica mais marcante de metodologias de alta ordem e deve-se ter um esforço em sua preservação.

O problema de *aliasing* também é explorado de forma parcial por Beck *et al.*⁶⁶, provando que esse fenômeno numérico não ocorre apenas para modelagens relacionadas à transformada de Fourier. Os autores investigaram a acurácia e eficiência do método de alta ordem de Galerkin descontínuo em simulações sub-resolvidas de escoamentos transicionais e turbulentos. Para realizar o estudo foi utilizado um processo de *dealiasing* dos termos não lineares para evitar o fenômeno e recuperar as propriedades fundamentais da formulação de Galerkin descontínua. Três escoamentos clássicos foram modelados neste trabalho: o escoamento sobre cilindro circular, o escoamento transicional sobre o aerofólio SD7003 e o escoamento confinado sobre colinas periódicas.

Os resultados de Beck *et al.*⁶⁶ provam que sem qualquer modelagem explícita de LES, consegue-se, utilizando puramente essa metodologia espectral, a mesma exatidão ou uma exatidão maior na modelagem de escoamentos com números de Reynolds, Re , medianos do que modelagens que utilizam estabelecidos esquemas LES associados a métodos de discretização inferiores. Os autores admitem, entretanto, que para modelagens com valores elevados de Re , *i.e.*, escoamentos com maior grau de turbulência, apenas o esquema de Galerkin descontínuo é insuficiente na correta modelagem do fenômeno físico.

Ainda assim, o trabalho de Beck *et al.*⁶⁶ ilustra o grande potencial de modelagens espectrais de alta ordem na modelagem de fenômenos físicos em decorrência da alta exatidão alcançada.

Modelagens LES do escoamento clássico em canais e sobre colinas periódicas foram realizadas através de uma combinação de alta ordem de um esquema espacial de diferenças espectrais com modelagem de parede para simulações LES, do inglês *Wall-Modelled Large-Eddy Simulation* (WMLES)⁶⁷. Essa hibridização é implementada em malhas não uniformes e não estruturadas e é uma variação da metodologia de elementos finitos descontínuos. Essa metodologia apresenta significativa acurácia, principalmente devido à troca de informação realizada para a modelagem de parede em escoamentos turbulentos, em que o ponto mais afastado da parede também contribui na resolução numérica, um comportamento global como esse é muito comum em abordagens espectrais.

O esquema de Galerkin descontínuo de alta ordem, utilizado por Beck *et al.*⁶⁶, também foi utilizado dentro do desenvolvimento de um *framework* livre, FLEXI, capaz da resolução computacional das equações de balanço para a fluidodinâmica compressível em clusters de CPU⁶⁸. O diferencial dessa abordagem livre é a proposta de utilização de métodos de alta ordem em todas as etapas relacionadas a uma simulação computacional e não apenas no *solver*.

Observa-se ao longo dos trabalhos comentados que, apesar de modelagens espectrais

apresentarem o potencial de alta exatidão, o fenômeno de *aliasing* pode impedir essa alta exatidão. A fim de remediar tal fenômeno, são utilizadas técnicas de *dealiasing* que correspondem a processos de filtragem que, como consequência, diminuem a exatidão dessas metodologias com um aumento do custo computacional de metodologias já onerosas.

2.6 Revisão de Paralelizações GPGPU

Desde o desenvolvimento do GPGPU, diversas metodologias utilizaram esse tipo de paralelização. Stone *et al.*⁶⁹ implementou em CUDA uma família de *solvers* de matrizes diagonais em blocos paralelos e analisou o desempenho de três códigos CFD utilizando GPGPU. Dentre as principais conclusões, destaca-se que utilizando-se parcialmente os GPUs obteve-se ganhos medidos de 3,2 a 4,3 vezes nos núcleos de volumes finitos implementados. Por outro lado, estimou-se que a eliminação completa da transferência de dados CPU-GPU poderia elevar esse ganho computacional até 21 vezes. Assim, destaca-se que para maximizar os ganhos de performance deve-se minimizar ao máximo possível a transferência de dados entre CPU e GPU.

Xiang *et al.*⁷⁰ também desenvolveu um código bidimensional inteiramente em CUDA através das abordagens HSMAC, do inglês *Highly Simplified Marker and Cell*, e do acoplamento pressão-velocidade SIMPLE, em inglês *Semi-Implicit Method for Pressure Linked Equations*, para resolver o clássico problema do escoamento em cavidade com tampa deslizante. Para uma malha de 512×512 , foi obtido um *speedup* igual a 78 para a metodologia HSMAC e um de 32 com o uso do SIMPLE em precisão simples. A metodologia HSMAC apresenta melhor performance visto que é necessária apenas uma sincronização global do código para realizar a correção da velocidade e pressão em cada iteração do algoritmo.

Em um contexto onde computação de alto desempenho é feita em CPU e GPU, Oyarzun *et al.*⁷¹ desenvolveram um código computacional capaz de modelar escoamentos turbulentos com a abordagem LES que tanto pode operar unicamente em CPUs quanto em GPUs. Suas análises foram realizadas no supercomputador MinoTauro localizado no Centro de Supercomputação de Barcelona, BSC, que apresenta grande quantidade de CPUs e GPUs disponíveis. Para os testes realizados foram utilizados nós compostos por dois CPUs 6-core Intel Xeon E5649 e dois GPUs NVIDIA M2090, com conexão via interconexão InfiniBand QDR. Foram modelados dois escoamentos: o escoamento sobre a geometria do carro ASMO, do alemão *Aerodynamisches Studien Modell*, com 5,5 milhões de células e o escoamento sobre uma cavidade quadrada com 1,2 milhões de células. Nota-se que Oyarzun *et al.*⁷¹ conseguiram estimar com relativa exatidão a performance do código tanto para a configuração com multi-CPU quanto para a configuração com multi-GPUs. Também é observado que, embora o código desenvolvido apresente melhor escalabilidade em CPUs do que em GPUs, a melhor performance para os dois escoamentos modelados é obtida com GPUs.

Modelando o escoamento subsônico e supersônico ao redor de uma esfera e a quebra de

gotícula líquida por um choque mecânico, Korneev; Levchenko⁷² apresentaram uma metodologia de segunda e terceira ordem de precisão no contexto da paralelização heterogênea GPU-CPU. A metodologia desenvolvida combina o método de Galerkin descontínuo com um algoritmo otimizado para GPUs denominado *DiamondTower*, utilizando o método de Runge-Kutta para o avanço temporal. De acordo com Korneev; Levchenko⁷², esse particionamento do domínio computacional é mais eficiente do que o usual particionamento nas três direções para a GPU NVIDIA Tesla V100, com um processamento de mais de 10^8 células por segundo. Os casos simulados apresentaram uma performance 8 vezes superior em uma GPU do que quando executados apenas em CPUs i5.

Esquemas numéricos de alta ordem também foram implementados utilizando GPU/CPU por Lei *et al.*⁷³ para a resolução das equações de Euler. Utilizando o GPU NVIDIA Tesla P100, os esquemas numéricos MUSCL, do inglês *Monotonic Upstream-centered Scheme for Conservation Laws*, e NND, do inglês *Non-Oscillatory and Non-Free-parameters Dissipations*, foram implementados. Lei *et al.*⁷³ atingiram um *speedup* de 325 em relação ao código executado em CPU para 10000 células modelando o problema unidimensional de Riemann. Para a modelagem bidimensional do escoamento supersônico sobre um degrau, os autores obtiveram um *speedup* de 260 vezes para o esquema MUSCL (800×400 células) e 144 vezes com o NND (400×200 células).

A viabilidade de simulações em tempo real de escoamentos turbulentos sobre ambientes urbanos utilizando GPGPU junto a uma variação do método de Lattice Boltzmann (LBM) foi explorada por Lenz *et al.*⁷⁴. A variação de LBM utilizada, denominada método de Lattice Boltzmann cumulativo, não necessita de modelagem LES explícita e é estável para malhas grosseiras. A junção dessa metodologia em um código paralelizado com CPU/GPU teve um desempenho comparável a metodologias pseudoespectrais com um custo computacional significativamente menor. As simulações foram realizadas no GPU Tesla P100 operando em precisão simples, alcançando até 706 milhões de atualizações de nós por segundo, equivalente a uma taxa de 2,49 tempo físico por custo computacional da simulação — valor significativamente superior aos 0,017 obtidos em 160 núcleos de CPU.

Alternativamente à API CUDA, exclusiva de GPUs da NVIDIA, existe a API OpenACC, que pode ser implementada em GPUs de outras marcas com usabilidade similar à implementação paralela OpenMP. Xue; Jackson; Roy⁷⁵ adaptaram o código CFD SENSEI em um paralelismo híbrido composto de MPI e OpenACC em 16 GPUs NVIDIA (P100 e V100) associados com 16 núcleos de CPU Xeon E5-2680V4. A performance do código obteve um ganho de 30 e 80 vezes maior, respectivamente, para as configurações de multi-GPUs com os modelos P100 e V100 em relação ao código executado apenas com CPU, com desempenho ligeiramente inferior ao CUDA.

Piscaglia; Ghioldi⁷⁶ adaptaram bibliotecas do OpenFOAM⁷⁷ ao uso de GPUs, criando a biblioteca AmgX4Foam responsável por conectar a biblioteca da NVIDIA, que resolve sistemas

lineares esparsos utilizando CUDA, ao OpenFOAM. Os dois casos validados — reatores em lote e combustão supersônica em *Scramjet* — atingiram um *speedup* 8,3 vezes mais rápido em relação aos casos simulados unicamente em CPUs. As implementações são compatíveis com versões mais recentes do OpenFOAM e suportam escalabilidade em múltiplos nós de CPU e GPUs.

Um código computacional em CUDA C foi desenvolvido por da Silva Reis *et al.*⁷⁸ para acelerar simulações de transferência de energia térmica conjugada, CHT, em uma cavidade quadrada aquecida, obtendo uma redução de 99,7% no custo computacional quando comparada com uma simulação em CPU, utilizando a GPU NVIDIA RTX 4090.

Outro avanço significativo foi realizado por Souza *et al.*⁷⁹ através da modelagem de um jato compressível, turbulento, circular e isolado utilizando a metodologia de alta ordem CABARET junto a GPGPU, com abordagem WMLES. A validação foi feita a partir de resultados experimentais do caso SHJAR para o bocal SMC00 cedidos pela NASA. Souza *et al.*⁷⁹ realizaram essa modelagem para uma malha de 40 milhões de células e com um tempo advectivo igual a 1400 unidades em menos de 24 h utilizando a GPU NVIDIA A100.

Com base nos trabalhos comentados, pode-se construir uma tabela comparativa, Tab. 1, dentre os diferentes cenários abordados em cada trabalho. Destaca-se que algumas informações estão ausentes nos trabalhos originais e por isso utiliza-se o -"na Tab. 1. Nota-se que a paralelização GPGPU mostrou-se uma alternativa viável para reduzir o custo computacional em diferentes metodologias e, portanto, infere-se que essa técnica de paralelização apresentará significativa redução de custo computacional para a metodologia IMERSPEC. Hipótese a ser considerada e validada nos capítulos 5 e 6.

Tabela 1 – Síntese comparativa de trabalhos com paralelização GPGPU abordados.

Referência	Método	GPU (Arquitetura)	CPU Comparada	Precisão	Dimensão	Caso simulado	Speedup
Stone <i>et al.</i> ⁶⁹	MBFLO3	NVIDIA C2050 (Fermi)	Intel X5677 Xeon	Simples/Dupla	3D	Escoamento sobre rampa	3,2–4,3
Xiang <i>et al.</i> ⁷⁰	HSMAC	GTX 980 (Maxwell)	Intel E5-2640V3	Simples	2D	Cavidade	78
Xiang <i>et al.</i> ⁷⁰	SIMPLE	GTX 980 (Maxwell)	Intel E5-2640V3	Simples	2D	Cavidade	32
Oyarzun <i>et al.</i> ⁷¹	DNS/LES	8–128 NVIDIA M2090 (Fermi 2.0)	8–128 Intel Xeon E5649 (6-core)	Dupla	3D	ASMO / Cavidade	3–8
Korneev; Levchenko ⁷²	DG + DiamondTorre	Tesla V100 (Volta)	CPU i5	–	3D	Esfera / Gotícula	–
Lei <i>et al.</i> ⁷³	MUSCL	Tesla P100 (Pascal)	Intel Xeon E5-2640V4	Dupla	1D	Problema de Riemann	25,9
Lei <i>et al.</i> ⁷³	MUSCL	Tesla P100 (Pascal)	Intel Xeon E5-2640V4	Dupla	2D	Degrau supersônico	260
Lenz <i>et al.</i> ⁷⁴	LBM cumulativo	Tesla P100 (Pascal)	160 núcleos CPU	Simples	3D	Ambiente urbano	146
Xue; Jackson; Roy ⁷⁵	MPI + OpenACC	16 Tesla P100 (Pascal)	16 núcleos Xeon E5-2680V4	Dupla	3D	NACA 0012 / Asa	30
Piscaglia; Ghioldi ⁷⁶	AmgX4Foam	Tesla V100 (Volta)	24 núcleos Intel Xeon Gold 6248	Dupla	3D	Reator / Scramjet	9,3
da Silva Reis <i>et al.</i> ⁷⁸	FVM/SIMPLE	RTX 4090 (Lovelace)	Intel Core i9-12900KF	Dupla	2D	Cavidade quadrada (CHT)	333
Souza <i>et al.</i> ⁷⁹	CABARET + WMLES	NVIDIA A100 (Ampere)	–	–	3D	Jato compressível	–

Fonte: Próprio autor.

Salienta-se, com base na Tab. 1, que o modelo de GPU/CPU utilizado, o algoritmo empregado e, principalmente, o ambiente de comparação (em especial o número de núcleos de CPU tomado como referência) impactam significativamente o *speedup* obtido na comparação entre a execução em CPU e em GPGPU. Nota-se, por exemplo, que comparações contra CPUs mono-core tendem a produzir *speedups* muito superiores (333 obtido por da Silva Reis *et al.*⁷⁸) àsquelas realizadas contra clusters multi-core (entre 3 e 146), evidenciando que o ganho reportado depende fortemente da referência adotada, e não apenas da arquitetura da GPU empregada.

Nesse capítulo foram apresentados os estados da arte das principais áreas da presente tese. Primeiro, foram apresentados os modelos submalha relacionados a modelagem LES, destacando quais foram usados nesse trabalho. Logo após, foi abordado trabalhos numérico-computacionais e materiais que modelaram o escoamento turbulento sobre um degrau descendente. Em seguida, se discutiu métodos da Fronteira Imersa destacando a suas limitações. Posteriormente, foi abordada a metodologia IMERSPEC, que é uma hibridização do MFI com MPF, nativa do MFLab. Também foi discutido o problema numérico-computacional de *aliasing* junto com procedimentos clássicos de *dealising*. Por fim, a possibilidade de paralelização por GPGPU foi apresentada destacando o potencial de minimização do custo computacional dos algoritmos de CFD com essa técnica.

Dessa forma, continuando com o desenvolvimento da tese, no próximo capítulo será apresentada a formulação matemática necessária para a modelagem dos escoamentos considerados, laminares ou turbulentos.

3 MODELAGEM MATEMÁTICA

Neste capítulo será apresentada a formulação matemática de escoamentos incompressíveis utilizada no IMERSPEC. Assim, será apresentada a formulação de escoamentos gerais e em seguida será desenvolvida a abordagem de Simulações de Grandes Estruturas, do inglês *Large Eddy Simulations*, (LES). Após a discussão da metodologia LES, são detalhados três modelos funcionais de fechamento de turbulência submalha utilizados no presente trabalho: Smagorinsky clássico junto a função empírica amortecedora *ad-hoc* Van-Driest, WALE e dinâmico Germano-Lilly.

3.1 Modelagem de Escoamentos Incompressíveis

Considerando a hipótese do contínuo, fluidos newtonianos, escoamentos incompressíveis, tem-se a seguinte formulação matemática:

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (3.1)$$

$$\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j} (u_i u_j) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + f_i, \quad (3.2)$$

em que u_i representa a velocidade do fluido na direção i , x_i corresponde a posição geométrica na direção i considerada, t equivale ao tempo físico, $p = p_e / \rho$ em que p_e representa pressão estática e ρ é a massa específica do fluido newtoniano considerado, ν é a viscosidade cinemática molecular do fluido de interesse e $f_i = f_i^o / \rho$ no qual f_i^o representa um termo fonte de força de campo por unidade de massa, podendo englobar forças gravitacionais, de empuxo, eletromagnéticas ou, no contexto do presente trabalho, forças de fronteira imersa, conforme detalhado no Capítulo 4. Destaca-se que todas as variáveis tem suas dimensões definidas de acordo com o Sistema Internacional de Unidades (SI).

A equação da continuidade, Eq. 3.1, e as equações de balanço de *momentum* linear, Eq. 3.2, em suas formas apresentadas podem ser utilizadas para a modelagem dos escoamentos incompressíveis laminares e turbulentos. Entretanto, as formas das equações diferenciais apresentadas exigem que a malha computacional utilizada resolva todos os graus de liberdade do escoamento modelado para a correta representação física²⁶. Essa abordagem na resolução completa dos graus de liberdade em um escoamento turbulento é denominada por Simulação Computacional Direta, do inglês *Direct Numerical Simulation* (DNS). Esse é o melhor modelo, a nível do contínuo, para a modelagem de escoamentos dessa classe, ainda sim, seu custo computacional pode ser proibitivo ou impossível devido a alta resolução computacional necessária e

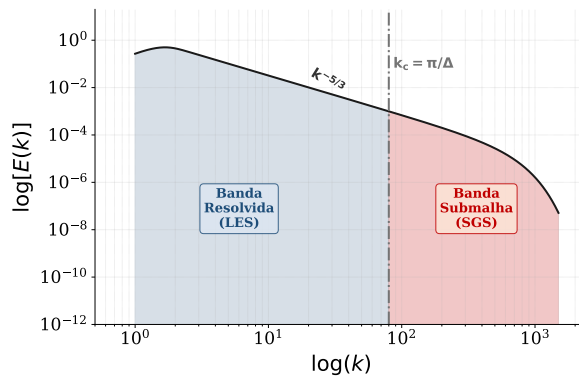
os recursos computacionais disponíveis. No presente trabalho, DNS não é empregado, mas sim, a modelagem de Simulações de Grandes Estruturas (LES).

3.2 Simulação de Grandes Estruturas (LES)

Esta abordagem reduz o número de graus de liberdade da modelagem através da filtragem das variáveis das Eq. 3.1 e 3.2. O processo de filtragem elimina altos números de onda, k_i , do sinal original. Assim, o comprimento característico do sinal filtrado permite o uso de uma malha mais grosseira para representar corretamente o fenômeno físico, *i.e.*, menos graus de liberdade a ser resolvidos com a malha computacional. Destaca-se, que a solução computacional ainda apresenta um número de onda de corte, $k_c = \pi/\Delta_i$, diretamente relacionado a malha computacional usada através do incremento espacial, Δ_i , e neste contexto, para LES ser efetiva na modelagem, k_c deve estar na região inercial da distribuição de densidade espectral de energia cinética específica turbulenta.

Na abordagem LES considera-se a distribuição espectral de energia cinética específica turbulenta, $E(f)$ associada ao sinal original oriundo de DNS ou de um experimento material. O número de onda de corte, k_c , claramente delimita até onde foi filtrado o sinal e, portanto, a banda resolvida, $k_i \leq k_c$ enquanto $k_i > k_c$ representa a parte da informação que não foi calculada. Nenhuma informação não calculada pode ser calculada diretamente e assim serão utilizados modelos submalha para modelar seu comportamento (seção 3.3). A Figura 3 esquematiza a região de $E(k)$ resolvida e sua parte que deverá ser modelada em função de k_c , enfatiza-se que caso DNS fosse a abordagem utilizada a banda de números de onda modelados tenderia a zero.

Figura 3 – Esquema da abordagem LES representada em um espectro $E(k)$.



Fonte: Próprio autor.

3.2.1 Filtragem Espacial

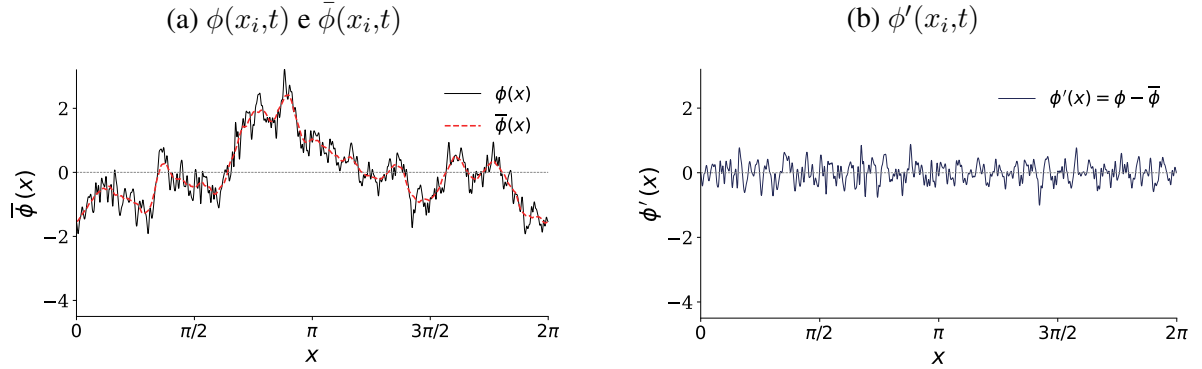
Conforme mencionado, a abordagem LES se baseia na filtragem das informações do sinal original. Assim, considerando uma informação genérica de um sinal, $\phi(x_i, t)$, obtido por

um experimento material ou DNS contendo todas as suas frequências ou números de onda, pode ser decomposto em uma parte filtrada, $\bar{\phi}(x_i, t)$, acrescida da flutuação dessa informação geral em relação ao sinal filtrado e o original, $\phi'(x_i, t)$. Essa decomposição do sinal original é exemplificada pelas Fig. 4 e é representada matematicamente pela Eq. 3.3,

$$\phi(x_i, t) = \bar{\phi}(x_i, t) + \phi'(x_i, t). \quad (3.3)$$

Considerando a Fig. 4a, nota-se que o sinal filtrado corresponde a suavização do sinal original, conforme previamente observado enquanto a Fig. 4b ilustra a distribuição da disparidade entre o sinal filtrado e original, ponto a ponto, *i.e.*, $\phi'(x_i, t)$.

Figura 4 – Esquema de decomposição do sinal genérico $\phi(x_i, t)$.



Fonte: Próprio autor.

Definida a forma como a decomposição do sinal original é feita, convém apresentar como é feito de fato o procedimento de filtragem. A filtragem é representada no domínio físico como um produto de convolução entre o sinal original, $\phi(x_i, t)$, e a função filtro, G , e é definida formalmente conforme Eq. 3.4²⁸:

$$\bar{\phi}(x_i, t) = G * \phi = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \phi(\xi, t') G(x - \xi, t - t') dt' d^3\xi, \quad (3.4)$$

em que dt' é o integrando temporal, ξ representa o integrando espacial em cada dimensão e a convolução é dependente dos parâmetros k_c e uma frequência de corte, f_c . Usualmente a abordagem LES possui uma filtragem explícita, ativamente usando uma função filtro G , ou implícita, filtragem por malha computacional inferior a requerida para uma modelagem DNS, relativo ao espaço e não ao tempo. Entretanto, conforme Sagaut²⁸, quando um filtro espacial é aplicado, automaticamente é imposto um filtro temporal devido a formulação matemática das equações de Navier-Stokes torna possível associar um tempo característico com um comprimento característico. Assim, a definição formal representa corretamente um processo de filtragem.

Destaca-se que o filtro considerado pela Eq. 3.4 é isotrópico. Assim, as propriedades do filtro são independentes de x_i e da orientação de referência no espaço, o que implica que o filtro

é aplicado em um domínio computacional não limitado e que o comprimento característico de corte, Δ_i , é constante e idêntico em todas as direções do escoamento²⁸.

Outro aspecto a ser destacado sobre o processo de filtragem é a operação em si de convolução definida no domínio físico. O uso da transformada de Fourier na análise de escoamentos turbulentos é uma alternativa. A transformada de Fourier será discutida em detalhes no capítulo 4 entretanto, de acordo com o teorema da convolução, uma convolução no domínio físico equivale a um produto no domínio espectral, e vice-versa⁸⁰. Assim, caso G seja aplicado explicitamente no sinal, essa operação pode ser feita no domínio espectral com:

$$\hat{\bar{\phi}} = \hat{G}\hat{\phi}, \quad (3.5)$$

em que o símbolo $\hat{}$ representa a propriedade transformada para o espaço espectral. Assim, o processo de filtragem pode ser feito de forma simplificada e com menor custo computacional no domínio de Fourier.

Como o presente trabalho utiliza o MPF, essa vantagem é explorada na implementação computacional do modelo submalha de Dinâmico de Germano (seção 3.3.3). Nas demais abordagens da presente tese, o filtro é imposto ao domínio computacional de forma implícita através de uma malha computacional inferior ao requerido para uma abordagem DNS. A filtragem implícita através de uma malha computacional é representada como a imposição de um filtro-caixa²⁶.

3.2.2 Equações de Balanço de Massa e de *Momentum* Linear Filtradas

A fim de se obter as equações de Navier-Stokes filtradas, considera-se um filtro com as seguintes propriedades:

- Conservação de constantes: $\bar{a} = a$ em que a é uma constante qualquer;
- Linearidade: $\overline{\phi + \psi} = \bar{\phi} + \bar{\psi}$ para ϕ e ψ iguais a sinais genéricos;
- Comutação com derivadas (temporais ou espaciais): $\overline{\frac{\partial \phi}{\partial t}} = \frac{\partial \bar{\phi}}{\partial t}$.

Enfatiza-se que filtros que apresentam as propriedades elencadas não são necessariamente operadores de Reynolds²⁸, *i.e.*, $\bar{\bar{\phi}} \neq \bar{\phi}$ e $\bar{\phi'} \neq 0$. Com as propriedades e a definição de decomposição do sinal obtido pela Eq. 3.3, chega-se às equações da continuidade e de Navier-Stokes filtradas:

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0, \quad (3.6)$$

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\overline{u_i u_j}) = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] + \bar{f}_i. \quad (3.7)$$

Analisando-se a Eq. 3.7, tem-se o tensor submalha de Boussinesq-Reynolds, $\overline{u_i u_j}$. Este tensor representa um problema na implementação da Eq. 3.7 visto que seria necessário o sinal original u_i para a sua determinação. Portanto, este tensor deve ser reescrito como uma função de $\bar{u}$, que é sinal filtrado computado na abordagem LES. A equivalência do tensor de submalha de Boussinesq-Reynolds para uma expressão matemática de tensores não é única mas, na presente tese, é utilizada a formulação de Germano²⁵:

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j, \quad (3.8)$$

em que τ_{ij} é o tensor global. Com base na Eq. 3.8, pode-se afirmar que o tensor modela a diferença entre o tensor submalha de Boussinesq-Reynolds obtido caso se conhecesse o sinal original u e a atividade não linear que pode ser obtida com os campos de velocidade filtrados. Com a definição do tensor global, a Eq. 3.7 é redefinida como:

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\bar{u}_i \bar{u}_j) = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} - \tau_{ij} \right) \right] + \bar{f}_i. \quad (3.9)$$

O termo $\bar{u}_i \bar{u}_j$ é quadrático e, portanto, contém em teoria números de onda superiores ao número de onda de corte k_c de cada componente isolada. Isso é evidente no domínio espectral, onde o produto no espaço físico equivale a uma convolução espectral cujos modos resultantes se estendem até $2k_c$ ⁸⁰. Como o termo advectivo filtrado resultante apresenta números de onda superiores aos representáveis com a malha computacional disponível, sua representação exata exigiria uma malha duas vezes mais refinada em cada direção do escoamento em relação à malha de $\bar{u}$ ²⁸. Um refinamento assim é irreal, caso a malha computacional mais refinada esteja disponível, ela será usada e não conservada para o cálculo de apenas um termo das Eq. 3.9. Ainda sim, caso essa malha fosse viável, o refinamento necessário implicaria um aumento de oito vezes no número de pontos originais.

Na prática, o termo é calculado na mesma malha computacional, de modo que o que se resolve efetivamente é $\overline{\bar{u}_i \bar{u}_j}$, ou seja, uma versão filtrada do produto, os modos entre k_c e $2k_c$ são perdidos, e, se não tratados, dobram-se de volta ao espectro resolvido caracterizando o erro de *aliasing*⁶³.

Outro termo significativo da Eq. 3.9 é o τ_{ij} . Como a abordagem LES pode ser entendida como uma metodologia intermediária entre DNS e URANS. Entende-se que o operador média é o extremo do operador filtragem em que todos os números de onda são filtrados restando apenas o comportamento médio do sinal filtrado²⁶. Por outro lado, a abordagem LES tende a abordagem DNS quando nenhuma redução de graus de liberdade do escoamento é feito e assim $\tau_{ij} \rightarrow 0$.

3.2.3 Problema de Fechamento e Hipótese de Boussinesq

Para a resolução numérico-computacional das equações de Navier-Stokes filtradas, Eq. 3.9, junto a equação da continuidade filtrada, Eq. 3.6 deve-se ter um custo adicional para a modelagem do tensor global, τ_{ij} . Esse tensor é simétrico, assim, possui 6 componentes indeterminadas. Tem-se assim um problema de fechamento onde há mais variáveis do que equações²⁶. A fim de se resolver o problema de fechamento da turbulência, é necessário a modelagem do tensor global também denominado como tensor submalha, τ_{ij}^{SGS} , o qual deve seguir duas restrições: física e numérica²⁸.

A restrição física é necessária para que a modelagem SGS seja consistente para a representação do fenômeno físico, *i.e.*, o modelo deve²⁸: manter as mesmas propriedades básicas das equações originais (Eq. 3.6 e 3.9), por exemplo, a invariância galileana; ser nulo quando a abordagem LES tende a DNS; induzir um efeito de mesma natureza nas Eq. 3.9 que está sendo modelado, por exemplo, o difusivo; e não impactar negativamente na dinâmica das grandes estruturas turbilhonares resolvidas, especialmente não inibir os mecanismos precursores do escoamento.

Por outro lado, a restrição numérica é fundamental, uma vez que o modelo SGS será aplicado em um contexto numérico-computacional. Essa restrição impõe que a modelagem do tensor global não possua custo computacional proibitivo e não desestabilize a simulação computacional. Adicionalmente, o modelo deve ser insensível ao método de discretização adotado, garantindo que os efeitos físicos induzidos pela modelagem submalha não sejam inibidos pela discretização utilizada²⁸.

Seguindo as restrições mínimas, os modelos de fechamento SGS podem ser classificados em duas categorias principais²⁸: modelos estruturais, que constroem uma aproximação direta do tensor τ_{ij}^{SGS} a partir do campo de velocidade filtrado $\bar{u}_i$ ou de uma expansão em série formal, de modo que a hipótese de fechamento modela exclusivamente τ_{SGS} ; e modelos funcionais, que modelam a ação do tensor de submalha sobre as escalas resolvidas, por exemplo, introduzindo um efeito dissipativo ou dispersivo, sem necessariamente reproduzir a estrutura τ_{ij}^{SGS} . Nesse caso, a hipótese de fechamento modela a forma $\frac{\partial \tau_{ij}^{SGS}}{\partial x_j}$ e não apenas o tensor.

Essas duas abordagens não requerem o mesmo conhecimento prévio sobre a dinâmica do escoamento e, em teoria, não oferecem o mesmo potencial em termos de qualidade dos resultados²⁸. No presente trabalho adotam-se modelos funcionais, cuja estratégia de fechamento é baseada na hipótese de Boussinesq. Essa consideração modela o tensor de Boussinesq-Reynolds com o conceito de viscosidade turbulenta, ν_{SGS} , o qual Boussinesq idealizou essa modelagem através de uma analogia com o modelo de Stokes para as tensões viscosas moleculares^{26,81}. Com essa analogia, tem-se:

$$-\tau_{ij}^{SGS} = 2\nu_{SGS}\bar{S}_{ij} - \frac{2}{3}K\delta_{ij}, \quad (3.10)$$

em que $\bar{S}_{ij}$ é o tensor filtrado de taxa de deformação e é igual $\frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right)$, K é a energia cinética específica turbulenta igual a:

$$K = \frac{1}{2} \tau_{ii} = \frac{1}{2} \overline{u_i u_i} = \frac{1}{2} \left(\overline{u_1^2} + \overline{u_2^2} + \overline{u_3^2} \right); \quad (3.11)$$

e δ_{ij} é o delta de Kronecker que é usado para a compatibilizar a definição de K com o traço do tensor global. Com base na definição da Eq. 3.10, a Eq. 3.9, pode ser reduzida a:

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\bar{u}_i \bar{u}_j) = -\frac{\partial \bar{p}^m}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] + \bar{f}_i, \quad (3.12)$$

em que $\bar{p}^m$ é a pressão modificada e é igual a $\bar{p} + \frac{2}{3} K$ considerando $\frac{\partial}{\partial x_j} \left(\frac{2}{3} K \delta_{ij} \right) = \frac{2}{3} \frac{\partial K}{\partial x_i}$ e ν_{eff} é a viscosidade efetiva igual a $\nu + \nu_{SGS}$, destacando que ν_{SGS} é uma propriedade do escoamento²⁶.

Dentro da metodologia o termo advectivo é discretizado em sua forma antissimétrica. Matematicamente, para escoamentos incompressíveis, a forma divergente, $\frac{\partial(\bar{u}_j \bar{u}_i)}{\partial x_j}$, é equivalente a forma não divergente, $\bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j}$, entretanto isso não ocorre para as suas respectivas formas discretas. Assim, utiliza-se a média entre as duas formas, o que resulta na forma antissimétrica, para estabilidade numérico-computacional conforme apresentado por Canuto *et al.*⁸² e validado para a metodologia IMERSPEC no trabalho de Mariano¹⁰.

Considerando-se a forma antissimétrica do termo advectivo, a metodologia IMERSPEC utiliza a seguinte formulação matemática para a abordagem LES: a equação da continuidade filtrada, aqui reescrita:

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0, \quad (3.6)$$

e as equações de Navier-Stokes filtradas com a hipótese de Boussinesq inclusa e a forma discreta antissimétrica do termo advectivo,

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{1}{2} \left[\frac{\partial(\bar{u}_i \bar{u}_j)}{\partial x_j} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} \right] = -\frac{\partial \bar{p}^m}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] + \bar{f}_i. \quad (3.13)$$

O sistema formado pelas Eqs. 3.6 e 3.13 constitui a formulação matemática adotada no presente trabalho para a abordagem LES e será utilizado como base para a discretização numérica apresentada no Capítulo 4. O valor de ν_{SGS} não é conhecido a priori e deve ser determinado por um modelo de submalha.

3.3 Modelos de Fechamento Funcionais Submalha

Antes de apresentar as especificidades de cada um dos modelos de fechamento funcionais submalha utilizados, convém elencar algumas hipóteses que são necessárias para o uso desses

modelos. São elas²⁸:

- O efeito das estruturas turbilhonares submalha sobre as resolvidas é, essencialmente, energético. Assim, o balanço de transferência de energia entre os dois tipos de estruturas é suficiente para modelar o comportamento das estruturas turbilhonares submalha;
- Nos modelos utilizados na presente tese, é modelada apenas a transferência direta de energia em direção as estruturas turbilhonares submalha. Portanto, não se modela a transferência inversa denominada por *backscatter*;
- Um comprimento característico e um tempo característico são suficientes para descrever estruturas turbilhonares submalha.

Com as hipóteses elencadas, destaca-se que todos os modelos de fechamento funcionais submalha explorados no presente trabalho se baseiam em propriedades das estruturas turbilhonares resolvidas. Dessa forma, ν_{SGS} é determinada a partir de quantidades globais do escoamento relativas especificamente as estruturas resolvidas²⁸.

O problema de cada um dos modelos para determinar ν_{SGS} , passa a ser determinar o comprimento e o tempo característicos das estruturas submalha. Em dois dos modelos submalha utilizados, Smagorinsky clássico e dinâmico de Germano, isso é feito da mesma forma enquanto, para o modelo WALE, tem-se uma ligeira modificação. Adicionalmente, as constantes adimensionais envolvidas no cálculo de ν_{SGS} são determinadas de formas diferentes para cada modelo, conforme apresentado nas subseções seguintes.

Adicionalmente, para os modelos de Smagorinsky clássico e o modelo WALE são consideradas mais duas hipóteses. A primeira é de que assume-se uma separação espectral entre a banda de números de onda das estruturas turbilhonares resolvidas e a banda das estruturas submalha, de modo que não há sobreposição significativa entre essas duas bandas. E a segunda é de que o escoamento está em contínuo equilíbrio espectral, *i.e.* não há acúmulo de energia em nenhuma frequência e o formato do espectro de energia será constante ao longo do tempo. Estas hipóteses não são necessárias ao modelo dinâmico de Germano-Lilly^{1,2}.

3.3.1 Modelo Smagorinsky Clássico

Smagorinsky²⁷ desenvolveu este modelo para a modelagem de escoamentos meteorológicos. O modelo submalha se baseia na hipótese do equilíbrio local para as estruturas turbilhonares submalha junto a todas as outras hipóteses apresentadas para modelos submalha. Em específico, para este modelo, considera-se que a energia cinética turbulenta da banda resolvida para a não resolvida do espectro de energia cinética turbulenta, seja igual a potência específica de transformação viscosas de energia cinética turbulenta em energia térmica²⁶.

Assim, o modelo funcional de fechamento submalha Smagorinsky Clássico, é representado pela Eq. 3.14,

$$\nu_{SGS} = (C_s \Delta)^2 \sqrt{2\bar{S}_{ij}\bar{S}_{ij}}, \quad (3.14)$$

em que C_s é uma constante *ad-hoc* e adimensional, $\Delta = \sqrt[3]{\Delta_x \Delta_y \Delta_z}$ representa o tamanho característico das estruturas submalha e a norma do tensor taxa de deformação, $\sqrt{2\bar{S}_{ij}\bar{S}_{ij}}$, se relaciona ao tempo característico da banda não resolvida. Conforme destacado anteriormente, a Eq. 3.14, exemplifica, para este modelo submalha, que quantidades globais do escoamento relativas especificamente as estruturas resolvidas são usadas para a modelagem submalha.

A constante C_s não possui um valor universal, por exemplo, para a turbulência homogênea e isotrópica ela assume o valor de 0,17⁸¹. De acordo com Silveira Neto²⁶, essa constante pode assumir valores entre 0,065 até 0,25 (valor acima de um escoamento homogêneo e isotrópico). No presente trabalho é adotado o valor de $C_s = 0,09$ ajustado especificamente para o caso modelado junto a metodologia IMERSPEC.

Analisando novamente a Eq. 3.14, nota-se uma inconsistência física no modelo de Smagorinsky. Mesmo para um escoamento inteiramente laminar, onde todas as estruturas formadas são resolvida, o modelo apresentará $\nu_{SGS} \neq 0$ já que o campo de velocidade u_i apresentará variação espacial que causa $\bar{S}_{ij} \neq 0$. As hipóteses apresentadas no início dessa seção impedem que os modelos utilizados na presente tese sejam usados para modelar escoamentos em transição, eles performam melhor apenas para a modelagem de turbulência completa²⁸.

Outra limitação do modelo de Smagorinsky está relacionado ao seu comportamento próximo a paredes. No fenômeno físico, observa-se que em camadas do escoamento muito próximas de uma parede, a turbulência é amortecida pelos efeitos viscosos moleculares²⁶. Em contrapartida, pela Eq. 3.14, a viscosidade submalha não é nula próxima a paredes visto que nessa região há significativas derivadas dos componentes da velocidade implicando também que os valores da norma de $\bar{S}_{ij}$ não sejam desprezíveis. A partir da disparidade entre fenômeno físico e comportamento do modelo, infere-se que o comprimento característico desse modelo deve ser ajustado para que haja uma redução dessa grandeza próxima a paredes²⁸.

Esse ajuste é feito através de uma função de amortecimento empírica *ad-hoc* denominada como função de Van Driest²⁹, igual a:

$$f_w(y^+) = 1 - \exp\left(-\frac{y^+}{26}\right), \quad (3.15)$$

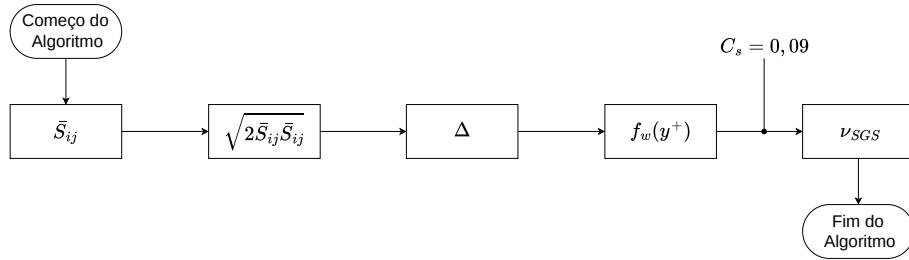
em que y^+ é a distância adimensional pontual da parede sólida para o primeiro ponto colocante da malha e é igual a $y^+ = u_\tau y / \nu$ no qual u_τ é a velocidade característica de regiões próximas às paredes, igual a $u_\tau = \sqrt{\tau_w / \rho}$ onde τ_w é a tensão viscosa na parede. Conforme mencionado, a

função de Van Driest tem o papel de ajuste do comprimento característico do modelo submalha. Dessa forma, a Eq. 3.14 é reescrita com esse ajuste *ad-hoc* e assume a forma:

$$\nu_{SGS} = (C_s \Delta f_w)^2 \sqrt{2\bar{S}_{ij}\bar{S}_{ij}}. \quad (3.16)$$

A implementação computacional desse modelo funcional submalha pode ser exemplificado através do esquema apresentado na Fig. 5 onde nota-se sua forma simples e direta de implementação.

Figura 5 – Esquema de implementação computacional de Smagorinsky clássico com a função de Van Driest.



Fonte: Próprio autor.

A fim de se evitar o uso de amortecimento de funções *ad-hoc* como a função de Van Driest pode-se usar um procedimento dinâmico conforme apresentado na subseção 3.3.3. Alternativamente, destaca-se que modelos de fechamento funcionais submalha podem ser desenvolvidos para ter o comportamento físico da turbulência próximo às paredes. Um exemplo disso é o modelo submalha WALE, do inglês *Wall-Adapting Local Eddy-viscosity*³⁰.

3.3.2 Modelo WALE

O modelo WALE também é um modelo funcional de fechamento submalha. Esse modelo foi desenvolvido como alternativa ao modelo clássico de Smagorinsky, compartilhando a mesma estrutura funcional baseada na hipótese de Boussinesq, Eq. 3.10. Assim, o modelo submalha apresenta a mesma estrutura de implementação mas altera a forma do invariante utilizado na Eq. 3.14. A ν_{SGS} é determinada de acordo com a Eq. 3.17,

$$\nu_{SGS} = (C_{WA} \Delta)^2 \frac{(\bar{\mathcal{G}}_{ij}^d \bar{\mathcal{G}}_{ij}^d)^{3/2}}{(\bar{S}_{ij} \bar{S}_{ij})^{5/2} + (\bar{\mathcal{G}}_{ij}^d \bar{\mathcal{G}}_{ij}^d)^{5/4}}, \quad (3.17)$$

em que C_{WA} é uma constante *ad-hoc* do modelo e $\bar{\mathcal{G}}_{ij}^d$ representa o tensor deviatórico do quadrado do gradiente da velocidade filtrada a ser discutido nessa subseção.

O operador alterado no modelo WALE, foi idealizado para seguir as seguintes características³⁰:

- o termo é um invariante para qualquer translação ou rotação de coordenadas do escoamento;

- pode ser facilmente determinado por qualquer tipo de malha computacional;
- é uma função das taxas de deformação e rotação do escoamento modelado;
- tende naturalmente a zero próximo a parede para que nenhuma função de amortecimento (subseção 3.3.1) ou procedimento dinâmico (subseção 3.3.3) sejam necessários.

Assim, o operador foi pensado a partir da definição de um tensor do quadrado do gradiente da velocidade filtrada. Esse tensor será usado na determinação de ν_{SGS} e por isso deve seguir a hipótese de Boussinesq, Eq. 3.10, isto é, o tensor deve ser deviatórico e simétrico. Com esses requisitos, chega-se a:

$$\bar{\mathcal{G}}_{ij}^d = \frac{1}{2}(\bar{g}_{ij}^2 + \bar{g}_{ji}^2) - \frac{1}{3}\delta_{ij}\bar{g}_{kk}^2, \quad (3.18)$$

em que $\bar{g}_{ij} = \frac{\partial \bar{u}_i}{\partial x_j}$ e como $\bar{g}_{ij}^2$ não é simétrico, é utilizado o artifício matemático $\frac{1}{2}(\bar{g}_{ij}^2 + \bar{g}_{ji}^2)$ que resulta em um tensor simétrico.

Considerando o tensor taxa de deformação filtrado, $\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right)$, e o tensor taxa de rotação filtrado, $\bar{W}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i} \right)$, pode-se reescrever a Eq. 3.18,

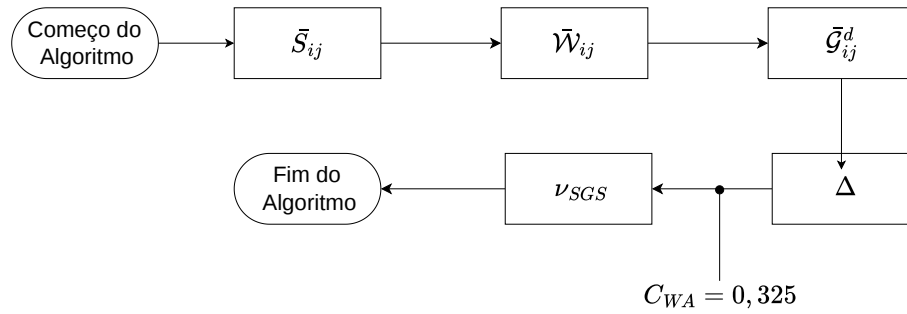
$$\bar{\mathcal{G}}_{ij}^d = \bar{S}_{ik}\bar{S}_{kj} + \bar{W}_{ik}\bar{W}_{kj} - \frac{1}{3}\delta_{ij}(\bar{S}_{mn}\bar{S}_{mn} - \bar{W}_{mn}\bar{W}_{mn}). \quad (3.19)$$

Segundo Nicoud; Ducros³⁰, usando o segundo invariante do tensor $\bar{\mathcal{G}}_{ij}^d$, $\bar{\mathcal{G}}_{ij}^d \bar{\mathcal{G}}_{ij}^d$, consegue-se detectar estruturas turbilhonares com significativa taxa de deformação, taxa de rotação ou mista³⁰. Adicionalmente, também se destaca que este invariante tende a zero em escoamentos laminares, tornando possível o uso do modelo submalha WALE na modelagem de escoamentos em transição, mais uma diferença em relação ao modelo de Smagorinsky.

Com a definição do invariante do tensor deviatórico do quadrado do gradiente da velocidade filtrada, o operador do modelo submalha foi formulado de modo que apresente comportamento assintótico $\mathcal{O}(y^{+3})$ próximo às paredes, tenha estabilidade numérica e não assuma valores igual a zero no seu denominador levando-se a uma impossibilidade matemática. O resultado do operador foi apresentado na Eq. 3.17.

Conforme ilustrado na Eq. 3.17, o modelo funcional de fechamento submalha depende de uma constante *ad-hoc*. Nicoud; Ducros³⁰ demonstraram a equivalência da constante $C_s = 0,17$ para um escoamento turbulento homogêneo e isotrópico para esse modelo implica em $C_{WA} = 0,55$. No entanto, para o escoamento modelado utilizou-se a constante de $C_{WA} = 0,325$, valor utilizado no *software* livre OpenFOAM. A Fig. 6 apresenta o esquema de implementação computacional do modelo funcional de fechamento submalha WALE em que nota-se sua forma similar a Fig. 5.

Figura 6 – Esquema de implementação computacional WALE.



Fonte: Próprio autor.

3.3.3 Modelo Dinâmico Germano-Lilly

Conforme mencionado a modelagem dinâmica pode solucionar alguma das deficiências da abordagem clássica da modelagem submalha explicitada na subseção anterior. Esse procedimento foi desenvolvido por Germano *et al.*¹ e proporciona um ajuste da constante de modelos submalha em cada ponto espacial e instante temporal da simulação computacional. Assim, este modelo se anula em regiões laminares de escoamentos modelados e também próximo às paredes (sem a necessidade de artifícios numéricos de amortecimento).

O procedimento dinâmico foi desenvolvido para o modelo clássico de Smagorinsky¹ e por isso é denominado usualmente como modelo de Smagorinsky dinâmico e essa implementação pode ser estendida para qualquer modelo de fechamento funcional submalha que utiliza uma constante arbitrária análoga a C_s na Eq. 3.14. No presente trabalho é usado o termo modelo dinâmico de Germano-Lilly a fim de dar destaque aos modificadores do modelo clássico de Smagorinsky.

Assim, neste modelo dinâmico, a constante *ad-hoc* C_s se torna uma função $C_d(x_i, t)$. O procedimento utiliza dois filtros com comprimentos característicos diferentes. O primeiro é denominado como filtro em nível de malha e é aplicado com a dimensão efetiva da malha computacional Δ (caso a filtragem seja explícita, do contrário a própria malha realiza a filtragem implicitamente). O segundo filtro é o filtro teste que é definido com seu comprimento característico, $\tilde{\Delta}$, como um múltiplo de Δ . No presente estudo, utiliza-se $\tilde{\Delta} = 2\Delta$, um valor ótimo para esse modelo²⁸.

A formulação dinâmica do modelo se baseia na identidade multiplicativa de Germano²⁸,

$$L_{ij} = T_{ij} - \tilde{\tau}_{ij}, \quad (3.20)$$

em que o símbolo $\sim$ representa as propriedades filtradas pelo filtro teste; τ_{ij} é o tensor global submalha de Boussinesq-Reynolds, igual a $\overline{u_i u_j} - \bar{u}_i \bar{u}_j$; T_{ij} é o tensor das tensões relativas ao segundo filtro ou tensor subteste e é igual a $\widetilde{\overline{u_i u_j}} - \tilde{\bar{u}}_i \tilde{\bar{u}}_j$; e L_{ij} é o tensor global de Leonard

definido como $\widetilde{\bar{u}_i \bar{u}_j} - \tilde{\bar{u}_i} \tilde{\bar{u}_j}$.

A Eq. 3.20 pode ser reescrita considerando as partes deviatóricas dos tensores envolvidos:

$$L_{ij} - \frac{1}{3}\delta_{ij}L_{ij} = T_{ij} - \frac{1}{3}\delta_{ij}T_{ij} - \tilde{\tau}_{ij} - \frac{1}{3}\delta_{ij}\tilde{\tau}_{ij}. \quad (3.21)$$

Enquanto o tensor L_{ij} pode ser computado diretamente do campo de velocidades resolvido é assumido que os tensores τ_{ij} e T_{ij} podem ser modelados pela mesma constante $C_d(x_i, t)$ utilizando a hipótese de Boussinesq. Assim, a modelagem dos tensores é igual a:

$$\tau_{ij} - \frac{1}{3}\delta_{ij}\tau_{ij} = -2\Delta^2 \|\bar{S}\| \bar{S}_{ij} C_d(x_i, t), \quad (3.22a)$$

$$T_{ij} - \frac{1}{3}\delta_{ij}T_{ij} = 2\tilde{\Delta}^2 \|\tilde{S}\| \tilde{S}_{ij} C_d(x_i, t). \quad (3.22b)$$

em que $\|\bar{S}\|$ é a norma do tensor taxa de deformação, igual a $\sqrt{2\bar{S}_{ij}\bar{S}_{ij}}$. Substituindo as aproximações das Eq. 3.22 na Eq. 3.21 e colocando em evidência $C_d(x_i, t)$, tem-se:

$$L_{ij} - \frac{1}{3}\delta_{ij}L_{ij} = C_d(x_i, t) \underbrace{[-2\tilde{\Delta}^2 \|\tilde{S}\| \tilde{S}_{ij} + 2\Delta^2 \|\bar{S}\| \bar{S}_{ij}]}_{=M_{ij}}, \quad (3.23)$$

onde o termo entre colchetes é condensado no tensor M_{ij} .

Assim, a determinação da função $C_d(x_i, t)$ é realizada a partir dos resíduos da Eq. 3.23, *i.e.*,

$$Res = L_{ij} - \frac{1}{3}\delta_{ij}L_{ij} - C_d(x_i, t) M_{ij}, \quad (3.24)$$

onde Res são os resíduos da Eq. 3.23. Destaca-se que a definição das Eqs. 3.24 consiste em seis componentes independentes para o tensor simétrico τ_{ij}^{SGS} , o que em teoria torna possível definir seis funções C_d , possibilitando a definição de um modelo de fechamento submalha tensorial²⁸. Para manter C_d como uma função escalar, Germano *et al.*¹ propuseram reduzir o sistema contraindo a identidade de Germano com o tensor taxa de deformação filtrado $\bar{S}_{ij}$:

$$C_d = \frac{L_{ij}\bar{S}_{ij}}{M_{ij}\bar{S}_{ij}}. \quad (3.25)$$

A Eq. 3.25, apesar de funcional, apresenta um problema significativo de indeterminação quando $\bar{S}_{ij} \rightarrow 0$, tornando C_d indefinido nessas regiões²⁸. Alternativamente, Lilly² propôs determinar $C_d(\mathbf{x}, t)$ pelo método dos mínimos quadrados, minimizando o resíduo quadrático:

$$\frac{\partial Res^2}{\partial C_d} = 0, \quad (3.26)$$

resultando em:

$$C_d(\mathbf{x}, t) = \frac{L_{ij}M_{ij}}{M_{kl}M_{kl}}. \quad (3.27)$$

Assim, a formulação desse modelo é denominada Germano-Lilly, em reconhecimento à contribuição dos dois pesquisadores. Analisando o procedimento de determinação de $C_d(\mathbf{x}, t)$, percebem-se as seguintes propriedades:

- C_d pode assumir valores negativos, modelando assim, o efeito *backscatter*;
- a Eq. 3.27 não é limitada matematicamente por nenhum valor e o denominador pode ter valores nulos implicando em indeterminação matemática severa.

Considerando que essas propriedades de C_d comprometem a estabilidade numérica-computacional de simulações, violando assim uma das restrições numéricas fundamentais de modelos submalha discutidos na subseção 3.2.3, são necessário artifícios computacionais para evitar a divergência numérica da implementação do modelo.

Assim, no presente trabalho, duas restrições são impostas: a média de C_d nas direções homogêneas do escoamento e a clipagem de valores de ν_{SGS} e C_d . Em relação ao processo de média de C_d nas direções homogêneas dos escoamentos modelados, salienta-se que esse processo é feito separadamente para o numerador e denominador da Eq. 3.27, assumindo a forma:

$$C_d(x_i, t) = \frac{\langle L_{ij}M_{ij} \rangle}{\langle M_{kl}M_{kl} \rangle}. \quad (3.28)$$

Em relação a clipagem, esta é realizada considerando os valores: $\nu_{SGS} \geq 0$ e $C_d \leq C_{d,max} = 0.3$ em que $C_{d,max}$ é definido na mesma ordem de grandeza do valor máximo da constante de Smagorinsky. Destaca-se que os valores que não atendem as condições de clipagem assumem os, respectivos, valores limitantes. Nota-se que o processo de clipagem elimina a possibilidade de modelagem do efeito *backscatter* do modelo submalha, entretanto, garante-se a estabilidade numérica da simulação²⁸.

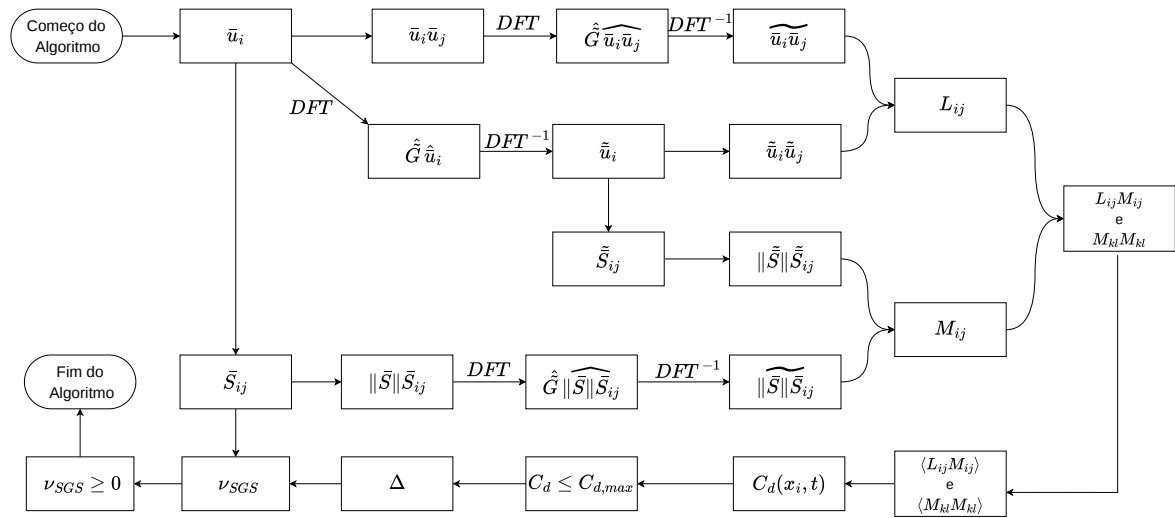
Destaca-se que todo o procedimento até aqui foi feito apenas para determinar a função escalar $C_d(x_i, t)$, a implementação do modelo submalha é feito ainda de uma forma análoga a Eq. 3.14, ou seja,

$$\nu_{SGS} = C_d(x_i, t)\Delta^2\|\bar{S}\|. \quad (3.29)$$

Salientam-se, de acordo com Sagaut²⁸, os requisitos mínimos para esse modelo submalha: $\tilde{\Delta} > \Delta$, em que esses comprimentos característicos devem estar na região inercial do espectro de energia cinética turbulenta; e os filtros utilizados na modelagem dinâmica devem ser de

mesma natureza (*self-similar*), condição necessária para que a hipótese de similaridade de escala implícita no procedimento de Germano-Lilly seja válida²⁸. A implementação computacional desse modelo funcional submalha pode ser exemplificado através do esquema apresentado na Fig. 7 que é análoga a Fig. 5. Nota-se que o processo é significativamente mais complexo devido ao procedimento dinâmico ao mesmo tempo que a filtragem teste é feita de forma simplificada no espaço espectral através do uso da Transformada Discreta de Fourier (DFT), do inglês *Discrete Fourier Transform*, e sua inversa (DFT^{-1}).

Figura 7 – Esquema de implementação computacional do modelo dinâmico de Germano-Lilly^{1,2}.



Fonte: Próprio autor.

Nesse capítulo foram apresentadas a formulação matemática utilizada no presente trabalho para a modelagem de escoamentos laminares ou turbulentos. Foi explorada as equações utilizadas para modelar o fenômeno físico de interesse laminar até a possível abordagem LES implementada como forma de modelagem de escoamentos turbulentos com os três modelos SGS explorados no presente trabalho. Estabelecida a formulação matemática para a representação do fenômeno físico, o próximo capítulo apresentará a metodologia IMERSPEC e os seus aspectos numérico-computacionais.

4 METODOLOGIA IMERSPEC

Neste capítulo é apresentada a metodologia híbrida do Método Pseudoespectral de Fourier com o Método da Fronteira Imersa denominada IMERSPEC. Para maior compreensão da metodologia IMERSPEC o capítulo é subdividido em: Método da Fronteira Imersa (MFI), Método Pseudoespectral de Fourier (MPF) e os aspectos numérico-computacionais da metodologia. Essa última seção é necessária para apresentar mecanismos necessários à implementação computacional do MPF, suas formas de discretização e um detalhamento do método de avanço temporal empregado no código computacional.

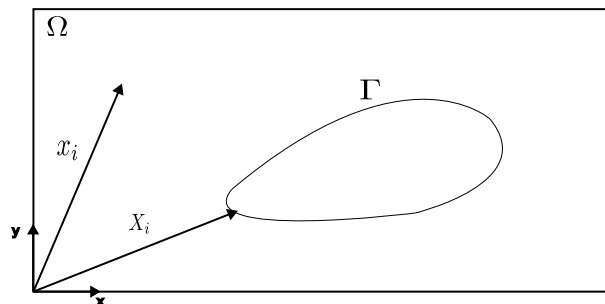
No presente capítulo, utiliza-se as Eq. 3.6 e 3.13 como ponto de partida por essa formulação, principalmente para o MPF, oferecer mais desafios do que a sua versão sem modelagem submalha. Reforça-se, porém, que a sua versão, sem processo de filtragem, pode ser utilizada com mais facilidade ainda.

4.1 Método da Fronteira Imersa

Na abordagem IMERSPEC, utiliza-se o método da fronteira imersa (MFI), no qual faz-se uso de dois domínios de cálculo: um euleriano (Ω) e um lagrangiano (Γ). Em Ω são resolvidas computacionalmente as Eq. 3.6 e 3.13 através do método pseudoespectral de Fourier (MPF) e em Γ , as condições de contorno do escoamento modelado são impostas através de interfaces. Dependendo do escoamento modelado, Γ pode representar uma interface de um sólido fixo com um fluido ou a interface entre dois tipos de fluidos.

Na Figura 8 são apresentados os dois domínios de cálculo sob uma perspectiva geral. Destaca-se que as variáveis maiúsculas representam as variáveis lagrangianas e as minúsculas, as variáveis eulerianas. As interfaces representadas em Γ se tornam linhas e superfícies, respectivamente, em modelagens bidimensionais e tridimensionais. Com base nisso, a Fig. 8 ilustra uma abordagem bidimensional.

Figura 8 – Esquema dos domínios Ω e Γ na modelagem de um escoamento qualquer.



Fonte: Próprio autor.

4.1.1 Múltipla Imposição com Domínios Ω e Γ Coincidentes

Conforme já explicitado, o MFI opera com dois domínios de cálculo. Sendo assim, deve haver comunicação entre os dois domínios. A troca de informações entre Γ e Ω é feita através do termo fonte $\bar{f}_i$ da Eq. 3.13 e é definido como:

$$\bar{f}_i(x_i, t) = \begin{cases} \bar{F}(X_i, t) & \text{se } x_i = X_i \\ 0 & \text{se } x_i \neq X_i \end{cases}, \quad (4.1)$$

em que $\bar{F}(X_i, t)$ representa o termo correspondente em Γ . Destaca-se que acontece a transferência de informação entre os domínios pela Eq. 4.1 apenas quando há coincidência de pontos e, portanto, o termo f_i é descontínuo.

Em casos onde não há coincidência de pontos entre Ω e Γ , distribui-se o termo fonte $\bar{F}_i(X_i, t)$ em relação a sua vizinhança através de uma função distribuição de força. Esse caso especial será apresentado na seção 4.1.2.

Em Γ ocorre o cálculo de $F_i(X_i, t)$, o IMERSPEC utiliza o método de imposição direta, em inglês *Direct Forcing Method*, (DFM) desenvolvido por Uhlmann⁴³. O termo fonte $\bar{f}_i$, análogo ao $\bar{F}_i$, é obtido isolando-o na Eq. 3.13 e obtendo-se:

$$\bar{f}_i(x_i, t) = \frac{\partial \bar{u}_i}{\partial t} + \frac{1}{2} \left[\frac{\partial(\bar{u}_i \bar{u}_j)}{\partial x_j} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} \right] + \frac{\partial \bar{p}^m}{\partial x_j} - \frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right]. \quad (4.2)$$

A Equação 4.2 é aplicada em Ω e uma das considerações que foram usadas em seu desenvolvimento foi a hipótese do contínuo. Conforme já mencionado, sabe-se que o domínio Γ está contido em Ω , portanto as hipóteses válidas à Eq. 4.2 também são aplicáveis para Γ . Assim, a Eq. 4.2 pode ser reescrita no domínio lagrangiano:

$$\bar{F}_i(X_i, t) = \frac{\partial \bar{U}_i}{\partial t} + \frac{1}{2} \left[\frac{\partial(\bar{U}_i \bar{U}_j)}{\partial X_j} + \bar{U}_j \frac{\partial \bar{U}_i}{\partial X_j} \right] + \frac{\partial \bar{P}^m}{\partial X_j} - \frac{\partial}{\partial X_j} \left[\nu_{eff} \left(\frac{\partial \bar{U}_i}{\partial X_j} + \frac{\partial \bar{U}_j}{\partial X_i} \right) \right], \quad (4.3)$$

em que as variáveis em letras maiúsculas representam essas informações em Γ .

Para fins didáticos do MFI, discretiza-se a derivada temporal da Eq. 4.3 com um esquema de Euler explícito e condensa-se os demais termos da expressão em uma variável denominada por $\overline{RHS}_i^t$, do inglês *Right Hand Side* chegando-se a Eq. 4.4,

$$\bar{F}_i(x_i, t) = \frac{\bar{U}_i^{t+\Delta t} - \bar{U}_i^t}{\Delta t} + \overline{RHS}_i^t, \quad (4.4)$$

onde Δt é o incremento temporal.

O DFM consiste na adição e subtração de um parâmetro temporário, $\bar{U}_i^*$, na Eq. 4.4 resultando-se em:

$$\bar{F}_i(x_i, t) = \frac{\bar{U}_i^{t+\Delta t} + \bar{U}_i^* - \bar{U}_i^* - \bar{U}_i^t}{\Delta t} + \overline{RHS}_i^t. \quad (4.5)$$

Com base na inserção e retirada de $\bar{U}_i^*$ pode-se subdividir a Eq. 4.5 em duas expressões:

$$\frac{\bar{U}_i^* - \bar{U}_i^t}{\Delta t} + \overline{RHS}_i^t = 0, \quad (4.6)$$

$$\bar{F}_i(X_i, t) = \frac{\bar{U}_i^{t+\Delta t} - \bar{U}_i^*}{\Delta t}. \quad (4.7)$$

Destaca-se novamente que Γ está contido em Ω e, portanto, tudo que foi desenvolvido no domínio lagrangiano, principalmente as Eq. 4.6 e 4.7, também se torna aplicável no domínio euleriano. Sendo assim, pode-se utilizar uma equação análoga a Eq. 4.6 no domínio euleriano para se estimar um parâmetro temporário $\bar{u}_i^*$. A Equação 4.8,

$$\frac{\bar{u}_i^* - \bar{u}_i^t}{\Delta t} + \overline{rhs}_i^t = 0 \rightarrow \bar{u}_i^* = \bar{u}_i^t - \Delta t \overline{rhs}_i^t, \quad (4.8)$$

apresenta a expressão análoga à Eq. 4.6 e a sua respectiva forma quando isola-se o parâmetro temporário euleriano.

Após a determinação de $\bar{u}_i^*$ em Ω , ocorre a transferência dessa informação de Ω para a interface imersa, Γ , de maneira similar a apresentada pela Eq. 4.1, *i.e.*,

$$\bar{u}_i^* = \begin{cases} \bar{U}_i^* & \text{se } x_i = X_i \\ 0 & \text{se } x_i \neq X_i \end{cases}. \quad (4.9)$$

Com o parâmetro temporário em Γ , $\bar{U}_i^*$, pode-se determinar $\bar{F}_i(X_i, t)$ com o auxílio da Eq. 4.7, pois o termo $\bar{U}_i^{t+\Delta t}$ representa a velocidade desejada na interface imersa, $\bar{U}_i^{IM}$. Essa velocidade $\bar{U}_i^{IM}$ é conhecida para modelagens onde é possível determinar o valor da velocidade na interface previamente, como por exemplo na modelagem de um escoamento sobre uma placa plana em que na superfície $\bar{U}_i^{IM} = 0 \text{ m/s}$, ou é obtida por algum modelo que rege a interface imersa em modelagens do tipo interação fluido-fluido ou fluido-estrutura.

Com base que $\bar{U}_i^{t+\Delta t} = \bar{U}_i^{IM}$ e por razões didáticas na explicação do DFM, reescreve-se a Eq. 4.7 com essa atualização:

$$\bar{F}_i(X_i, t) = \frac{\bar{U}_i^{IM} - \bar{U}_i^*}{\Delta t}. \quad (4.10)$$

Definido o termo fonte lagrangiano, $\bar{F}_i(X_i, t)$, essa informação é passada ao domínio euleriano de acordo com a Eq. 4.1 e é utilizada para a determinação de $\bar{u}_i^{t+\Delta t}$ através da Eq. 4.11,

$$\bar{f}_i(x_i, t) = \frac{\bar{u}_i^{t+\Delta t} - \bar{u}_i^*}{\Delta t} \rightarrow \bar{u}_i^{t+\Delta t} = \bar{u}_i^* + \Delta t \bar{f}_i(x_i, t), \quad (4.11)$$

em que a Eq. 4.11 é análoga à Eq. 4.7 e foi aplicada em Ω sob a hipótese do contínuo.

Comparando-se o DFM com o método preditor-corretor, o parâmetro temporário u_i^* pode ser compreendido como um campo de velocidade predita ou estimada, *i.e.*, é a solução das equações de *momentum* linear sem a influência do termo fonte⁴². Após a determinação de $\bar{f}_i(x_i, t)$, corrige-se então $\bar{u}_i^*$ obtendo-se o campo $\bar{u}_i^{t+\Delta t}$.

Apesar da utilização de um esquema de Euler explícito na explicação do DFM, esse não é o método utilizado no IMERSPEC. O avanço temporal é feito através do Método de Runge-Kutta de Quarta Ordem com 6 Passos (RK46) proposto por Allampalli *et al.*⁸³ a ser explicado ao longo deste capítulo.

A velocidade do escoamento de fluido sobre a interface imersa deveria ser igual a $\bar{U}_i^{IM}$ segundo o desenvolvimento do DFM, porém, isso não ocorre devido aos processos de discretização temporal e espacial envolvidos na solução computacional e, principalmente, pelo processo de satisfazer a equação de balanço de massa (Eq. 3.6). Para aumentar a exatidão do DFM, Wang; Fan; Luo⁴⁴ propuseram o Método de Múltiplas Imposições Diretas, em inglês *Multi Direct Forcing Method*, (MDFM) que representa um aperfeiçoamento do DFM.

A ideia por trás do MDFM é tornar o DFM um processo iterativo até que atinja-se determinado critério pré imposto resultando em $\bar{u}_i^{t+\Delta t} \rightarrow \bar{U}_i^{IM}$. Durante a primeira iteração do MDFM, esse método apresenta exatamente os processos descritos para o DFM a exceção da Eq. 4.11 que ao invés de determinar $\bar{u}_i^{t+\Delta t}$ define $\bar{u}_i^{it}$, *it* representa o número de iterações do MDFM.

O campo de velocidades $\bar{u}_i^{it}$ é então utilizado como parâmetro temporário $\bar{u}_i^*$ na nova iteração. Com $\bar{u}_i^*$ transfere-se a informação para o domínio lagrangiano onde há coincidência de pontos, conforme a Eq. 4.9, e então, obtêm-se um novo campo de força $\bar{F}_i^{it}$.

Com o novo campo de força $\bar{F}_i^{it}$ determinado, transfere-se essa informação a Ω , conforme a Eq. 4.1, e determina-se $\bar{u}_i^{it+1}$ através da Eq. 4.12,

$$\bar{u}_i^{it+1} = \bar{u}_i^{it} + \Delta t \bar{f}_i^{it}(x_i, t), \quad (4.12)$$

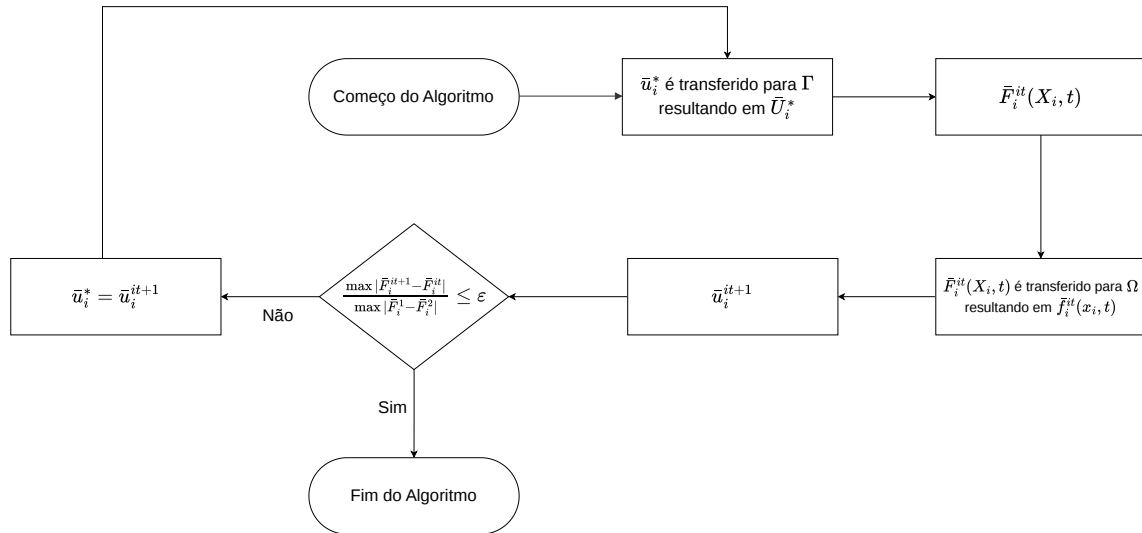
que se aproxima ainda mais de $\bar{U}_i^{IF}$ do que $\bar{u}_i^{it}$. Esse processo iterativo é realizado *NL* vezes. Dentro do IMERSPEC utiliza-se, como critério de parada, a máxima diferença das forças lagrangianas entre duas iterações consecutivas adimensionalizadas pela diferença entre as forças calculadas nas duas primeiras iterações²³, *i.e.*,

$$\frac{\max |\bar{F}_i^{it+1} - \bar{F}_i^{it}|}{\max |\bar{F}_i^1 - \bar{F}_i^2|} \leq \varepsilon. \quad (4.13)$$

Esse critério de parada é imposto ao MDFM pois não se sabe qual o número de iterações necessário para se obter um resultado computacional fiel ao fenômeno físico modelado²³. No início de cada simulação ou quando o escoamento modelado esta em sua fase transiente são necessárias mais iterações para uma modelagem satisfatória do que quando o escoamento entra em regime permanente. Utilizando o critério da Eq. 4.13, o MDFM fica dependente da máxima diferença entre as velocidades lagrangianas em iterações consecutivas, tendo mais iterações durante o seu regime transiente e, após atingir o regime permanente, diminuindo a quantidade de iterações por ciclo de forma otimizada.

A Figura 9 apresenta um fluxograma do algoritmo MFI utilizando o MDFM quando há coincidência de pontos entre Ω e Γ . Salienta-se que o algoritmo apresentado é apenas uma porção do IMERSPEC que utiliza o MFI.

Figura 9 – Diagrama de blocos do MDFM/MFI para pontos de Γ coincidentes.



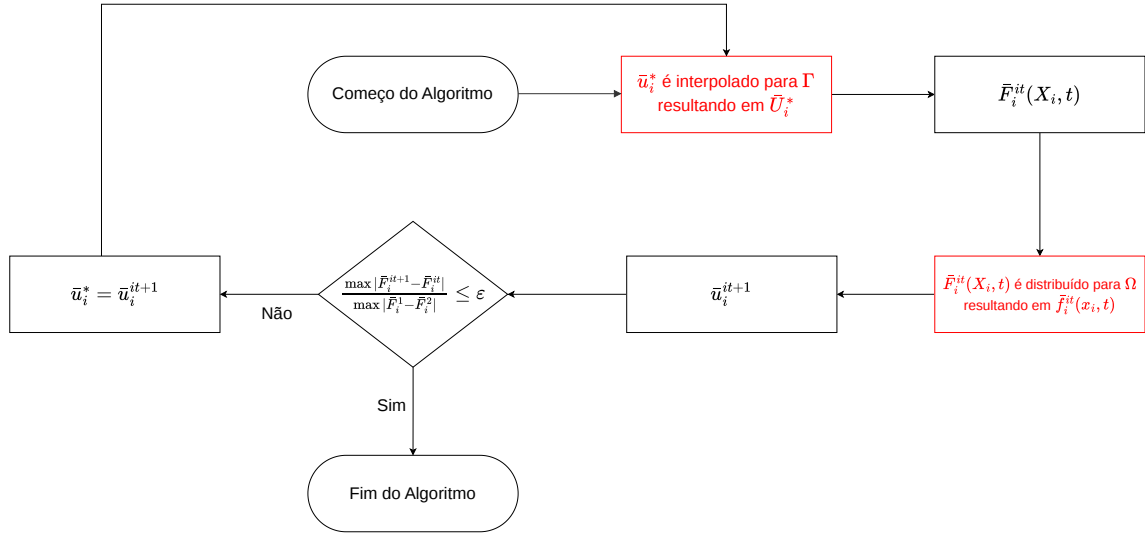
Fonte: Próprio autor.

4.1.2 Múltipla Imposição com Domínios Ω e Γ Não Coincidentes

Quando uma geometria complexa é imersa dentro de Ω não há como impor condições de contorno diretamente nos pontos correspondentes visto que explicitamente tais pontos não existem. Assim, torna-se necessário novas formas de transferência de informação entre Ω e Γ denominadas por interpolação e distribuição.

Em linhas gerais, o procedimento do MDFM/MFI para pontos coincidentes se mantém para a nova circunstância. O que se difere é como as informações são transferidas entre os domínios conforme destacado em vermelho na Fig. 10.

O processo de interpolação para determinar, $\bar{U}_i^*$, é apresentado na Eq. 4.14,

Figura 10 – Diagrama de blocos MDFM/MFI para pontos de Γ não coincidentes.

Fonte: Próprio autor.

$$\bar{U}_i^* = \sum_{\Omega} \bar{u}_i^* \delta_h(x_i - X_i) h^3, \quad (4.14)$$

enquanto o processo de distribuição é exposto na Eq. 4.15,

$$\bar{f}_i^{it} = \sum_{\Gamma} \bar{F}_i^{it} \delta_h(x_i - X_i) \Delta s^3, \quad (4.15)$$

em que δ_h é a função interpoladora/distribuidora, considerada ambas iguais no presente trabalho; Δs é o incremento lagrangiano geral; e h representa o incremento euleriano geral.

A função distribuição/interpolação, conforme apresentado nas Eq. 4.14 e 4.15, tem um papel significativo na comunicação entre as malhas eulerianas e lagrangianas através do espalhamentos dos termos fonte para o escoamento modelado e na interpolação dvelocidades eulerianos. Há três critérios principais na escolha de uma função distribuição/interpolação: invariância translacional, exatidão da interpolação e eficiência computacional⁴⁸.

É desejável que as operações de interpolação ou distribuição sejam independentes dos marcadores lagrangianos relativos aos eulerianos, daí a necessidade do critério de invariância translacional⁴⁸. Há funções que podem parecer boas candidatas para os processos de interpolação/distribuição em termos de exata invariância translacional da malha, por exemplo, a função $\text{sinc}(x)$. Entretanto essa função tem suporte ilimitado e exata invariância translacional da malha é inconsistente com a necessidade de suporte compacto³⁹, *i.e.*, a função interpolação/distribuição deve ser definida apenas próximo ao ponto de interesse x_i .

A fim de discutir os demais critérios, considera-se a forma tridimensional da função interpolação/distribuição, δ_h , definida através da Eq. 4.16,

$$\delta_h(x_i) = \frac{1}{h^3} \phi\left(\frac{x_1}{h}\right) \phi\left(\frac{x_2}{h}\right) \phi\left(\frac{x_3}{h}\right), \quad (4.16)$$

onde x_1 , x_2 , e x_3 são os componentes cartesianos de x_i e ϕ representa a função interpolação/distribuição aplicada em uma única direção.

A exatidão ansiada na escolha da função de interpolação/distribuição é justificável visto que o IMERSPEC é a junção do MPF, um método de alta ordem, com o MFI, um método de baixa ordem. Sendo assim, deseja-se uma máxima exatidão no MFI para que o resultado final da solução numérico-computacional mantenha ou se aproxime o máximo possível da acurácia do MPF.

Em relação ainda a exatidão, Peskin³⁹ define que: ϕ deve ser inteiramente contínua a fim de evitar variações bruscas na velocidade interpolada; e que a função deve satisfazer n momentos em que n corresponde a ordem da função interpolação/distribuição. A medida que se tem uma função que satisfaz mais n momentos, o seu suporte compacto é maior, *i.e.*, a quantidade de pontos ao redor do ponto de interesse necessários para definir ϕ em cada uma das três direções se torna maior, o que torna pior a eficiência computacional do método.

Analisando a Eq. 4.16, nota-se que para cada ponto são feitas operações nas três direções do escoamento tridimensional. Dessa forma, para que haja eficiência computacional no método a função tem de ter um domínio compacto.

Com base nos três critérios discutidos, a função de interpolação/distribuição escolhida é a função cúbica definida como:

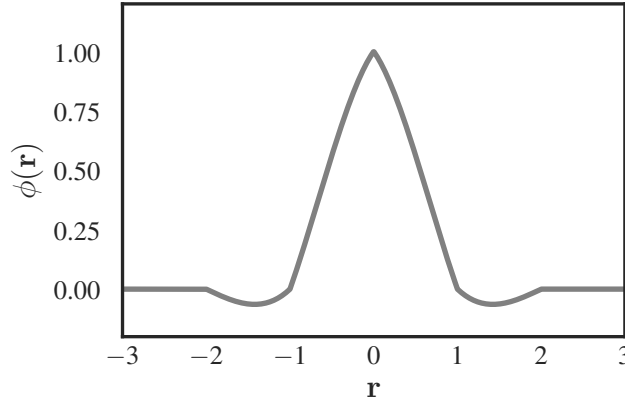
$$\phi_4(r) = \begin{cases} 1 - \frac{1}{2}|r| - |r|^2 + \frac{1}{2}|r|^3, & 0 \leq |r| < 1, \\ 1 - \frac{11}{6}|r| + |r|^2 - \frac{1}{6}|r|^3, & 1 \leq |r| < 2, \\ 0, & 2 \leq |r|, \end{cases} \quad (4.17)$$

em que r é a coordenada geral da função, o sobrescrito 4 representa o suporte necessário a função e ϕ_4 é ilustrada na Fig. 11.

A função cúbica não satisfaz inteiramente os momentos propostos por Peskin³⁹ entretanto, Mariano⁴², prova que sua exatidão/eficiência computacional é superior a quaisquer outras funções que seguem tais axiomas. Essa informação ainda se mantém para o IMERSPEC nos casos modelados no presente trabalho mesmo quando comparadas a funções mais recentes como a função de 6 pontos otimizada por Bao; Kaye; Peskin⁴⁸.

4.1.3 Imposição Espectral Direta com Domínios Ω e Γ Não Coincidentes

Conforme já mencionado o IMERSPEC é o resultado do acoplamento entre duas metodologias. Um método de baixa ordem, MFI, e outro de alta ordem, o MPF. A maior limitação

Figura 11 – Gráfico da função $\phi_4(r)$.

Fonte: Próprio autor.

do MFI/MFDM ocorre quando as malhas Ω e Γ não são coincidentes e a exatidão do método depende de funções de interpolação/distribuição às quais devem possuir um suporte compacto o que implica em perda de exatidão.

Neste contexto descrito e a partir do trabalho de Chen; Peskin⁵⁰ em que também é proposta uma fronteira imersa espectral para a modelagem de uma condição de contorno móvel e flexível. Se idealizou adaptar o método de imposição direta utilizando a transformada de Fourier Não-Uniforme, NUFFT, (do inglês, *Non-Uniform Fourier Transform*) e assim propor uma nova metodologia denominada por Método de Imposição Espectral Direta, MIED.

A nova metodologia segue conforme descrito anteriormente para o Método de Imposição direta, entretanto as Eq. 4.14 e 4.15 são reescritas para que a transferência de informação entre os domínios aconteçam através da NUFFT. Sendo assim, o processo de interpolação é definido através da Eq. 4.18,

$$U_i^* \approx \sum_{\Omega} \hat{u}(k_i) \exp(i k_i X_j), \quad (4.18)$$

e o processo de distribuição é definido a partir da Eq. 4.19,

$$f_i^{it} \approx \prod_{i=1}^3 \frac{1}{L_i} \sum_{\Gamma} \omega_j F_j \exp(-i k_i X_j), \quad (4.19)$$

em que $i = \sqrt{-1}$, $j = 1, \dots, N_b$ se refere ao número total de pontos lagrangianos N_b , ω_j representa pesos de quadratura e o símbolo $\hat{\cdot}$ representa as respectivas variáveis no domínio complexo. Para a implementação do MIED, considerou-se como pesos de quadratura a área de cada triângulo do arquivo de malha lagrangiana no formato STL. Destaca-se que para a realização das transformadas de Fourier descritas através das Eq. 4.18 e 4.19 foram utilizadas o pacote

*cuFINUFFT*⁸⁴ que utiliza NUFFT utilizando GPUs. Nesse pacote o fator de superamostragem utilizado foi definido internamente no algoritmo a partir da escolha da tolerância igual a 10^{-12} .

Analisando as Eq. 4.18 e 4.19, assumindo que as fronteiras imersas definidas em Γ apresentem malhas não uniformes e que a malha seja uniforme em Ω é razoável afirmar que na interpolação, Eq. 4.18, é feita uma transformada de Fourier inversa de uma distribuição uniforme para uma distribuição não-uniforme N_b enquanto na distribuição, Eq. 4.19, é feita uma transformada de Fourier de uma distribuição não uniforme N_b para uma distribuição uniforme em Ω . Segundo a denominação interna do pacote *cuFINUFFT*, respectivamente, essas transformadas são do tipo 2 e do tipo 1.

De acordo com Chen; Peskin⁵⁰, a transformada de Fourier presente nas operações de transferência entre Ω e Γ impõe exata invariância translacional, entretanto, o seu suporte já não é mais compacto visto que essas operações, transformadas, são operações globais. Destaca-se que Chen; Peskin⁵⁰ provam que o uso de transformadas de Fourier equivale a uma função de interpolação/distribuição na forma de $\frac{1}{h} \text{sinc}\left(\frac{\pi x}{h}\right)$ o que prova o argumento de que o suporte é não compacto.

Os processo de interpolação e distribuição são definido através de aproximações conforme visto nas Eq. 4.18 e 4.19. Essas aproximações são sensíveis em relação ao quão não uniforme é a distribuição de Γ com N_b pontos e, para a Eq. 4.19, o método de cálculo de ω_j . Assim, por se tratar de aproximações, um método iterativo pode aumentar ruído entre sucessivas aproximações durante iterações e prejudicar a exatidão do método. Por esse motivo, o uso das NUFFT no MFI é restrito ao MIED.

Dentre as características gerais do MFI, pode-se citar a força total aplicada em cada ponto lagrangiano, $\bar{F}_i^{Total}(X_i, t)$. Essa força é a soma das forças, $\bar{F}_i^{it}(X_i, t)$, em NL vezes atuando em cada ponto lagrangiano⁴⁴,

$$\bar{F}_i^{Total}(X_i, t) = \sum_{it=1}^{NL} \bar{F}_i^{it}(X_i, t), \quad (4.20)$$

em que para o MIED considera-se $NL = 1$. Destaca-se que $\bar{F}_i^{Total}(X_i, t)$ refere-se a força lagrangiana sobre o escoamento e deve ser exatamente igual a força que o escoamento aplica sobre as partículas lagrangianas, porém, com sentido oposto.

Nos trabalhos de Uhlmann⁴³ e Wang; Fan; Luo⁴⁴ o campo de pressão é calculado para que se satisfaça a Eq. 3.6 entretanto, dentro do IMERSPEC, isso não é necessário pois o Método pseudoespectral de Fourier utiliza uma projeção do campo de velocidades que já garante o balanço de massa intrinsecamente^{42,23,14}. Esse método será apresentado na seção 5.1.1 onde tal fato será melhor explorado.

4.2 Método Pseudoespectral de Fourier

4.2.1 Metodologia Pseudoespectral

Métodos espectrais de Fourier se baseiam na transformação das variáveis do espaço físico para o espaço espectral ou de Fourier. A transformação de Fourier é feita do espaço físico para o espectral e é apresentada na Eq. 4.21⁸⁰,

$$\hat{\varsigma}(k_i, t) = \int_{-\infty}^{\infty} \varsigma(x_i, t) e^{-i2\pi k_i x_i} dx_i, \quad (4.21)$$

enquanto a transformação inversa de Fourier é feita do espaço espectral para o espaço físico e é explicitada na Eq. 4.22⁸⁰,

$$\varsigma(x_i, t) = \frac{1}{N} \int_{-\infty}^{\infty} \hat{\varsigma}(k_i, t) e^{i2\pi k_i x_i} dk_i, \quad (4.22)$$

onde k_i são números de onda em $1/m$, N é o número de pontos discretos, ς representa uma variável do escoamento modelado no domínio físico, enquanto a mesma informação no domínio espectral corresponde a $\hat{\varsigma}$.

De acordo com Briggs; Henson⁸⁰, a utilidade da transformada de Fourier, como qualquer outra transformada, vem da possibilidade de que uma formulação no domínio físico tenha uma forma mais simples em outro domínio, como o espectral. O emprego da transformada de Fourier possibilita, por exemplo, que derivadas se tornem produtos de vetores no domínio de Fourier gerando simplicidade e acurácia na resolução das Eq. 3.6 e 3.13. Além disso, normalmente uma equação diferencial parcial é reduzida à uma equação diferencial ordinária.

Apesar das vantagens do Método Espectral de Fourier, existe uma grande desvantagem nessa abordagem: o produto de convolução entre duas funções transformadas. Dadas duas variáveis no domínio físico, ς e ψ , o produto entre elas no domínio físico é igual a $\sigma = \varsigma\psi$, entretanto, essa expressão não se mantém no domínio espectral. O produto entre duas variáveis espectrais no domínio de Fourier é denominado produto de convolução e é definido através da Eq. 4.23,

$$\hat{\sigma}(k_i, t) = [\hat{\varsigma} * \hat{\psi}](k_i, t) = \int_{k_i=r_i+s_i} \hat{\varsigma}(r_i) \hat{\psi}(k_i - r_i) dr_i, \quad (4.23)$$

em que r_i e s_i são os números de onda de, respectivamente, $\hat{\varsigma}$ e $\hat{\psi}$.

O produto de convolução é uma propriedade das transformadas de Fourier e é uma desvantagem para métodos espectrais pois possui um alto custo computacional para a sua implementação, assim na abordagem IMERSPEC utiliza-se o Método Pseudoespectral de Fourier (MPF) em que o produto entre duas variáveis, ς e ψ por exemplo, é feito no domínio físico e

então o resultado dessa operação é submetida à transformada de Fourier. A Equação 4.24 ilustra os passos de MPF quando há um produto entre duas variáveis quaisquer,

$$\hat{\sigma} = \widehat{\zeta\psi}. \quad (4.24)$$

Dessa forma, o MPF concilia as vantagens de operações matemáticas no domínio espectral eliminando aquelas operações que sejam desvantajosas, sempre dando importância à acurácia dos resultados e velocidade de processamento.

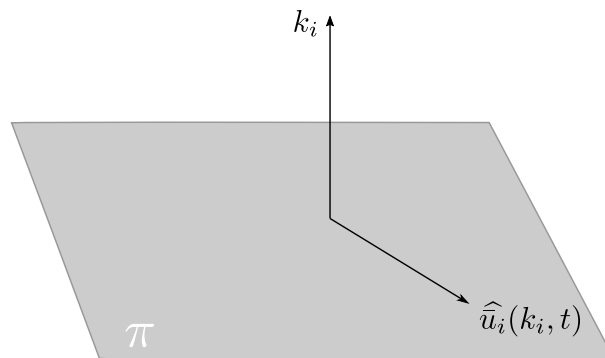
4.2.2 Procedimento para o MPF

Após a apresentação do MPF e suas principais características, busca-se aplicar esse método às Eq. 3.6 e 3.13. Aplicando a transformada de Fourier e suas propriedades sobre a Eq. 3.6, tem-se a Eq. 4.25,

$$ik_i \widehat{u}_i = 0. \quad (4.25)$$

Analisando a Eq. 4.25, nota-se que a equação da continuidade no domínio espectral se torna um produto escalar entre o vetor da velocidade espectral, $\widehat{u}_i$, e o vetor número de onda, k_i . Esse produto escalar resulta em valor nulo para escoamentos incompressíveis, logo, afirma-se, desprezando a solução trivial, que o vetor k_i é perpendicular ao vetor $\widehat{u}_i$. A fim de explicitar o MPF, define-se um plano hipotético que contém ao vetor $\widehat{u}_i$ denominado plano π . A Figura 12 esquematiza a posição de π em relação aos vetores k_i e $\widehat{u}_i$.

Figura 12 – Esquema do plano π com $\widehat{u}_i$ e k_i .



Fonte: Próprio autor.

A transformada de Fourier junto a suas propriedades também são aplicadas nas equações de *momentum* linear, Eq. 3.13, resultando nas Eq. 4.26,

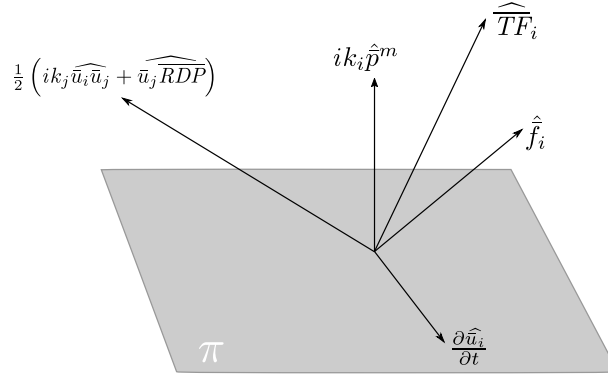
$$\underbrace{\frac{\partial \widehat{u}_i}{\partial t}}_I + \underbrace{\frac{1}{2} \left(ik_j \widehat{u}_i \widehat{u}_j + \widehat{\bar{u}_j RDP} \right)}_{II} = - \underbrace{ik_i \widehat{p}^m}_{III} - \underbrace{\widehat{TF}_i}_{IV} + \underbrace{\widehat{f}_i}_V, \quad (4.26)$$

em que $k^2 = k_i k_j$, $\widehat{RDP}$ representa o Resultado da Derivada Parcial $\frac{\partial \widehat{u}_i}{\partial x_j}$ previamente calculado no domínio espectral e submetido a transformada inversa de Fourier para então ser multiplicado a u_j e $\widehat{TF}_i$ representa a discretização do termo difusivo da Eq. 3.13 em quaisquer das formas discutidas na seção 4.3. Nota-se nos termos advectivos da Eq. 4.26 o processo visto na Eq. 4.24 a fim de evitar-se a resolução de um produto de convolução em cada uma das porções divergente e não divergente da forma antissimétrica.

Analisando a Eq. 4.26 a cerca do plano π , tem-se que o termo I refere-se a taxa temporal de mudança de *momentum* linear. Essa taxa temporal é obtida aplicando um operador diferencial que não altera a posição espacial de $\widehat{u}_i$, portanto, esse termo está contido no plano π .

O termo II corresponde ao termo advectivo na sua forma antissimétrica mas nada pode-se afirmar a respeito de sua posição em relação a π . Em relação ao termo III , o gradiente de pressão, nota-se que a pressão é uma variável escalar, assim a posição vetorial do termo é a mesma de k_j , *i.e.*, o termo é perpendicular ao plano π . O termo IV , modela o comportamento viscoso, e esta apresentado na Eq. 4.26 de uma maneira geral, logo, nada pode ser afirmado sobre a posição desse termo em relação ao plano π . O termo V representa o termo fonte e nada pode-se afirmar sobre sua posição em relação a π . Com base nas análises das posições dos termos da Eq. 4.26 apresenta-se a Fig. 13 onde estão ilustrados os termos da Eq. 4.26 em relação ao plano π .

Figura 13 – Esquema do plano π em relação aos termos das equações de *Momentum* Linear.



Fonte: Próprio autor.

Reorganizando os termos nas Eq. 4.26 com base em suas respectivas posições em relação a π , tem-se a Eq. 4.27,

$$\underbrace{\frac{\partial \widehat{u}_i}{\partial t}}_{\in \pi} + \left[\frac{1}{2} \left(ik_j \widehat{u}_i \widehat{u}_j + \widehat{u}_j \widehat{RDP} \right) + ik_i \widehat{p}^m + \widehat{TF}_i - \widehat{f}_i \right] = 0. \quad (4.27)$$

De acordo com a geometria analítica, dois vetores são colineares se a soma desses vetores é nula. Analisando a Eq. 4.27 nota-se que a soma entre o termo pertencente a π e os demais é nula, assim, pode-se afirmar que o vetor resultante da soma dos termos II , III , IV e V pertence ao plano π .

Embora a soma dos termos *II*, *III*, *IV* e *V* pertence ao plano π pode-se projetar esses termos ao plano em questão para desacoplar o campo de pressão da solução numérico-computacional. Essa operação matemática é feita a partir de um tensor projeção, φ_{ij} , definido como:

$$\varphi_{ij} = \delta_{ij} - \frac{k_i k_j}{k^2}, \quad (4.28)$$

onde

$$\delta_{ij} = \begin{cases} 1 & \text{se } i = j \\ 0 & \text{se } i \neq j \end{cases}, \quad (4.29)$$

é do delta de Kronecker. Silveira Neto²⁶ verifica, de acordo com a Eq. 4.28, que o tensor φ_{ij} é responsável por projetar um vetor qualquer no plano π .

Conforme já apresentado, a soma dos termos *II*, *III*, *IV* e *V* devem pertencer ao plano π e qualquer vetor que é submetido a operação da projeção de φ_{ij} também pertence a este plano. Logo, chega-se a Eq. 4.30,

$$\left[\frac{1}{2} \left(ik_j \widehat{u_i u_j} + \widehat{u_j RDP} \right) + ik_i \widehat{p^m} + \widehat{TF_i} - \widehat{f_i} \right] = \varphi_{im} \left[\frac{1}{2} \left(ik_j \widehat{u_m u_j} + \widehat{u_j RDP} \right) + \right. \quad (4.30) \\ \left. + ik_m \widehat{p^m} + \widehat{TF_m} - \widehat{f_m} \right],$$

em que a expressão pode ser simplificada considerando que a projeção de um vetor perpendicular sobre o plano π é nula. Assim, a Eq. 4.30 resulta na Eq. 4.31:

$$\left[\frac{1}{2} \left(ik_j \widehat{u_i u_j} + \widehat{u_j RDP} \right) + ik_i \widehat{p^m} + \widehat{TF_i} - \widehat{f_i} \right] = \varphi_{im} \left[\frac{1}{2} \left(ik_j \widehat{u_m u_j} + \widehat{u_j RDP} \right) + \widehat{TF_m} - \widehat{f_m} \right]. \quad (4.31)$$

Substituindo a Eq. 4.31 na Eq. 4.27 e isolando-se o termo transiente da expressão, tem-se:

$$\frac{\partial \widehat{u_i}}{\partial t} = -\varphi_{im} \left[\underbrace{\frac{1}{2} \left(ik_j \widehat{u_m u_j} + \widehat{u_j RDP} \right)}_{TNL} + \widehat{TF_m} - \widehat{f_m} \right], \quad (4.32)$$

onde *TNL* representa o termo não linear da equação. A aplicação da projeção na Eq. 4.32 tem dois aspectos significativos. O primeiro é que a equação da continuidade é satisfeita por essa operação já que os vetores resultantes estão contidos no plano π e são perpendiculares ao vetor número de onda, k_i , tal qual apresentado pela Eq. 4.25.

O segundo aspecto é relacionado ao termo de pressão. O gradiente de pressão se torna uma propriedade de pós-processamento já que a projeção é responsável por desacoplá-la das

equações de *momentum* linear, ou seja, substitui-se o processo de solução de um sistema linear para obter o campo de pressão por um produto vetor-matriz, que em termos computacionais é menos oneroso.

4.2.3 Recuperação da Pressão

O campo de pressão espectral pode ser determinado posteriormente, como uma variável de pós processamento, a partir da Eq. 4.31. Isolando-se o gradiente de pressão na Eq. 4.31, tem-se:

$$ik_i \hat{\hat{p}}^m(k_i) = \varphi_{im} \left[\frac{1}{2} \left(ik_j \widehat{\bar{u}_m \bar{u}_j} + \widehat{\bar{u}_j RDP} \right) + \widehat{TF}_m - \widehat{f}_m \right] + I_{im} \left[\frac{1}{2} \left(ik_j \widehat{\bar{u}_m \bar{u}_j} + \widehat{\bar{u}_j RDP} \right) + \widehat{TF}_m - \widehat{f}_m \right], \quad (4.33)$$

onde I_{im} é um tensor identidade, com forma análoga a δ_{ij} , introduzido para compatibilização de notação tensorial e sem impacto na Eq. 4.33. Colocando-se em evidência os termos entre colchetes, chega-se a:

$$ik_i \hat{\hat{p}}^m(k_i) = (\varphi_{im} - I_{im}) \left[\frac{1}{2} \left(ik_j \widehat{\bar{u}_m \bar{u}_j} + \widehat{\bar{u}_j RDP} \right) + \widehat{TF}_m - \widehat{f}_m \right]. \quad (4.34)$$

Em seguida, pode-se realizar um produto escalar entre a Eq. 4.34 e o vetor número de onda k_i , resultando em:

$$ik_i^2 \hat{\hat{p}}^m(k_i) = (\varphi_{im} - I_{im}) k_i \left[\frac{1}{2} \left(ik_j \widehat{\bar{u}_m \bar{u}_j} + \widehat{\bar{u}_j RDP} \right) + \widehat{TF}_m - \widehat{f}_m \right]. \quad (4.35)$$

Para melhor compreensão do termo $(\varphi_{im} - I_{im}) k_i$, pode-se substituir a variável φ_{im} pela sua definição, apresentada na Eq. 4.28. Assim, o detalhamento do termo junto a sua simplificação é visto na Eq. 4.36,

$$\left(\delta_{im} - \frac{k_i k_m}{k^2} - I_{im} \right) k_i = \underbrace{\delta_{im} k_i}_{k_m} - \underbrace{\frac{k_i k_i k_m}{k^2}}_{k_m} - \underbrace{I_{im} k_i}_{k_m} = -k_m. \quad (4.36)$$

Substituindo $-k_m$ e multiplicando a Eq. 4.35 por $-i/k^2$, chega-se na Eq. 4.37,

$$\hat{\hat{p}}^m(k_i) = \frac{ik_m}{k^2} \left[\frac{1}{2} \left(ik_j \widehat{\bar{u}_m \bar{u}_j} + \widehat{\bar{u}_j RDP} \right) + \widehat{TF}_m - \widehat{f}_m(k_i) \right], \quad (4.37)$$

onde através da transformada inversa de Fourier, Eq. 4.22, determina-se o campo de pressão no espaço físico.

Destaca-se que as fronteiras ou condições de contorno do MPF não foram abordadas. Isso ocorre por dois motivos distintos. O primeiro é que já foi apresentado como parte das condições de contorno dos escoamentos modelados são implementadas via MDFM. A outra é que o uso da FFT, a ser abordado na subseção 4.3.2, impõe um tratamento de fronteira do domínio computacional rígido.

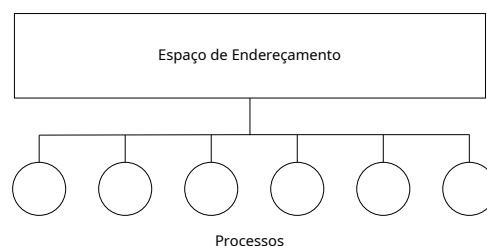
4.3 Aspectos Numérico-Computacionais

4.3.1 Paralelismos Implementados

O IMERSPEC é uma metodologia de alta ordem e assim, naturalmente, apresenta um custo computacional considerável quando comparado a métodos usuais como o método de Volumes Finitos⁸⁵. A fim de reduzir o custo computacional desta metodologia utilizou-se distintas técnicas de paralelização.

Dentre as técnicas de paralelismo disponíveis, destaca-se a memória compartilhada, a passagem de mensagens e o paralelismo de dados.

Figura 14 – Esquema de paralelização do tipo memória compartilhada.



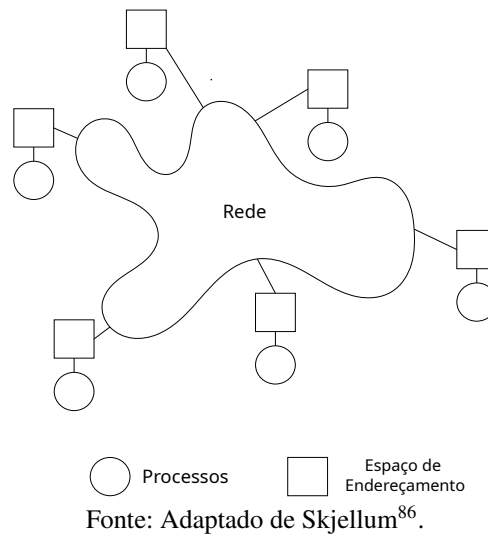
Fonte: Adaptado de Skjellum⁸⁶.

A paralelização de um código com memória compartilhada é caracterizada por processos com acesso a todo o espaço de endereçamento do carregamento e armazenamento de operações. Ainda sim, cada processo tem o seu acesso a apenas uma porção de toda a informação. O acesso às informações é, usualmente, coordenado e controlado por algum tipo de “tranca”⁸⁶. Um esquema de execução desse tipo de paralelização é apresentado na Fig. 14.

Segundo essa abordagem, o código paralelizado é executado em um ambiente com *multi-threads* com um único espaço de endereçamento com diversos contadores do programa e *stacks* de execução. O principal desafio dessa paralelização é que qualquer que seja o “estado” do programa, definido por um valor das variáveis do programa, é compartilhado por todas as *threads* simultaneamente⁸⁶. A maior representante desse tipo de paralelização é a biblioteca OpenMP.

Outro tipo de paralelização é a do tipo de passagem de mensagens, em inglês *message passing*, Fig. 15. Ao contrário da abordagem anterior, nesta cada processo ou núcleo físico possui apenas memória local mas conseguem se comunicar com outros processos. Os processos se comunicam entre si a partir de uma rede e para ocorrer comunicação um processo deve,

Figura 15 – Esquema de paralelização do tipo passagem de memória.



explicitamente, enviar informação enquanto o outro processo deve, explicitamente, receber tal informação.

Uma característica limitante desse processo é justamente a rede a qual transfere a informação entre os processos. A troca de informação entre os processos é limitada pela a velocidade que a rede consegue transmitir. A biblioteca computacional mais popular relacionada a esse tipo de paralelização é MPI.

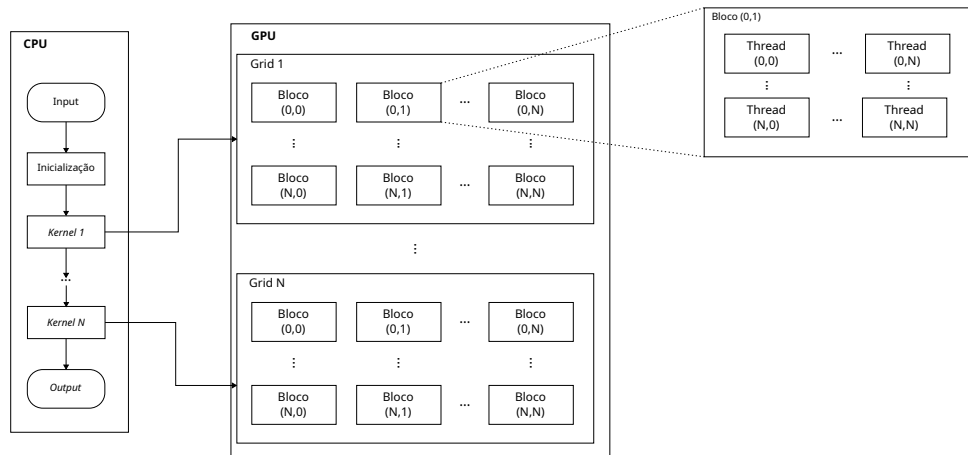
No paralelismo de dados, tem-se uma única instrução para múltiplos dados, do inglês *Single-Instruction, Multiple-Data*, (SIMD). Seguindo esse estilo de programação, a paralelização vem inteiramente a partir dos dados enquanto o programa em si se assemelha a um programa sequencial⁸⁶. Esse tipo de paralelismo é explorado em programas que utilizam Unidades Gráficas de processamento, do inglês *Graphical Processing Units*, (GPU).

A partir de uma unidade de processamento gráfico de Propósito Geral, do inglês *General Purpose Graphics Processing Unit*, (GPGPU) é possível desenvolver códigos, os quais utilizam GPU/CPU para a modelagem de fenômenos físicos. Nesse tipo de aplicação o GPU atua como coprocessador junto a unidade central de processamento, do inglês *Central Processing Unit*, (CPU) através da arquitetura de dispositivo de computação unificada, do inglês *Computer Unified Device Architecture*, (CUDA).

Analisando a configuração de um GPGPU, nota-se um paralelismo heterogêneo entre o dispositivo, GPU, e o *host*, CPU, entretanto a maior vantagem dessa união é a execução de tarefas pelo GPU, um ambiente massivamente paralelo pelas dezenas de *threads* disponíveis.

A interface de programação de aplicações, do inglês *Application Programming Interface*, (API) CUDA foi desenvolvida pela NVIDIA e tornou possível a chamada de funções que são executadas no GPU apesar de serem chamadas no CPU. Essas funções são denominadas como *kernels*. Cada *kernel* é executado por um conjunto de blocos de *threads*. A quantidade de blocos

Figura 16 – Esquema de paralelização heterogêneo CPU/GPU.



Fonte: Adaptado de Xiang *et al.*⁷⁰.

e a quantidade de *threads* por bloco são definidas na linha de chamada do *kernel* no CPU. A Figura 16 é uma representação esquemática de uma paralelização heterogênea CPU/GPU. Em relação ao paralelismo GPGPU destaca-se a memória do CPU e do GPU são, em geral, separadas e assim deve haver transferência de informação explícita entre os dois processadores.

A metodologia IMERSPEC foi implementada em dois códigos computacionais com técnicas de paralelização diferentes. A paralelização computacional é utilizada como uma forma de aumento de performance e capacidade computacional em relação a um código serial. O primeiro código foi desenvolvido em Fortran 2018 com paralelização MPI-3 (do inglês, *Message Passing Interface*) enquanto o segundo código foi desenvolvido em CUDA C/C++ (do inglês, *Compute Unified Device Architecture*) com a paralelização de dados vetorizados na forma do uso de GPU junto ao CPU. Assim, respectivamente, desenvolveu-se o código IMERSPEC MPI e o código IMERSPEC CUDA.

O modelo de paralelização *message passing*, presente em IMERSPEC MPI, consiste de uma série de processos independentes entre si que cada um deles apresentam apenas memória local mas ainda sim podem se comunicar entre si enviando e recebendo mensagens⁸⁶. Destaca-se que há uma restrição nesse tipo de paralelismo computacional onde a transferência de dados só acontece se um processo explicitamente recebe dada informação enquanto outro explicitamente a envia e que a paralelização é feita em apenas duas das três direções dos escoamentos modelados devido a restrição da biblioteca utilizada para computar a transformada de Fourier.

O segundo tipo de paralelismo explorado no presente trabalho é o paralelismo de dados obtido através do emprego de GPU, presente no código IMERSPEC CUDA. Em termos gerais, enquanto o CPU foi desenvolvido para executar uma sequência de operações denominada *thread* o mais rápido possível e consegue executar apenas um grupo de *threads* em paralelo, o GPU foi desenvolvido para maximizar a execução de centenas dessas *threads* em paralelo (amortizando a performance lenta de uma única *thread* para alcançar maior rendimento)⁸⁷. Qualquer código

computacional é composto de regiões paralelizáveis e sequenciais, assim o código IMERSPEC CUDA é implementado com operações realizadas tanto em GPU quanto em CPU.

Destaca-se que a implementação que utiliza NUFFTs são exclusivas do código em CUDA C/C++ devido ao suporte da biblioteca em questão⁸⁴, *i.e.*, não há um correspondente pacote para o código que utiliza MPI.

4.3.2 Transformadas Rápidas de Fourier

No presente trabalho busca-se modelar escoamentos através de uma abordagem computacional. Sendo assim, as Eq. 4.21 e 4.22 são utilizadas em suas formas discretas denominadas por Transformada Discreta de Fourier, em inglês *Discrete Fourier Transform*, (DFT)⁸⁰,

$$\hat{\varsigma}_k = \sum_{n=-\frac{N}{2}+1}^{\frac{N}{2}} \varsigma_n \exp\left(-\frac{i2k_n\pi}{N}\right), \quad (4.38)$$

e por Transformada Discreta Inversa de Fourier, em inglês *Inverse Discrete Fourier Transform*, (IDFT)⁸⁰,

$$\varsigma_n = \frac{1}{N} \sum_{k=-\frac{N}{2}+1}^{\frac{N}{2}} \hat{\varsigma}_k \exp\left(\frac{i2nk\pi}{N}\right), \quad (4.39)$$

em que ς_n é a informação ς discretizada no domínio físico com n nós de colocação e $\hat{\varsigma}_k$ se refere a mesma informação discretizada no domínio espectral de Fourier; n representa a posição x_n dos nós de colocação já que $x_n = n\Delta x$; $\Delta x = \frac{L_x}{N}$ e N é o número de pontos que o domínio computacional foi discretizado na direção x .

A aplicação de um algoritmo computacional baseado nas Eq. 4.38 e 4.39 é computacionalmente caro, pois N^2 operações são feitas para cada variável que é transformada. Por outro lado, o algoritmo denominado por transformada rápida de Fourier, do inglês *Fast Fourier Transform*, (FFT) consegue reduzir de N^2 operações para $N \log_2 N$ operações explorando periodicidade e simetria de funções trigonométricas⁸⁸. Devido ao seu menor custo computacional, utiliza-se a FFT dentro do código computacional já que a abordagem pseudoespectral na solução computacional da Eq. 4.32 implica em diversas operações de transformada e transformada inversa de Fourier em um mesmo instante temporal t .

Apesar da redução do custo computacional causada pelo uso da FFT, esse algoritmo usualmente é aplicado em malhas computacionais uniformes, o que invalida procedimentos usuais de refinamento de malha local dentro do IMERSPEC. Outro destaque do uso da FFT é a imposição de condições de periodicidade a função que será transformada. Dessa maneira, são exigidas condições de contorno periódicas nos escoamentos modelados com o IMERSPEC, logo, artifícios específicos devem ser utilizados a fim de evitar modelar condições de contorno físicas, ou seja, não periódicas, como as condições de Dirichlet, Neumann ou Robin.

O pacote de FFT utilizado no código Fortran-MPI é escrito por completo, no pacote clássico⁸⁹, ou majoritariamente, no pacote otimizado por *SPIRAL*⁹⁰, em fortran denominado por *FFTE*. Takahashi; Franchetti⁹⁰ implementaram otimizações no pacote clássico *FFTE* através do sistema de código aberto para a geração de código otimizado voltado principalmente para processamento de sinais, *SPIRAL*, onde obtiveram FFTs 3,16 vezes mais rápidas. O código IMERSPEC MPI utiliza essas duas versões da biblioteca *FFTE* onde a paralelização é feita apenas em duas das três direções do domínio de cálculo, y e z .

No código em CUDA C/C++, por outro lado, foi utilizada a biblioteca da *NVIDIA* denominada por *cuFFT*, que por sua vez, foi baseada no pacote FFTW (do inglês, *Fastest Fourier Transform in the West*).

A aplicação da FFT depende da determinação dos números de onda k_i ,

$$k_i(n) = \begin{cases} \frac{2\pi}{L_i}(n-1) & 1 \leq n \leq \frac{N}{2} + 1 \\ \frac{2\pi}{L_i}(n-1-N) & \frac{N}{2} + 2 \leq n \leq N \end{cases} \quad (4.40)$$

onde k_i é o componente i do vetor número de onda; N é o número de nós de colocação em dada direção; L_i é o comprimento do domínio na dada direção; e n representa a posição no vetor. Além do uso do vetor na aplicação da FFT, k_i também é necessário para algumas operações no domínio de Fourier, como por exemplo, o cálculo de derivadas.

Conforme discutido na seção 4.1.3, há a oportunidade de utilizar NUFFTs do tipo 1 do tipo 2 para a implementação do MIED. Destaca-se que esses procedimentos se tratam de aproximações e assim não é desejável que essa biblioteca seja implementada dentro do MPF. A NUFFT do tipo 1, responsável pela conversão de uma malha não uniforme no domínio físico para uma malha uniforme espectral é representada pela Eq. 4.41,

$$\hat{g}_k = \sum_{j=1}^M c_j \exp^{-ik_j x_j}, \quad (4.41)$$

enquanto a NUFFT do tipo 2, responsável pela conversão de uma malha uniforme espectral para uma malha não uniforme no domínio físico, é caracterizada pela Eq. 4.42,

$$c_j = \sum_{k_i \in \mathbb{C}} \hat{g}_k \exp^{ik_i x_j}. \quad (4.42)$$

Conforme visto nas Eq. 4.41 e 4.42, NUFFT do tipo 1 e do tipo 2 tem um comportamento oposto entre si, assim, o procedimento das duas é similar o que se altera é a ordem. De acordo com Barnett; Magland; af Klinteberg⁹¹, os seguintes passos para realizar a transformada discreta são:

1. Define-se uma malha regular com $n = \sigma^d N$ pontos onde d representa a dimensão do problema e σ é um fator *upsampling* com limites $1 < \sigma \leq 2$;

2. Calcula-se a convolução dos dados com uma função suave, ψ , pré estabelecida pela biblioteca, cujo suporte tem v pontos na malha definida anteriormente. Assim, essa operação requer $v^d N_b$ avaliações onde N_b é o tamanho da malha não uniforme;
3. Aplica-se a FFT da malha N resultante das operações anteriores com complexidade $O(N \log N)$;
4. Após a operação FFT, deseja-se eliminar o efeito da função ψ sobre os resultados. Dessa forma, emprega-se a deconvolução ou o correção *roll-off* onde são extraídas as N baixas frequências da transformada e esse resultado é então dividido por $\hat{\psi}$, finalizando o procedimento da NUFFT.

A escolha da função suave ψ utilizada no algoritmo é tal que haja um erro ϵ que decresça exponencialmente com v e, portanto, passa-se definir $v \approx cte |\log \epsilon|$. Logo, pode-se definir que o gasto computacional da NUFFT é definido por $O(N_b |\log \epsilon|^d + N \log N)$.

4.3.3 Periodização de Escoamentos Não Periódicos

Conforme mencionado, o uso da FFT restringe a aplicação da metodologia a escoamentos periódicos. Logo, a fim de transformar escoamentos com condições de contorno não periódicas em periódicas, utiliza-se o método de *Buffer*. O método é imposto dentro do IMERSPEC através do termo fonte da Eq. 4.32 e é implementado em uma região do escoamento específica, uma zona, denominada por domínio complementar. Nessa porção do domínio o termo fonte é definido através da Eq. 4.43,

$$f_i = \Psi (\bar{u}_i - Qt x_i), \quad (4.43)$$

em que Ψ é uma função de amortecimento suave, responsável pela intensidade da imposição do método sob o escoamento modelado, que apresenta mínimo igual a 0 e máximo igual a unidade; $\bar{u}_i$ é a velocidade filtrada do fluido no instante t ; e $Qt x_i$ são as soluções alvo desejada na direção x_i .

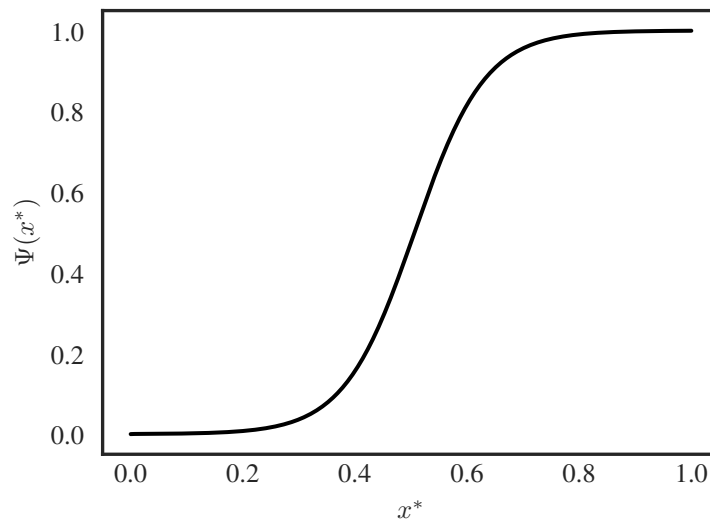
No domínio complementar são utilizados artifícios para que escoamentos aperiódicos sejam modelados. A porção do domínio onde é modelado o fenômeno físico de interesse recebe a nomenclatura de domínio físico. Portanto, o domínio computacional é subdividido em domínio físico e complementar. Destaca-se que no domínio complementar não há modelagem de fenômenos naturais porém, ainda sim, a equação da continuidade é satisfeita nessa zona.

O método *buffer* impõe, ao longo do domínio complementar, a solução desejada na entrada do domínio físico, entretanto essa imposição é suavizada na entrada da região complementar devido aos baixos valores de $\Psi(x)$. Esse método é comumente utilizado no IMERSPEC^{42,23,14,53} e é caracterizado pela função $\Psi(x)$,

$$\Psi(x) = \frac{1}{2} \left[1 - \tanh \left(4 - 8 \frac{x_i - x}{x_i - x_f} \right) \right], \quad (4.44)$$

em que x_i se refere ao começo da zona de *Buffer* e x_f representa o fim da zona. A função Ψ é ilustrada na Fig. 17 para uma coordenada normalizada x^* . Em relação a Eq. 4.43, $Qt x_i$ apresenta valores constantes e iguais a condição de entrada do escoamento modelado no domínio físico. Outro destaque é que esse método é implementado em uma zona de *Buffer* a qual possui, a jusante, uma zona de forçagem em que a condição de entrada é imposta através do MDFM e ambas as zonas compõe o domínio complementar.

Figura 17 – Função amortecedora do método *buffer*.

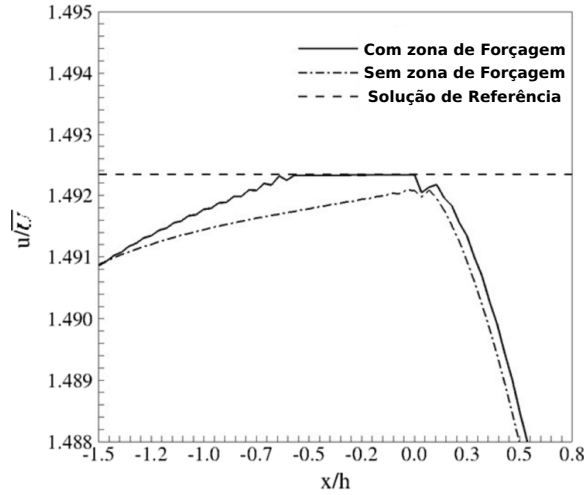


Fonte: Próprio autor.

Alternativamente ao método de *buffer*, tem-se um método similar denominado por método *fringe*. A forma de implementação é análoga ao método *buffer* entretanto a função amortecedora é suave e nula no início e fim do domínio complementar. O método *fringe* foi utilizado previamente no IMERSPEC para a modelagem bidimensional do escoamento de Blasius, onde este performou melhor do que o método de *buffer* especificamente para essa classe de escoamento⁶².

Tradicionalmente na metodologia IMERSPEC^{42,23,55,53,14}, o domínio complementar é composto de uma região denominada por zona de *buffer* onde o método de *buffer* é aplicado seguida de uma região onde a condição de entrada do escoamento é imposta integralmente por MDFM para aumentar a exatidão, denominada por zona de forçagem. A Figura 18 ilustra uma modelagem IMERSPEC realizada com apenas a zona *buffer* e outra com a zona de forçagem e *buffer*, comparando-se com a solução de referência, nota-se que o domínio complementar composto das duas zonas apresenta maior exatidão.

Figura 18 – Impacto da zona de forçagem na exatidão da modelagem IMERSPEC.



Fonte: Adaptado de Mariano *et al.*⁵².

4.3.4 O problema de *aliasing*

Após a apresentação do MFI e do MPF, a junção dos dois métodos resulta na metodologia IMERSPEC, que é uma metodologia pseudoespectral. Sendo assim, fenômenos numéricos já identificados para abordagens espectrais também estarão presentes nas modelagens realizadas com o IMERSPEC.

Dentre esses fenômenos destaca-se o *aliasing* que é definido, por Boyd⁶³, a partir de um intervalo $x \in [-\pi, \pi]$ que é discretizado em uma malha uniforme h e então os seus números de onda do interpolante trigonométrico são tais que $k \in [-K, K]$ onde $K = \pi/h$ e é definido como o limite de *aliasing* dos números de onda, também conhecido como o limite de Nyquist. Os erros devido ao *aliasing* são introduzidos pela discretização de sinais no espaço de Fourier, a periodicidade do domínio devido ao uso de FFT e a regra da ortogonalidade das bases de Fourier. Os elementos da base de Fourier são $\exp(ikx)$ com k sendo um inteiro. Altos números de onda k que sejam $|k| > K$ se comportarão numericamente como um correspondente $k_A = k \pm 2mK$ em que m é um inteiro e $k_a \in [-K, K]$. Os números de onda fora do intervalo $[-K, K]$ são denominados por “*aliased*” em relação aos números de onda dentro da faixa definida e k_a é dito como “*alias*”.

Em convoluções no domínio espectral é explícito o aumento da banda de números de onda. A convolução apresentada para duas funções contínuas pela Eq. 4.23 pode ser representada em um contexto discreto como:

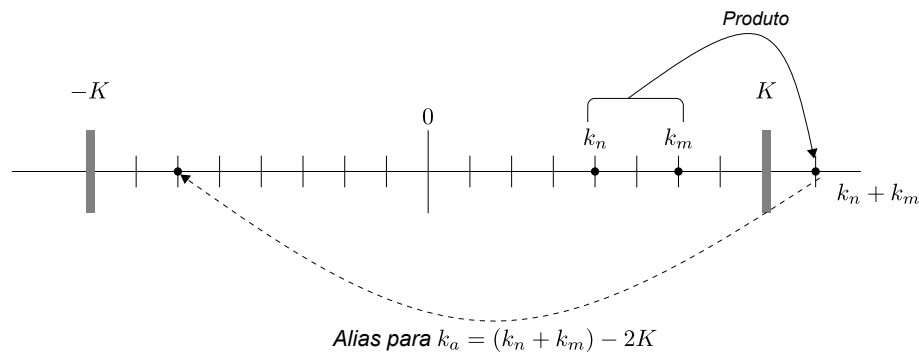
$$\hat{\sigma}_k = [\hat{\varsigma} * \hat{\psi}]_k = \sum_{p_i + q_i = k_i} \hat{\varsigma}_p \hat{\psi}_q. \quad (4.45)$$

Assim, o aumento da banda de números de onda k_i é explicitada e, necessariamente, haverá $|k_i| > K$, isto é, erros de *aliasing*.

Um argumento usual ao se explicitar o fenômeno de *aliasing* em uma convolução é de que estas operações não são realizadas no domínio espectral visto que o IMERSPEC é uma abordagem pseudoespectral, assim convoluções do domínio de Fourier são substituídas por produtos das respectivas funções no domínio físico. Esse é um argumento falho. Como destacado anteriormente, o domínio físico e o espectral são equivalentes. Os efeitos de uma operação em um dos domínios terá o mesmo efeito no outro domínio. Isso decorre diretamente do teorema da convolução, segundo o qual um produto no domínio físico corresponde exatamente a uma convolução no domínio espectral, e vice-versa⁸⁰. O que se ganha na abordagem pseudoespectral é tempo e acurácia computacionais. Mesmo assim, o resultado de um produto de duas funções realizados no domínio físico, apresentará igualmente erros de *aliasing*.

O problema numérico decorrente dessa convolução é esquematizado na Fig. 19, em que se ilustra, para um único par de números de onda k_n e k_m pertencentes à somatória da Eq. 4.45, o mecanismo pelo qual o produto $k_n + k_m$ ultrapassa o limite de Nyquist K e é incorretamente representado pelo seu *alias* $k_a = (k_n + k_m) - 2K$.

Figura 19 – Esquema de *aliasing*.



Fonte: Próprio autor.

O *aliasing* pode ser a causa de divergência numérica em códigos espectrais. Um dos precursores desse tipo de divergência são as ondas $2h$ no domínio físico, *i.e.*, ondas que o número de onda é duas vezes o espaçamento h da malha uniforme e que oscilam de sinal entre dois pontos colocantes adjacentes⁶³. Uma solução computacional com ondas $2h$ provavelmente apresentou problemas de *aliasing* ou apresentou instabilidades decorrentes da condição de estabilidade de *Courant-Friedrichs-Lewis*, CFL. Portanto, em suma, as ondas $2h$ são precursoras de divergências e devem ser evitadas.

Em operações não-lineares (como, usualmente, a advecção) o *aliasing* pode causar instabilidades numéricas devido as interações não lineares do espectro de números de onda resultante de convoluções ou produtos de termos físicos que causam altos números de ondas zonais, *i.e.*, componentes de onda em uma única direção do escoamento modelado. Altos números de onda zonais correspondem a ondas com comprimentos de onda muito pequenos ou com frequências altas⁶³. As ondas $2h$ são um exemplo de ondas zonais.

Tradicionalmente, há algumas opções para evitar o *aliasing* na solução numérico-computacional de abordagens espectrais, dentre elas pode-se citar algumas como⁶³:

- Forma Antissimétrica;
- Refinamento de h ;
- Filtragem ou utilização de viscosidade artificial;
- *Dealiasing*.

A primeira opção disponível será abordada em detalhes na próxima subseção. Quanto a segunda, o refinamento de malha computacional, essa opção pode retardar uma divergência numérica, mas não a erradica. Entretanto, caso o fenômeno físico exija uma resolução computacional maior para a correta modelagem essa solução é eficiente e não há necessariamente um problema de *aliasing* dentro do código.

O uso de filtragem dentro do código pode suavizar os fenômenos numéricos indesejados entretanto, ao mesmo tempo, causa uma perda de exatidão na simulação numérico-computacional, uma das principais vantagens em abordagens espectrais. A utilização de viscosidade artificial pode garantir uma solução computacional sem erros numéricos, entretanto, também pode causar experimentos virtuais sem quaisquer resquícios físicos não realizando o principal objetivo de qualquer simulação numérico-computacional, a modelagem do fenômeno físico.

O processo definido por *dealiasing* pode ser entendido como um processo especial de filtragem em que o objetivo é remover o *aliasing* de computações quadráticas não-lineares. A regra dos dois terços é um algoritmo que exemplifica o *dealiasing* no qual são descartados 19/27 de números de liberdade em um escoamento tridimensional, *i.e.*, a exatidão da solução é prejudicada significativamente. Essa ação de descarte do *dealiasing* é equivalente a um filtro “passa baixa” em que os coeficientes de Fourier são descartados ou se mantém inalterados, caracterizando assim uma filtragem abrupta⁶³.

A regra 2/3 pode ser escrita na forma de um filtro $\hat{G}$ definido pela Eq. 4.46:

$$\hat{G}(k_i) = \begin{cases} 1, & \text{se } |k_i| \leq \frac{2}{3}k_{max} \\ 0, & \text{se } |k_i| > \frac{2}{3}k_{max} \end{cases}, \quad (4.46)$$

em que $k_{max} = N/2$ para N pontos colocantes. Esse esquema clássico de *dealiasing* garante que o espectro de energia cinética se mantenha estritamente de acordo com o fenômeno físico sem blocagem espectral⁶³.

Diante da perda de exatidão inerente à regra dos 2/3, alternativas de *dealiasing* menos restritivas têm sido propostas na literatura, como a filtragem espectral suave discutida por Hou; Li⁶⁵. No presente trabalho, avalia-se tanto a regra clássica dos 2/3 quanto um filtro exponencial

como estratégia de *dealiasing* mais suavizado e com proposta para amenizar a perda de exatidão das simulações. Para um processo de *dealiasing* utilizando um filtro exponencial, a forma do filtro, Eq. 4.46, é alterada para:

$$\hat{G}(k_i) = \exp \left[-A \left(\frac{|k|}{k_{max}} \right)^s \right], \quad (4.47)$$

onde $A = 36$ é a intensidade do filtro, conforme proposto por Hou; Li⁶⁵, e $s = 16$ é a ordem adotada no presente trabalho. Destaca-se que Hou; Li⁶⁵ originalmente propõem $s = 36$, valor que garante que o filtro atinja a precisão de máquina quando $|k| \rightarrow k_{max}$. A redução para $s = 16$ adotada no presente trabalho resulta em um decaimento menos abrupto, preservando um maior número de modos espectrais na região de transição do filtro e, portanto, uma forma de evitar o fenômeno de Gibbs e contrair com a regra clássica de $2/3$.

Por fim, destaca-se que processos de *dealiasing* não são usados para todos os fenômenos físicos modelados. No contexto da presente tese, eles serão utilizados apenas nas modelagens discutidas no capítulo 6.

4.3.5 Formas de Discretização do Termo Difusivo

A forma antissimétrica é uma outra alternativa para remediar o fenômeno de *aliasing* e tradicionalmente utilizada apenas no termo advectivo das Eq. 3.13 conforme adotado até então na metodologia IMERSPEC a partir de análises realizadas por Mariano¹⁰. Blaisdell; Spyropoulos; Qin⁶⁴ analisaram os erros devido ao fenômeno de *aliasing* decorrentes da discretização do termo advectivo nas formas divergente, não divergente e antissimétrica através do domínio espectral que provaram que o *aliasing* é minimizado na forma antissimétrica quando comparado à outras discretizações.

Em todas as abordagens da forma antissimétrica, tradicionalmente, emprega-se esse tipo de discretização apenas no termo advectivo, entretanto, essa forma de discretização também pode ser utilizada em outros termos não lineares como, por exemplo, o termo difusivo com coeficiente de transporte não uniforme⁶⁴. Com base nessa proposta feita no presente trabalho, o termo difusivo filtrado, $\overline{TF}$, da Eq. 3.13 pode ser discretizado de forma antissimétrica da Eq. 4.48,

$$\overline{TF}_i = 0,5 \frac{\partial}{\partial x_i} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] + 0,5 \left[\frac{\partial \nu_{eff}}{\partial x_i} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) + \nu_{eff} \frac{\partial}{\partial x_i} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right]. \quad (4.48)$$

Considerando que a abordagem do presente trabalho também é feita para modelagens de escoamentos laminares, implica-se que $\frac{\partial \nu_{eff}}{\partial x_i} = 0$ visto que $\nu_{eff} = \nu$ e a Eq. 4.48 pode ser reduzida em:

$$TF_i = 0,5 \frac{\partial}{\partial x_i} \left[\nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + 0,5 \nu \frac{\partial}{\partial x_i} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \quad (4.49)$$

Usualmente a metodologia IMERSPEC tridimensional, utiliza para o termo difusivo na forma divergente visto na Eq. 4.50,

$$\overline{TF}_i = \frac{\partial}{\partial x_i} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right]. \quad (4.50)$$

Exclusivamente, para escoamentos laminares, a Eq. 4.50 pode ser reduzida na forma da Eq. 4.51,

$$TF_i = \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j}, \quad (4.51)$$

embora essa forma seja específica para essa classe de escoamentos e é utilizada majoritariamente na metodologia IMERSPEC bidimensional devido a sua especificidade.

Ressalta-se que a forma antissimétrica não elimina o fenômeno de *aliasing*, mas sim atua minimizando a transferência espúria de energia para os altos números de onda durante as interações não lineares, conservando-se discretamente a energia cinética do escoamento⁶⁴.

4.3.6 Método de Avanço Temporal de Alta Ordem

O MPF garante alta acurácia e alta ordem de convergência na discretização espacial, além de desacoplar a solução do gradiente de pressão, que passa a ser um resultado apenas de pós-processamento, e o uso da FFT garante uma redução do custo computacional na solução das equações de Navier-Stokes para escoamentos incompressíveis. Como um método de alta ordem é utilizado na discretização espacial, um método de alta ordem também deve ser empregado na discretização temporal¹⁰. Portanto, o IMERSPEC utiliza o Método de Runge-Kutta de 4ª Ordem com 6 passos (RK46).

A escolha por esse método de avanço temporal foi feita porque os coeficientes do RK46 foram otimizados no espaço espectral apresentando assim: uma redução do armazenamento de variáveis e uma baixa dispersão e dissipação numérico-computacional⁸³. O algoritmo RK46 é exemplificado pela Eq. 4.52,

$$\begin{cases} \overline{AUX}_i^l = \alpha^l AUX_i^{l-1} + \Delta t \left[-\varphi_{im} \left(\widehat{TNL}_m + \widehat{TF}_m \right) \right]^l, \\ \hat{u}_i^{*,l+1} = \hat{u}_i^{*,l} + \beta^l \overline{AUX}_i^l \end{cases} \quad (4.52)$$

onde $l = 1, 2, \dots, 6$ é o passo do método, $\overline{AUX}$ é uma variável filtrada auxiliar, α^l e β^l são constantes apresentadas na Tab. 2.

O incremento temporal utilizado no IMERSPEC pode ser variável. Caso seja variável, o incremento é dependente da condição CFL e igual a:

Tabela 2 – Parâmetros de MRK46.

l	α	β
1	0,0	0,122
2	-0,691750960670	0,477263056358
3	-1,727127405211	0,381941220320
4	-0,694890150986	0,447757195744
5	-1,039942756197	0,498614246822
6	-1,531977447611	0,186648570846

Fonte: Allampalli *et al.*⁸³.

$$\Delta t = CFL \cdot \min \left\{ \min \left[\frac{\Delta x_1}{\max ||\bar{u}_1||}; \frac{\Delta x_2}{\max ||\bar{u}_2||}; \frac{\Delta x_3}{\max ||\bar{u}_3||} \right]; \frac{2}{\max \nu_{eff}} \left(\frac{1}{\Delta x_1^2} + \frac{1}{\Delta x_2^2} + \frac{1}{\Delta x_3^2} \right)^{-1} \right\}. \quad (4.53)$$

O RK46 é um método temporal explícito, assim, o CFL deve ser sempre inferior a unidade para que se mantenha a estabilidade computacional.

O método de avanço temporal é responsável pelo cálculo do parâmetro temporário u_i^* , o qual ainda será atualizado através do MFI. Assim, a definição da velocidade no próximo instante temporal é feita através da projeção do termo fonte obtido pelo MFI conforme apresentado pela Eq. 4.54,

$$\hat{u}_i^{t+\Delta t} = \hat{u}_i^{it+1} = \hat{u}_i^* + \Delta t \varphi_{im} \hat{f}_m. \quad (4.54)$$

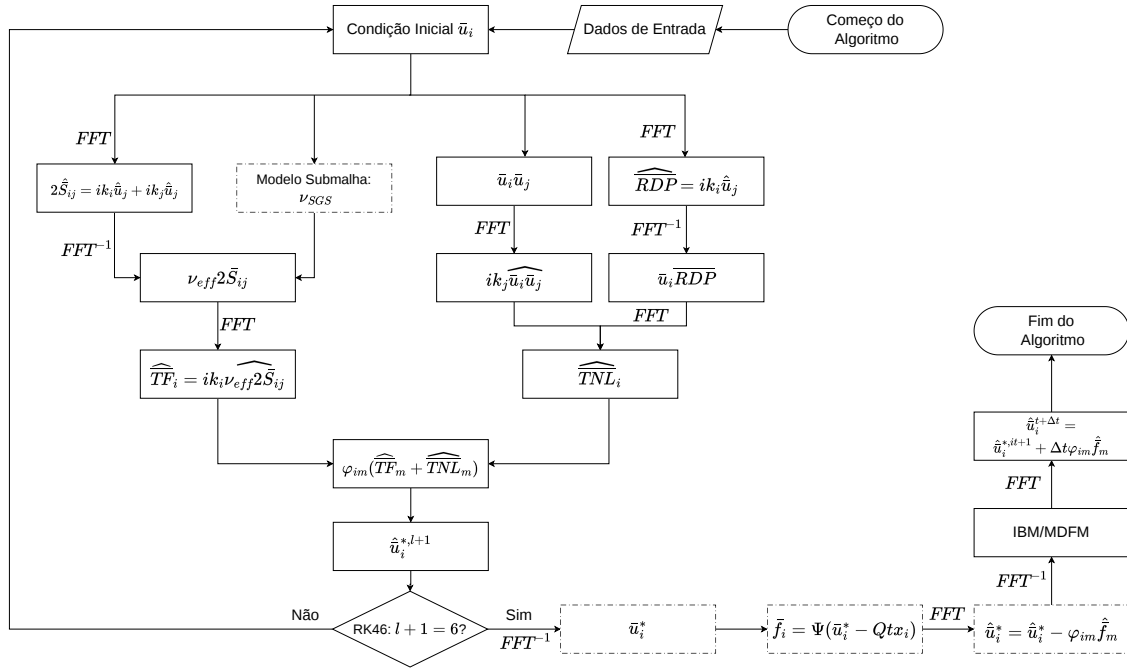
A Eq. 4.54 é uma forma atualizada da Eq. 4.12, a atualização de velocidades dentro do MDFM passa a ser feita no domínio espectral para que possa-se realizar a operação de projeção sobre o termo fonte, a qual garante que a equação de continuidade seja satisfeita. Outro aspecto da Eq. 4.54 é que ela está inserida dentro de um processo iterativo, o MDFM, e a velocidade $\hat{u}^{t+\Delta t}$ deve satisfazer critério do MDFM.

4.3.7 Estrutura Geral IMERSPEC

A fim de simplificar a compreensão global da metodologia IMERSPEC, na Fig. 20 apresenta-se um fluxograma simplificado da metodologia independente de técnicas de paralelizações ou tipo de FFT usada, sem representação de *dealising* e utilizando a forma do termo difusivo da Eq. 4.50. Apesar de simplificado, são apresentados algumas etapas do algoritmo não essenciais no fluxograma da Fig. 20. Essas etapas são representadas pelos blocos delimitados por linha traço ponto.

A primeira etapa opcional é a modelagem submalha, utilizada apenas quando se deseja representar a física de um escoamento turbulento, *i.e.* para modelagens laminares não é utilizado

Figura 20 – Fluxograma simplificado da metodologia IMERSPEC.



Fonte: Próprio autor.

qualquer modelo dessa natureza. O próprio termo difusivo não tem essa forma para modelagens laminares, é significamente mais simples, Eq. 4.51.

As outras etapas destacadas são relacionadas a modelagens de fenômenos não periódicos pelo IMERSPEC, nesse caso as etapas atuarão na zona *Buffer* enquanto a zona de forçagem é aplicada na etapa de MDFM junto com a geometria imersa necessária a modelagem. Analisando a Fig. 20 destaca-se outra significativa simplificação, quando comparado com os esquemas das Fig. 9 e 10, o MDFM é representado como apenas uma iteração, isso foi feito apenas por razões didáticas e salienta-se que o processo continua iterativo.

Nesse capítulo foi apresentado a metodologia IMERSPEC, começando pelos dois modelos principais que compõe a metodologia: MPF e IBM, seguidos de discussões de aspectos numérico-computacionais relativos a sua implementação computacional. Limitações do uso das FFTs como a sua necessidade de fronteiras periódicas foi salientado e, assim, também foi discutidos métodos para modelar fenômenos não periódicos na abordagem IMERSPEC. Destaca-se que também foi discutido o problema de *aliasing*, significativo em abordagens espectrais, e formas de erradicá-lo ou atenuá-lo. Conforme apresentado, foi desenvolvido dois códigos computacionais com técnicas de paralelização distintas para a metodologia IMERSPEC. A comparação da performance dos dois códigos distintos para a verificação numérica da metodologia e validação por um escoamento laminar são o tema do próximo capítulo.

5 RESULTADOS: VERIFICAÇÃO E VALIDAÇÃO

Este capítulo é dividido em relação aos processos de verificação e validação dos dois códigos IMERSPEC, desenvolvidos com técnicas de paralelização distintos. Em relação ao primeiro código, que emprega a paralelização MPI em fortran, é apresentada a sua verificação numérica nas diferentes junções do MFI/MPF e a validação da modelagem do escoamento laminar sobre degrau. Na segunda parte do capítulo, são explorados os resultados de verificação e validação relacionados ao código desenvolvido em CUDA C/C++ com a paralelização heterogênea CPU/GPU para os mesmos casos previamente testados. Os resultados são apresentados e comparados com os anteriores, relativos à paralelização MPI. Adicionalmente, ainda foi testada a nova metodologia MIED/MFI para uma fronteira imersa não coincidente em uma tentativa de tornar o processo de IBM também espectral.

5.1 Paralelização MPI

5.1.1 Verificação MPF

A confiabilidade de um código numérico-computacional é diretamente dependente de duas etapas fundamentais: a sua verificação e sua validação. A verificação é o processo que estabelece a exatidão matemática do modelo computacional com base em uma referência enquanto a validação é o processo de determinação do quão próximo os resultados numérico-computacionais conseguem representar dados empíricos da perspectiva de uso do código desenvolvido⁹².

Consequentemente, a verificação deve estabelecer que o modelo numérico-computacional para um conjunto de equações diferenciais parciais com suas condições de contorno convergem para determinada solução analítica de acordo com uma taxa de convergência assintótica determinada pelas discretizações temporal e espacial utilizadas no esquema numérico. Deve-se sempre realizar a verificação numérica antes da validação física caso contrário não se pode distinguir se o desacordo dos resultados numérico-computacionais obtidos com resultados materiais é fruto de erros fundamentais da modelagem ou é decorrente de um simples erro de implementação dentro do código desenvolvido, uma incorreta discretização ou ainda uma convergência incorreta para o esquema numérico utilizado⁹³.

Assim, a primeira etapa de confiabilidade do código desenvolvido é a verificação do IMERSPEC MPI. Nessa subseção é verificada apenas o MPF, a verificação da abordagem junto ao MFI serão exploradas nas subseções 5.1.2 e 5.1.3. O processo em si de verificação do código consiste na averiguação da correta implementação dos modelos matemáticos propostos no código

computacional e na identificação de possíveis erros no software. Técnicas para a verificação do código incluem o uso de soluções contínuas e o Método de Soluções Manufaturadas (MSM)⁹².

O MSM é uma abordagem para a criação de soluções exatas que consiste na “manufatura” dessas soluções através de ligeiras modificações nas equações de balanço. Devido ao propósito da verificação numérica, não é requerido que a solução manufaturada modele um problema físico real. O procedimento geral do método consiste na escolha da solução *a priori* e então a sua aplicação sobre as equações de balanço consideradas, gerando assim termos fontes contínuos⁹⁴. Desse modo, a solução manufaturada passa a ser uma solução exata do conjunto de equações de balanço modificadas, formadas pelo conjunto de equações originais mais os termos fontes contínuos.

Destaca-se que o MMS envolve a solução de um problema inverso, *i.e.*, dado um conjunto de equações de balanço e uma solução contínua, encontra-se o correspondente conjunto de equações modificadas que satisfaz a solução proposta⁹⁴.

Assim como a verificação de Moreira²³, foram utilizadas as soluções contínuas propostas por Henshaw⁹⁵ para a verificação do IMERSPEC MPI em que u^c , v^c , w^c e p^c são iguais a:

$$u^c(x, y, z, t) = \text{sen}(x) \cos(y) \cos(z) \cos(2\pi t); \quad (5.1a)$$

$$v^c(x, y, z, t) = \cos(x) \text{sen}(y) \cos(z) \cos(2\pi t); \quad (5.1b)$$

$$w^c(x, y, z, t) = -2 \cos(x) \cos(y) \text{sen}(z) \cos(2\pi t); \quad (5.1c)$$

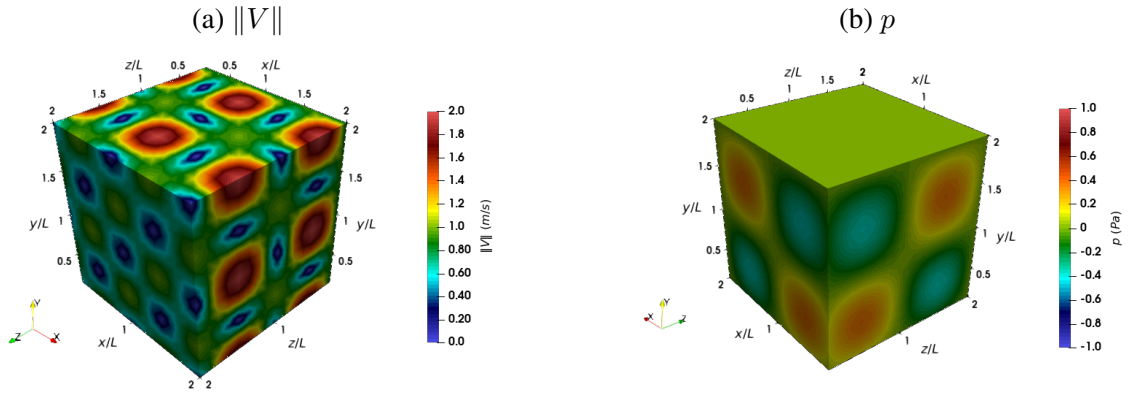
$$p^c(x, y, z, t) = \text{sen}(x) \text{sen}(y) \text{sen}(z) \cos(2\pi t). \quad (5.1d)$$

Salienta-se que a pressão dentro do IMERSPEC é uma variável de pós-processamento e, portanto, p^c só é utilizada na verificação da solução para a pressão. Outro destaque a ser mencionado é sobre os termos forçantes envolvidos na solução manufaturada. Como a simulação é tridimensional os termos forçantes para cada um dos componentes de velocidade são extensos e por isso não são apresentados explicitamente na presente tese, entretanto, ressalta-se que eles foram obtidos a partir das Eq. 3.2. A verificação da solução pode ser definida como o processo de determinação da exatidão de uma solução numérico-computacional específica em relação a solução exata do modelo computacional⁹². Nesta tese, a norma euleriana L_2^Ω , Eq. 5.2,

$$L_2^\Omega = \sqrt{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \sum_{k=1}^{N_z} \frac{(\phi_{i,j,k}^n - \phi_{i,j,k}^c)^2}{N_x N_y N_z}}, \quad (5.2)$$

é utilizada para mensurar a exatidão entre um componente numérico-computacional, ϕ^n , e o respectivo componente da solução contínua, ϕ^c , em todo o domínio computacional Ω . A Figura 21a apresenta a condição inicial da magnitude do vetor velocidade, $\|V\|$, obtida a partir das Eq. 5.1a, 5.1b e 5.1c enquanto a pressão, Eq. 5.1d, é ilustrada na Fig. 21b.

Figura 21 – Condições iniciais para a verificação numérica.



Fonte: Próprio autor.

Os parâmetros de simulação empregados na verificação são apresentados na Tab. 3 onde $Re = \frac{U_\infty L}{\nu}$ em que $L = \pi [m]$ e $U_\infty = 1,0 m/s$ é a velocidade característica da solução contínua. Destaca-se que a verificação numérica é feita com um passo de tempo físico constante.

Tabela 3 – Propriedades da Simulação Relacionada a Verificação MPF.

$L_x \times L_y \times L_z$	$N_x \times N_y \times N_z$	$\Delta t(s)$	$t_f(s)$	$U_\infty(m/s)$	Re
$2\pi \times 2\pi \times 2\pi$	$16 \times 16 \times 16$	10^{-4}	10	1	10

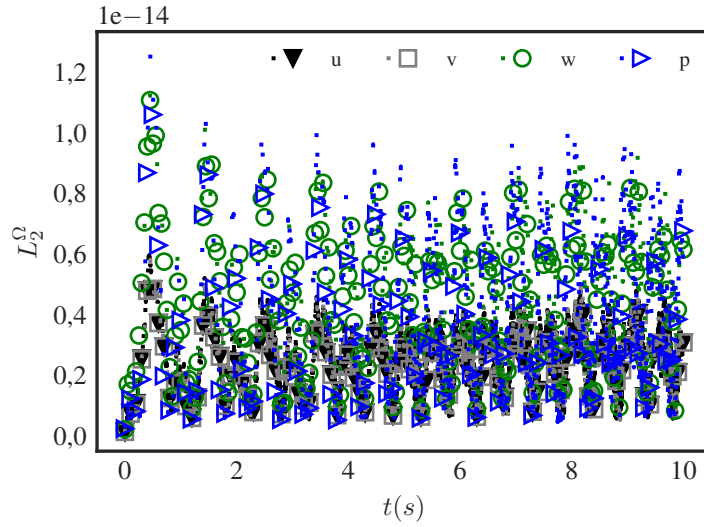
Fonte: Próprio Autor.

Com os parâmetros da Tab. 3, realizou-se a verificação da solução numérico-computacional utilizando a Eq. 5.2 para a variável pós-processamento p e as três componentes de velocidade u , v e w . As curvas das normas ao longo do tempo físico modelado são exibidas na Figura 22 onde nota-se que todas as variáveis se mantiveram com a exatidão de erro de máquina durante toda a simulação com destaque especial as normas L_2 para os componentes u e v que se mostram iguais. Atesta-se, portanto, a alta acurácia esperada de uma abordagem espectral como o MPF mesmo com uma malha grosseira de 16^3 pontos de colocação.

Após examinar a exatidão do código desenvolvido, foi analisada a eficiência da paralelização utilizada no IMERPSEC MPI ao mesmo tempo que se comparou as duas versões da FFTE disponíveis. Para mensurar a paralelização do código, é utilizada a definição de *speedup*, sp , conforme Eq. 5.3,

$$sp(p) = \frac{\text{Custo Computacional 1 processo}}{\text{Custo Computacional } p \text{ processos}}, \quad (5.3)$$

em que p é o número de processos considerados. Esse adimensional avalia quantas vezes mais rápido o código computacional se tornou com a sua paralelização em p processos e é limitado a um valor teórico considerando um aumento linear de acordo com a quantidade de p núcleos. Esse valor é teórico porque um código nunca é inteiramente paralelo, *i.e.*, sempre haverá porções dentro de qualquer algoritmo que devem ser executadas em *serial*.

Figura 22 – Curva de L_2 para u , v , w e p na verificação de MPF.

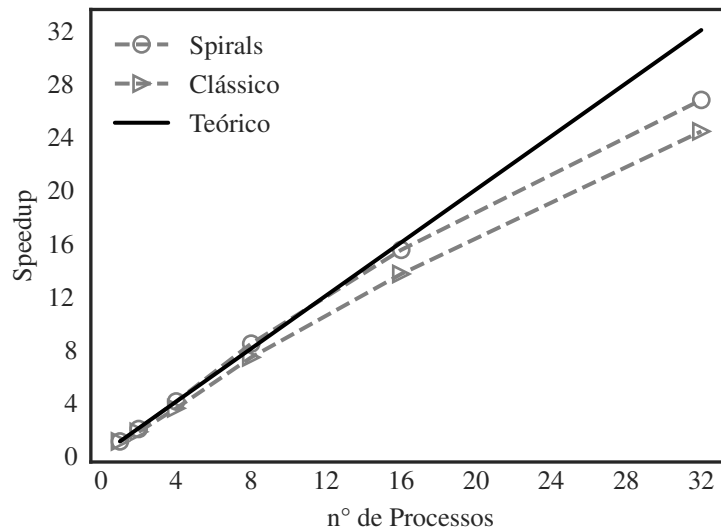
Fonte: Próprio autor.

A fim de esboçar o perfil de *speedup* por núcleos para o IMERSPEC MPI considerou-se uma malha de 64^3 pontos de colocação. A malha numérica considerada até então, 16^3 pontos de colocação, tornaria a quantidade de pontos de colocação em cada processo pequena ou insuficiente quando executada para, por exemplo, 32 núcleos.

O gráfico do *speedup* IMERSPEC MPI considerando apenas o MPF para as duas versões da FFTE, a utilizada até então na metodologia IMERSPEC e a versão otimizada usando *spirals*, foi criado a partir de simulações numérico-computacionais realizadas no cluster do LaMCAD/UFG⁹⁶ em que todo o ambiente está interconectado através de uma rede Infiniband de 100 Gbps. O Gráfico do *speedup* é visto na Fig. 23. Destaca-se que até 16 processos foi utilizado apenas um único núcleo e a partir de 32 processos que se tem dependência da rede Infiniband e esta configuração é mantida para todos os resultados ao longo presente capítulo.

Analizando o gráfico da Fig. 23 é notável o quão paralelizável é o código desenvolvido quando utilizado somente o MPF. Nota-se que ambas as versões da FFTE estão sobrepostas a curva teórica do *speedup* até 8 processos. A partir de 8 processos, para essa malha de 64^3 pontos de colocação, a comunicação entre os processos passa a impactar mais o custo computacional, afastando assim o comportamento do *speedup* medido em relação ao seu valor puramente teórico.

Ainda sim, verifica-se que a versão da FFT otimizada por *spirals* se mantém mais próxima a curva teórica para a maior quantidade de processos. Considerando este fato e que a exatidão apresentada na Fig. 22 se mantém para as duas versões da biblioteca, passou-se a empregar, unicamente, em posteriores simulações do IMERSPEC MPI, a nova versão da FFTE desenvolvida por Takahashi; Franchetti⁹⁰.

Figura 23 – *Speedup* de diferentes versões da FFTE para a Verificação do MPF.

Fonte: Próprio autor.

5.1.2 Verificação MFI Coincidente

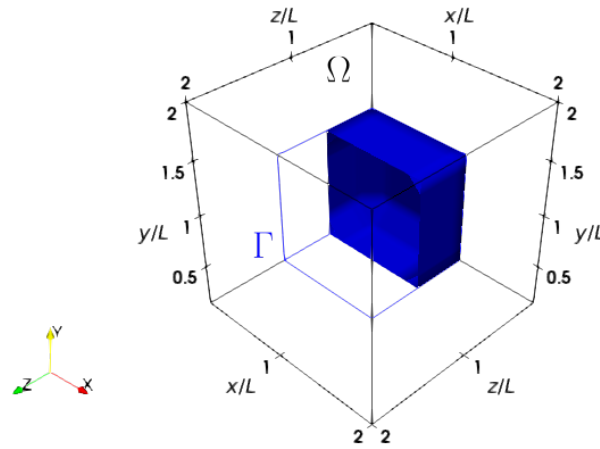
Verificado o código em relação ao MPF envolvido na solução numérica das Eq. 3.1 e 3.2, busca-se também verificar o MDFM/MFI, responsável pela implementação de condições de contorno dentro do escoamento modelado. A metodologia IMERSPEC apresenta dois tipos gerais de condições de contorno: as impostas com a coincidência de Ω e Γ e àquelas implementadas sem a coincidência desses domínios. Essa subseção explorará a verificação numérica do primeiro tipo de condição de contorno enquanto a subseção 5.1.3 verificará o outro tipo de implementação.

Dessa forma, para realizar a verificação proposta é implementado uma casca na forma de um cubo com lado, L , igual a $\pi [m]$ centralizado no domínio computacional Ω . Nessa casca são forçadas, através do MDFM/MFI, os componentes da velocidade provenientes das Eq. 5.1a, 5.1b e 5.1c em cada incremento computacional. Assim, o código utiliza o MPF para a resolução das equações de balanço, Eq. 3.1 e Eq. 3.2, enquanto implementa as condições de contorno na malha Ω uniforme através do MDFM/MFI. Na Figura 24 é ilustrado o esquema de implementação descrito em que o domínio euleriano corresponde ao cubo de lado $2L$ e o domínio lagrangiano (geometria imersa) corresponde a casca do cubo de lado L em azul.

Considerando os mesmos parâmetros de simulação da Tab. 3, tem-se a verificação da solução numérica ilustradas pela Fig. 25. A simulação realizada possui dois tipos de norma L_2 : a L_2^Ω , calculada em todo o domínio euleriano, e a L_2^Γ obtida a partir da norma calculada exclusivamente na casca lagrangiana.

Analisando às Fig. 25a e 25b é notável que a exatidão da abordagem espectral ainda é perceptível já que ambas as normas para todos os componentes da velocidade e para a variável de pós-processamento se mantém na ordem de erro de máquina para uma malha relativamente

Figura 24 – Esquema de implementação.



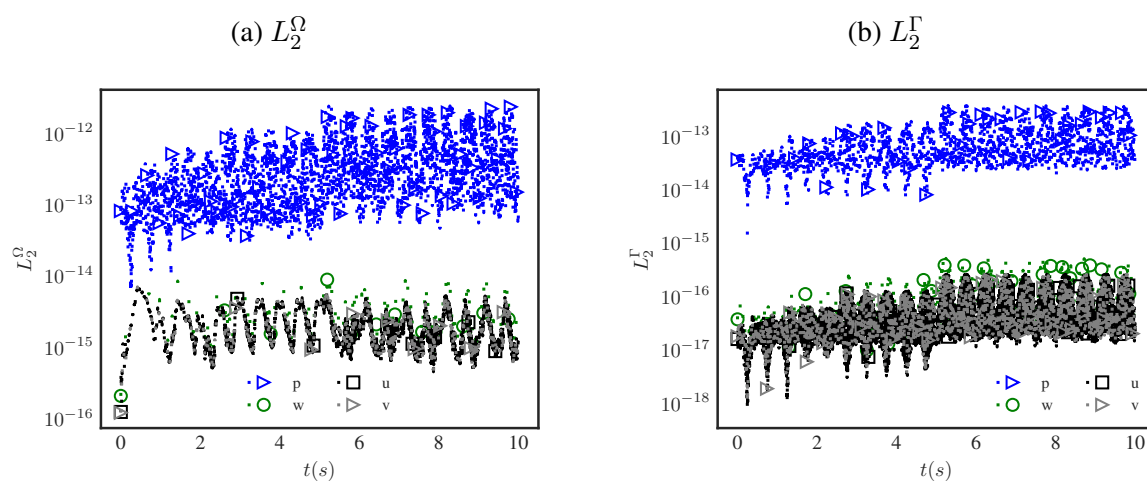
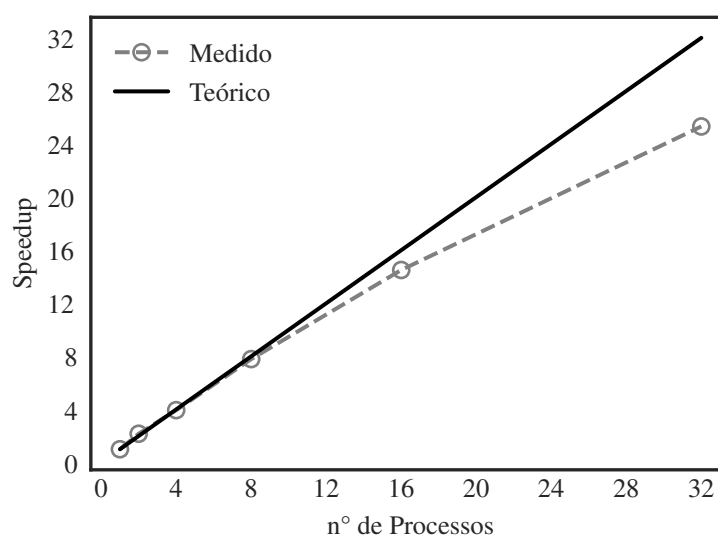
Fonte: Próprio autor.

grosseira de 16^3 pontos de colocação. Comparando a Fig. 25a com a Fig. 22, percebe-se que as normas das componentes tiveram uma ligeira melhora em relação a simulação sem a fronteira imersa enquanto a norma da pressão teve uma ligeira piora.

Examinando a Fig. 25b percebe-se que a norma lagrangiana é mais acurada para todas as variáveis do que as suas respectivas contrapartes eulerianas enquanto nota-se um aumento do valor das normas a medida que t também aumenta. Esse aumento também é visto em menor escala para a L_2^Ω da pressão e se deve ao erro numérico de cancelamento catastrófico envolvido na subtração de $\phi^n - \phi^c$ na Eq. 5.2.

O cancelamento catastrófico acontece quando dois números muito próximos são subtraídos e a maioria dos dígitos do resultado são perdidos⁹⁷. Esclarece-se que a operação de subtração não se torna menos precisa pela proximidade entre os dois números, essa operação numérica é exata, mas sim que esta amplifica erros numéricos pré-existentes formados a partir dos arredondamentos realizados anteriormente no cálculo das variáveis ϕ^n e ϕ^c . Conforme já dito, esse erro numérico acontece no cálculo da norma e, portanto, não impacta negativamente a resolução numérico-computacional da metodologia IMERSPEC além de que esse erro indica a grande aproximação entre os resultados da simulação e os da solução contínua.

A partir de uma nova malha numérica de 64^3 pontos de colocação e os demais parâmetros da Tab. 3 é levantada a curva de *speedup* do código IMERSPEC MPI para a execução do MPF junto ao MDFM/MFI com Γ e Ω coincidentes. O perfil de *speedup* levantado pelos processos simulados no cluster LaMCAD/UFG é visto na Fig. 26 junto ao seu perfil teórico, o qual considera um *speedup* linear. Comparando-se a Fig. 23 junto a Fig. 26 tem-se, qualitativamente, o mesmo perfil para os dois tipos de simulações, *i.e.*, o emprego do método MDFM/MFI com fronteira imersa coincidente ao domínio euleriano não prejudica a paralelização do código utilizando MPI.

Figura 25 – Verificação da solução numérica com malha de 16^3 pontos de colocação.Figura 26 – *Speedup* para verificação do MPF com MDFM/MFI coincidente.

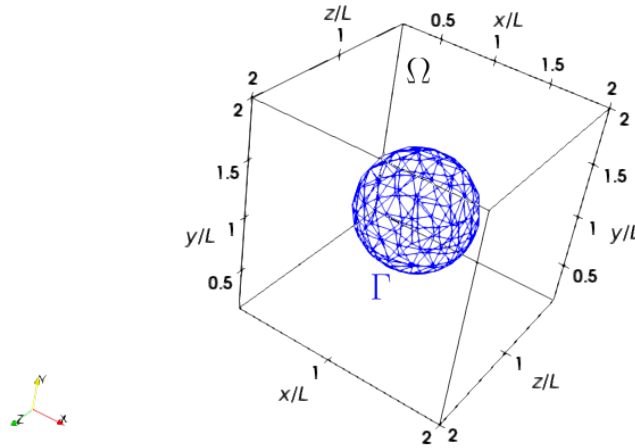
Assim, essa classe de fronteira imersa pode ser utilizada em modelagens sem qualquer desvantagem na exatidão da metodologia ou na execução em paralelo do código computacional.

5.1.3 Verificação MFI Não Coincidente

Após a verificação do IMERSPEC MPI com a fronteira imersa coincidente, é feita a verificação do último tipo de condição de contorno a ser imposta pelo MDFM/MFI: a fronteira imersa da malha Γ não coincidente com a malha uniforme Ω . Dessa forma, a malha euleriana é sempre uniforme, condição exigida pelo uso da biblioteca FFTE, enquanto a malha lagrangiana passa a ser uma malha não uniforme.

A malha responsável pelos pontos Γ , malha não uniforme, é feita externamente ao IMERSPEC MPI utilizando o *software* livre SALOME com o *plugin* GMSH. Essa malha também é uma casca similar ao cubo da subseção 5.1.2 entretanto, agora, possui formato de uma esfera de diâmetro π [m] a qual está centralizada no domínio euleriano, conforme apresentado na Fig. 27. A malha lagrangiana foi definida através do algoritmo *frontal Delaunay* com $\Delta s \approx \Delta x$ para cada malha Ω simulada.

Figura 27 – Esquema de implementação.



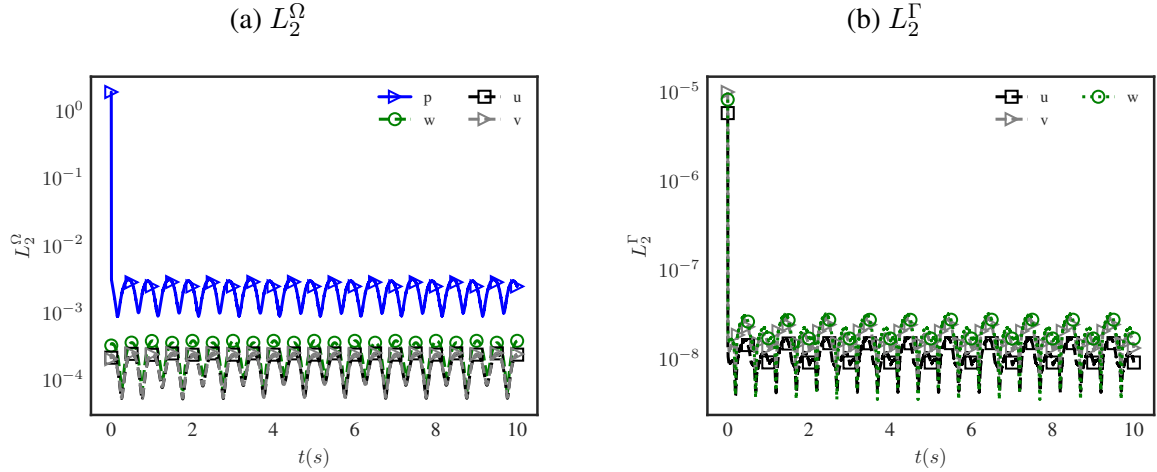
Fonte: Próprio autor.

A fim de analisar a exatidão da fronteira imersa não coincidente dentro do IMERSPEC MPI, foi feita uma simulação com os parâmetros da Tab. 3 junto ao critério de parada $\epsilon = 10^{-3}$ do MDFM. A verificação da solução para a malha de 16^3 pontos de colocação é vista na Fig. 28a para L_{Ω}^2 e na Fig. 28b para L_2^{Γ} .

Analisando as normas eulerianas, Fig. 28a, nota-se o comportamento oscilativo ao longo do tempo físico esperado pela presença do termo $\cos(2\pi t)$ no conjunto de Eq. 5.1 e, acima de tudo, nota-se a perda de exatidão causada pela fronteira imersa não coincidente. Para a fronteira imersa coincidente, subseção 5.1.2, tinha-se as normas com precisão de erro de máquina enquanto

agora as mesmas oscilam em torno da ordem de 10^{-4} para os componentes da velocidade e de $5 \cdot 10^{-2}$ para a pressão. Em relação às normas Γ , Fig. 28b, tem-se as mesmas características apontadas nas normas Ω , entretanto, percebe-se uma maior exatidão dessas normas em relação às suas contrapartes eulerianas.

Figura 28 – Verificação da solução numérica com malha de 16^3 pontos de colocação.

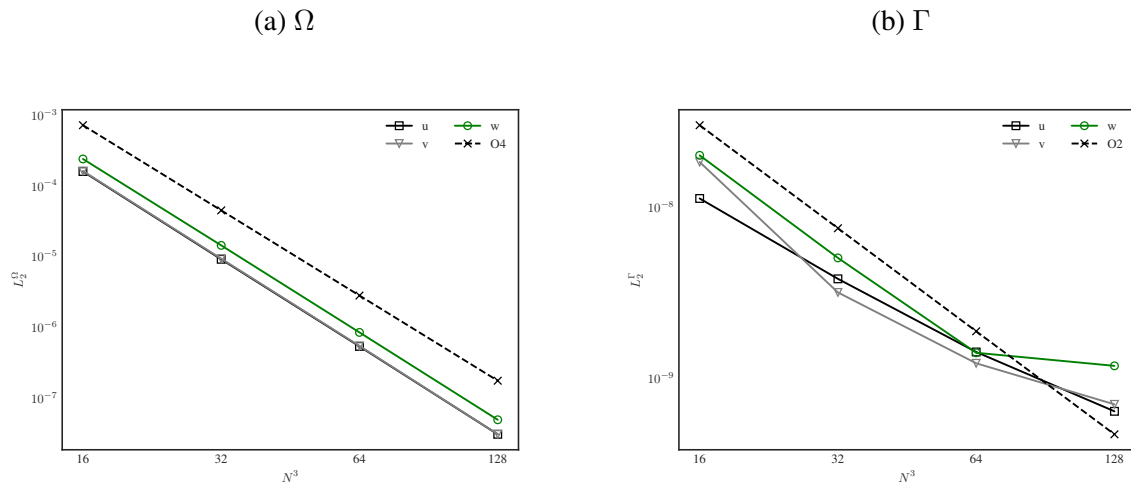


Fonte: Próprio autor.

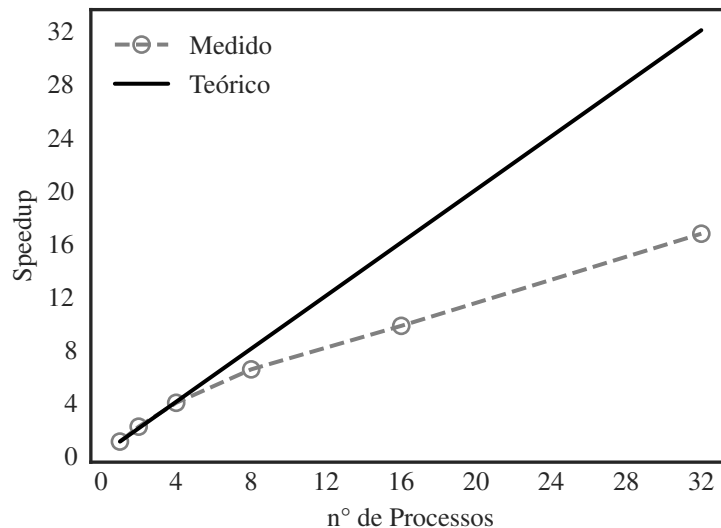
Como as normas L_2^Ω e as L_2^Γ não apresentam exatidão de erro de máquina para a malha mais grosseira simulada, 16^3 , é possível analisar a convergência da metodologia IMERSPEC com fronteira imersa não coincidente a partir de uma curva de convergência. Para criar a curva de convergência realiza-se a média de cada uma das normas L_2 ao longo do tempo físico simulado para diferentes malhas numéricas. As curvas de convergência das normas L_2^Ω para os componentes de velocidade são apresentadas na Fig. 29a, onde verifica-se que as normas eulerianas exibem convergência de quarta ordem.

Em relação as normas L_2^Γ , Fig. 29b, observa-se que as curvas do componente das velocidades apresentam, aproximadamente, segunda ordem de convergência até a malha com 64^3 . Com mais um refinamento de malha, as normas parecem atingir um plateau e as médias das normas da malha com 128^3 pontos de colocação possuem mesma ordem de grandeza que as suas equivalentes da malha anterior. No trabalho de Mariano⁴² é provado que a função interpoladora é a responsável principal pela limitação da exatidão da metodologia IMERSPEC junto a fronteira imersas não coincidentes. Dessa forma, fica claro o grau de limitação imposto pela função interpoladora cúbica quando nota-se a ordem de convergência obtida para as normas L_2 no domínio Γ mesmo que tal função seja a alternativa que resulte em maior acurácia. Mesmo assim, a metodologia IMERSPEC com a aplicação de fronteiras imersas não coincidentes ainda permanece sendo uma metodologia de alta ordem visto que esta possui quarta ordem de convergência.

Após a análise da convergência da metodologia IMERSPEC para malhas Γ não coincidentes, investiga-se o *speedup* do IMERSPEC MPI para uma malha com 64^3 pontos de colocação.

Figura 29 – Análise da Convergência do IMERSPEC MPI para Γ não coincidente.

A Figura 30 apresenta o perfil de *speedup* construído a partir da execução da mesma malha com diferentes números de processos em que nota-se significativo afastamento do perfil medido em relação ao perfil teórico a partir da execução do código com 8 núcleos. Esse comportamento não é visto para os perfis de *speedup* realizados nas verificações anteriores, Fig. 23 e 26, o que sugere que a limitação é decorrente da implementação do MDFM/MFI com pontos Γ não coincidentes.

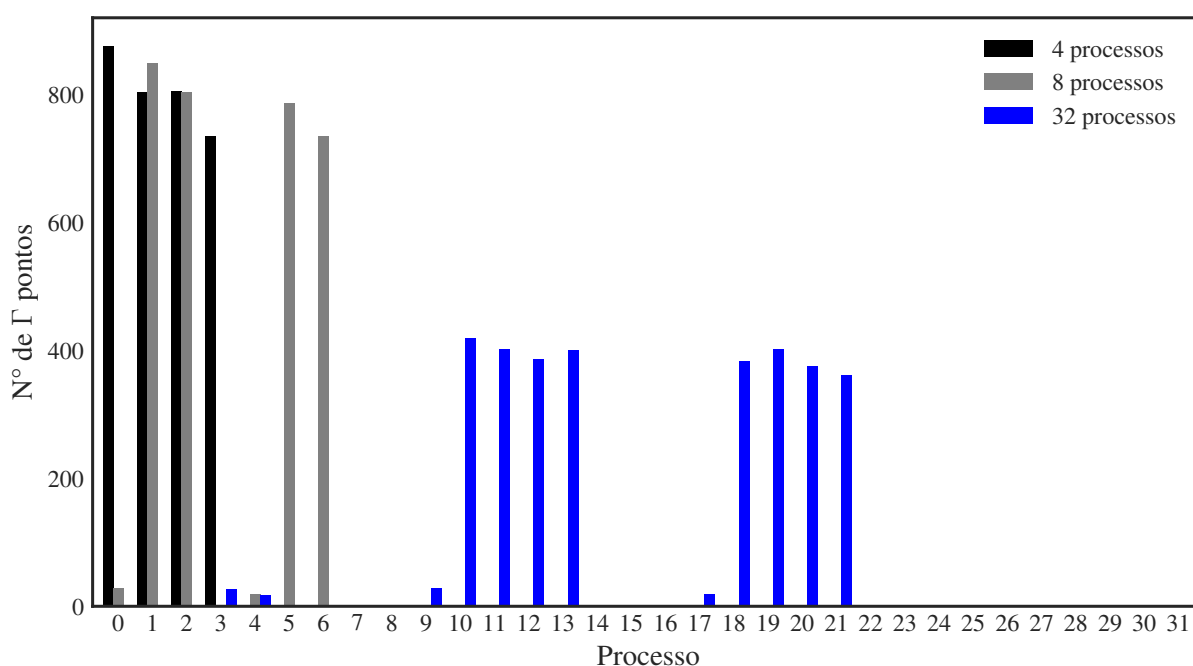
Figura 30 – *Speedup* para Verificação do MPF com MDFM/MFI não coincidente.

A explicação para a perda de eficiência do *speedup* para maior quantidade de processos está relacionada a distribuição dos pontos lagrangianos por processos nas simulações. A distribuição de pontos Γ por processos em 3 diferentes cenários é ilustrada pela Fig. 31 onde nota-se que, por exemplo, para a execução do código com 32 processos, há um desbalanço significativo na distribuição de pontos lagrangianos por núcleos.

A metodologia de paralelização usando o MPI utiliza processos independentes entre si, entretanto, conforme descrito na subseção 4.1.2, o processo de interpolação que utiliza a função cúbica necessita das informações da vizinhança euleriana do ponto Γ a ser calculado. Assim, pontos lagrangianos presentes nas “fronteiras” de processos independentes necessitam de informações contidas em outros processos, *i.e.*, é necessário uma forma de comunicação entre os processos antes da operação de interpolação.

Em termos de topologia virtual da paralelização MPI, realiza-se uma expansão das submatrizes dos componentes de velocidades. Analogamente, para o processo de distribuição torna-se necessária uma redução das submatrizes anteriormente expandidas para que estas voltem ao seu tamanho original. A fim de fazer os processos de expansão e redução, processos vizinhos enviam e recebem valores de velocidades das fronteiras. Uma das exigências dentro da paralelização MPI é a de que toda transferência de dados entre processos tenha um envio e recebimento explícito.

Figura 31 – Distribuição de Γ pontos de acordo com 4,8 e 32 processos.



Fonte: Próprio autor.

Assim, volta-se ao cenário apresentado pela Fig. 31 com 32 processos, primeiro ocorre a expansão das submatrizes dos componentes das velocidade em todos os núcleos com a troca de informações necessária por todos os processos. Após, são realizados os passos do MDFM/MFI apresentados na Fig. 10, todavia, o método é aplicado apenas em processos que possuem pontos lagrangianos. Logo, àqueles os quais não possuem tais pontos tem de aguardar o final do algoritmo MDFM/MFI para a posterior troca de informações entre todos os processos para a redução das submatrizes previamente expandidas.

A distribuição dos pontos Γ por processos é realizada seguindo a topologia virtual da

paralelização, ou seja, são distribuídos de acordo com sua posição espacial que esta contida na porção do domínio euleriano particionado. Sendo assim, a distribuição desigual e progressiva dos pontos lagrangianos entre os processos para a execução em paralelo com 8 e depois com 32 processos, Fig. 31, é esperada visto que a posição da casca esférica lagrangiana é posicionada apenas no centro do domínio Ω , Fig. 27.

Por meio das Fig. 30 e 31 constata-se que à medida que a distribuição de pontos Γ por núcleos se torna mais desigual, mais o *speedup* obtido pelo código IMERSPEC MPI se desvia do perfil teórico.

5.1.4 Escoamento Laminar sobre um Degrau

Realizada a verificação do IMERSPEC MPI, utiliza-se o mesmo na validação através da modelagem do clássico caso do escoamento sobre degrau. Para essa modelagem, as paredes superiores e inferiores do escoamento foram implementadas em formato de “casca”, *i.e.*, duas placas ocas com espessura igual a $3\%L_y$. Destaca-se, mais uma vez, que as condições de não deslizamento impostas nessas cascas foram realizadas por MFI com domínios Γ e Ω coincidentes e $U_i^{IM} = 0$ implementados seguindo o fluxograma da Fig. 9.

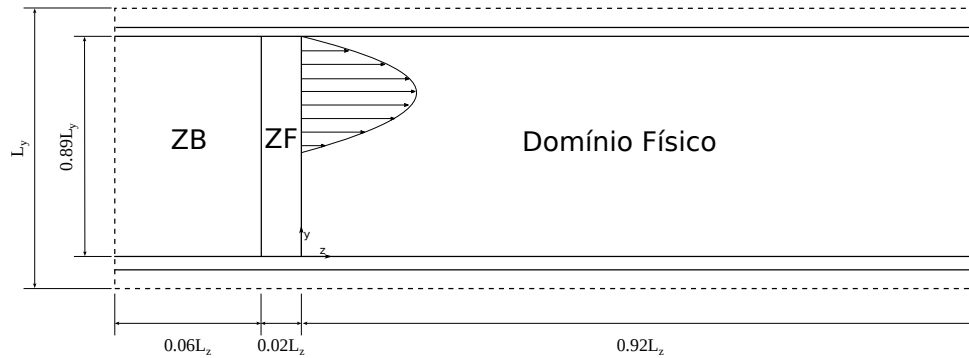
O artifício de “casca” é realizado a fim de implementar a condição de contorno mais suave e evitar variação abrupta de valores dentro do escoamento devido ao uso das FFT na resolução numérico-computacional^{98,62}. Parte do domínio acima e abaixo das placas não é modelado a fim de se preservar as condições de contorno periódicas que o escoamento deve ter nessa direção, também, devido ao emprego da FFT no IMERSPEC MPI. Na terceira direção do escoamento são utilizadas apenas as condições periódicas inerentes ao código. Para essa modelagem, essas condições são desejáveis.

A Figura 32 ilustra um corte em x do esquema de implementação do caso simulado em que as linhas tracejadas representam onde são aplicadas as condições de contorno na direção y e z . Destaca-se ainda que o esquema apresenta o domínio complementar composto pela zona de *buffer* e forçagem utilizadas na periodização artificial do escoamento modelado e na reformulação da condição de entrada do escoamento no domínio físico. Em relação a condição de entrada do domínio físico, tem-se que as velocidades u e v são nulas enquanto o perfil de entrada em w é apresentado pela Eq. 5.4,

$$w(y) = \begin{cases} 0 & \text{se } 0 \leq y < h \\ -24(2h - y)(h - y) & \text{se } h \leq y \leq 2h \end{cases}. \quad (5.4)$$

A modelagem desse escoamento complexo permite analisar a influência da forma de discretização do termo difusivo das Eq. 3.2, TF , na modelagem de escoamentos parietais. Sendo assim, a primeira análise em relação a essa modelagem é a comparação da simulação realizada com a discretização divergente do termo difusivo, utilizada até então em abordagens 3D do

Figura 32 – Esquema de implementação.



Fonte: Próprio autor.

IMERSPEC, e a discretização antissimétrica proposta no presente trabalho. Para esse estudo utilizou-se as propriedades de simulação apresentadas pela Tab. 4 onde L_x , L_y e L_z são o tamanho do domínio computacional; CFL é a condição de estabilidade Courant-Friedrichs-Lewis; t_a é o tempo adimensional igual a tU_∞/h em que U_∞ é a velocidade média da entrada do domínio útil e h é a altura do degrau; Re representa o número de Reynolds e é igual a $U_\infty h/\nu$; e ϵ é o critério de parada do MDFM.

Tabela 4 – Propriedades das simulações do escoamento sobre degrau.

$L_x \times L_y \times L_z$	CFL	t_a	Re	$U_\infty (m/s)$	ϵ	$h (m)$
$4\Delta_y \times 2,25h \times 72h$	0,5	500	400	1	10^{-3}	0,5

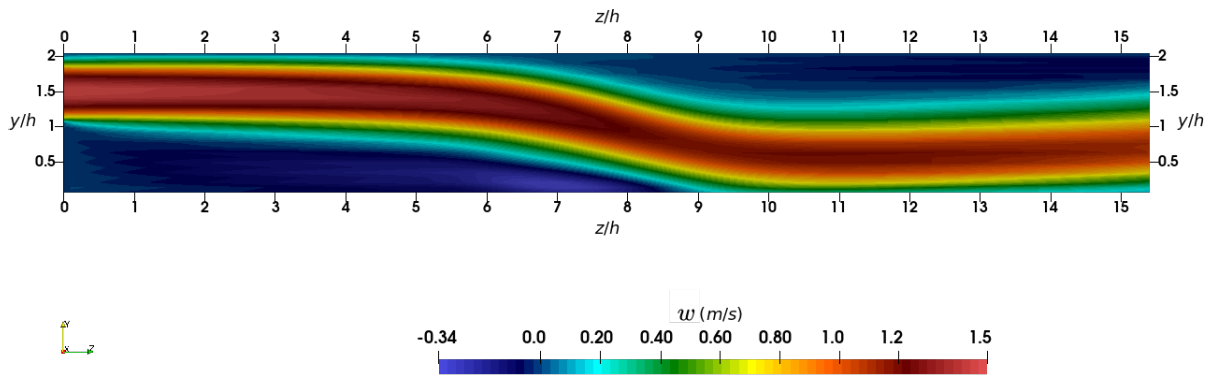
Fonte: Próprio Autor.

Inicialmente, realizou-se a comparação entre as discretizações de TF apenas com a malha de $4 \times 32 \times 1024$. Realizada a simulação para TF com discretização divergente, notou-se ondas zonais, em específico, ondas $2h$ perpendiculares às paredes presentes no componente de velocidade w na modelagem. A Fig. 33 ilustra uma porção de um plano yz de w para a simulação com a discretização TF divergente onde nota-se oscilações ao longo de toda região do domínio físico.

A Figura 34 apresenta em específico o perfil vertical de velocidade w em $z = 14h$ e $t_a = 100$ do plano da Fig. 33 para a malha de $4 \times 32 \times 1024$ pontos de colocação assim como perfis de velocidade para as malhas de $4 \times 64 \times 2048$ e $4 \times 128 \times 4096$ pontos de colocação sob a mesma discretização de TF e o mesmo perfil com a malha mais grosseira, $4 \times 32 \times 1024$ pontos de colocação, com a discretização antissimétrica de TF . Destaca-se que embora as dimensões L_x , L_y e L_z do escoamento apresentem diferentes tamanhos, os Δ eulerianos são iguais em todas as malhas, *i.e.*, $\Delta_x = \Delta_y = \Delta_z$.

As oscilações do perfil de w para a malha $4 \times 32 \times 1024$ não são tão significativas, assim, pode-se considerar que as mesmas sejam apenas uma consequência da robustez da malha e sucessivos refinamentos sanariam o efeito numérico. Essa hipótese se mostra incorreta visto que os sucessivos refinamentos apenas deixam mais claro que o fenômeno se refere ondas zonais visto

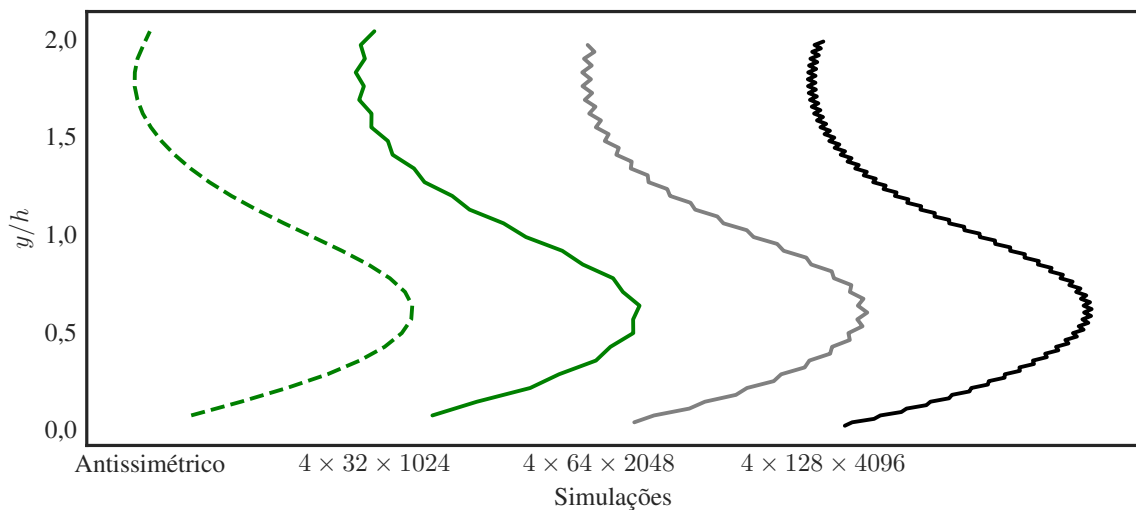
Figura 33 – Plano yz do domínio físico para a componente w e TF com discretização divergente em $t_a = 100$.



Fonte: Próprio autor.

que as oscilações apenas se apresentam na direção perpendicular às paredes e, ao mesmo tempo, são ondas $2h$ já que oscilam de um ponto discreto a outro com amplitude igual a $2h$ (dentro do IMERSPEC, $2\Delta y$). A proposta de discretização antissimétrica do termo difusivo se mostra uma melhor alternativa visto que não há formação de ondas $2h$ e assim esta discretização é adotada integralmente nas posteriores modelagens deste capítulo, ainda que essa discretização ainda não tinha sido testada na modelagem de escoamentos turbulentos neste momento da pesquisa.

Figura 34 – Perfis em $14h$ e $t_a = 100$ para TF antissimétrico com $4 \times 32 \times 1024$ pontos de colocação e para 3 malhas diferentes com TF divergente.



Fonte: Próprio autor.

Observando a Fig. 33 e 34, esperava-se que o código computacional divergisse visto que as ondas $2h$ são consideradas precursoras de divergência numérica através do fenômeno de *aliasing*, entretanto, as simulações ficaram restritas apenas às ondas $2h$. Neste momento

da pesquisa não havia sido identificado a origem das ondas $2h$ mas, indubitavelmente, nota-se que a forma antissimétrica conseguiu reduzir o efeito de *aliasing*. Entende-se que não ocorreu divergência numérica nem avançou nenhum problema de *aliasing* dentro da modelagem devido, principalmente, a discretização antissimétrica do termo advectivo das Eq. 3.2 que, caso fossem discretizados com alguma outra forma, poderiam amplificar as ondas em questão devido a sua natureza não linear. Além disso, o $Re_h = 400$ implica em um escoamento laminar, o que também limita o desenvolvimento de instabilidades e garante que os altos números de onda acima do limite de Nyquist, K , não tenham significativa energia.

Nota-se entretanto que para modelagens de escoamentos livres, a discretização divergente do termo difusivo pode ser empregada sem nenhuma desvantagem visto que Moreira²³ a utilizou satisfatoriamente na modelagem de jatos turbulentos livres em desenvolvimento espacial utilizando a metodologia IMERSPEC 3D. Com isso, tem-se o questionamento do porquê do surgimento das ondas $2h$ para modelagens com a presença de paredes no IMERSPEC. Isso será abordado no capítulo 6.

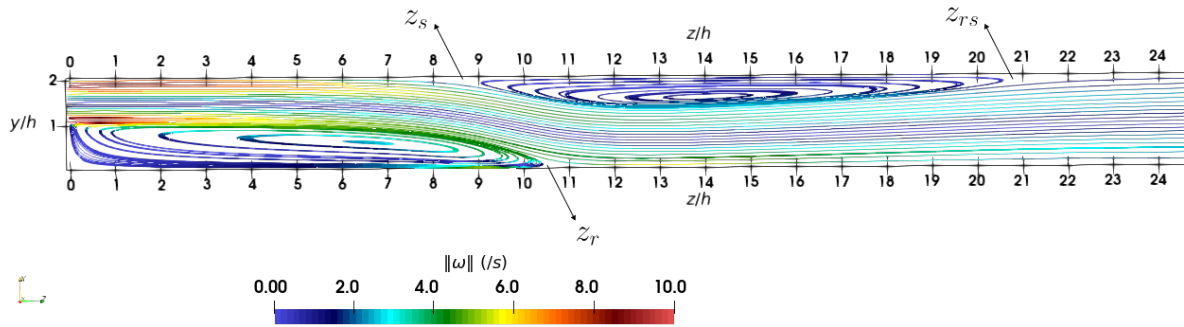
Definida a melhor discretização do termo difusivo para a presente modelagem, realizou-se uma análise de refinamento de malha da modelagem em questão. Por o escoamento modelado ser laminar, optou-se, inicialmente, por uma análise pseudo-3D a fim de priorizar a paralelização do código nas direções y e z . Sendo assim, utilizou-se três diferentes malhas com: $4 \times 32 \times 1024$, $4 \times 64 \times 2048$ e $4 \times 64 \times 4096$ pontos de colocação junto aos parâmetros de simulação vistos na Tab. 4.

Em relação ao escoamento laminar sobre degrau com $Re = 400$, há algumas características físicas importantes, definidas por pontos espaciais significativos a serem considerados na análise de convergência da malha computacional. A Figura 35 ilustra o escoamento modelado utilizando linhas de corrente coloridas pela magnitude da vorticidade $\|\omega\|$ em $t_a = 500$ para a malha mais grosseira, $4 \times 32 \times 1024$ pontos de colocação e para a formação dessas regiões de descolamento da camada limite na simulação com $4 \times 128 \times 4096$ pontos de colocação. Em relação a parede superior do domínio, observa-se que o escoamento começa colado junto a parede, posteriormente descola-se e, em seguida, recola-se novamente à parede superior. O ponto de descolamento é representado por z_s enquanto z_{rs} é utilizado para o ponto de recolamento.

Analisando a parede inferior do escoamento modelado, Fig. 35, nota-se que o escoamento entra descolado no domínio útil, devido justamente ao degrau, entretanto há um ponto significativo posteriormente de recolamento. Este ponto em questão é representado por z_r . Assim, nota-se que há regiões de descolamento do escoamento junto a parede inferior e a parede superior entretanto a origem do descolamento de cada região se difere. Enquanto a região inferior é formada devido ao degrau, a região superior tem sua formação devido ao recolamento da camada limite inferior.

Para validar o comportamento modelado pelas simulações, compara-se os resultados numérico-computacionais com dados materiais de Lee; Mateescu¹⁸. A comparação é feita através

Figura 35 – Linhas de corrente no plano yz para $t_a = 500$ para $4 \times 32 \times 1024$ pontos de colocação.



Fonte: Próprio autor.

dos perfis do componente w da velocidade na entrada do domínio físico, Fig.36a, em $14h$, Fig. 36b, e em $30h$, Fig. 36c. A partir das figuras, observa-se que os resultados da malha $4 \times 32 \times 1024$ pontos de colocação apresentam significativa divergência dos seus resultados com os dados materiais enquanto as sucessivas malhas já apresentam boa concordância.

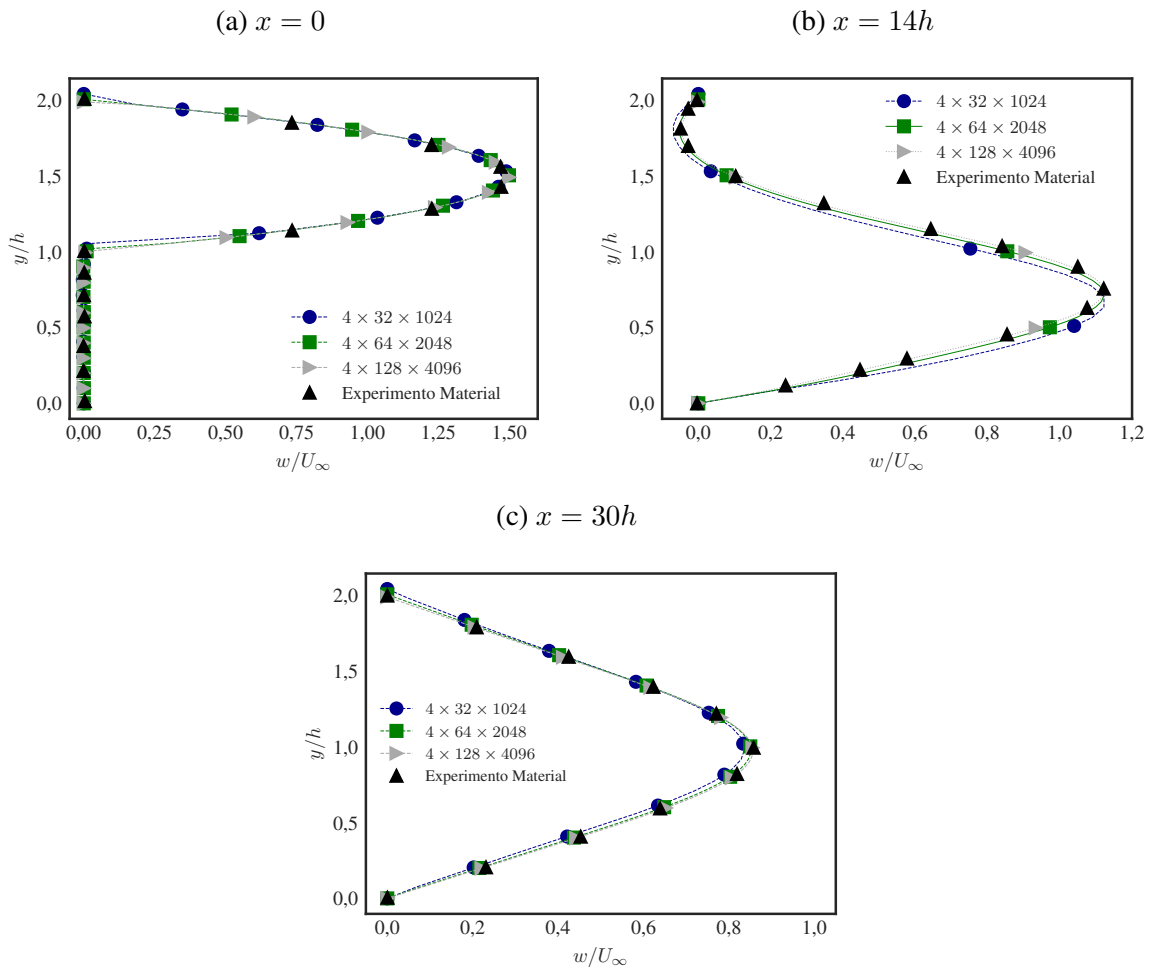
A Tabela 5 apresenta os valores obtidos das coordenadas z_r , z_s e z_{rs} junto aos custos computacionais de cada malha simulada e os resultados materiais obtidos por Lee; Mateescu¹⁸ para as coordenadas de interesse. Destaca-se que o custo computacional das simulações se refere a execução de cada malha com 64 processos na direção y e z . De maneira geral, nota-se que o refinamento de malha causa uma melhor aproximação dos resultados computacionais com os materiais com um aumento significativo dos custos computacionais. Comparando-se os resultados obtidos pela malha $4 \times 64 \times 128$ com os materiais, tem-se erros de 7%, 13,7% e 4,4% referentes, respectivamente, a z_r , z_s e z_{rs} com um custo computacional de, aproximadamente, 21 h .

A paralelização do IMERSPEC MPI ocorre apenas nas direções y e z , conforme mencionado. Dessa maneira, espera-se que uma malha totalmente 3D tenha um significativo aumento do custo computacional em relação aos apresentados na Tab. 5. Considerando este aspecto e os resultados provenientes das Fig. 36 e da Tab. 5, adota-se a malha com $4 \times 64 \times 2048$ pontos de colocação para ser estendida à uma malha 3D completa.

Tabela 5 – Coordenadas de interesse e custo computacional das simulações.

	z_r/h	z_s/h	z_{rs}/h	Custo Computacional
$4 \times 32 \times 1024$	10.73	7.68	22.32	3 h 9 min
$4 \times 64 \times 2048$	12.00	8.89	21.40	20 h 59 min
$4 \times 128 \times 4096$	12.39	9.34	21.06	8 d 15 h
Lee; Mateescu ¹⁸	12.90	10.30	20.50	-

Fonte: Próprio autor.

Figura 36 – Perfis de velocidade w/U_∞ .

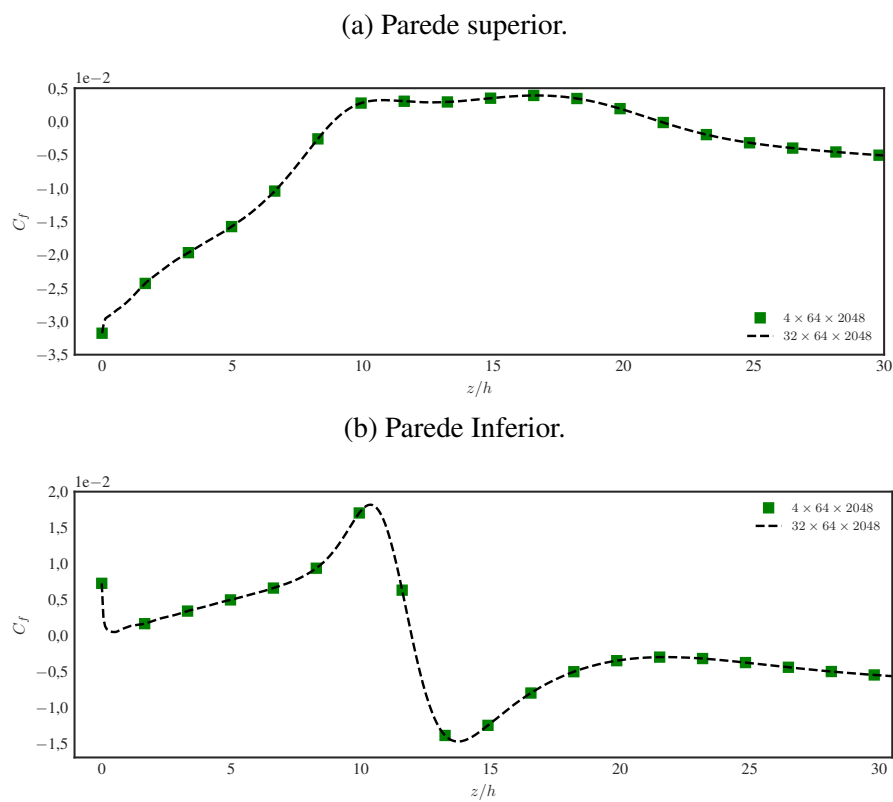
Fonte: Próprio autor.

A extensão da malha pseudo-3D para uma inteiramente 3D tem dois principais objetivos. O primeiro é detectar inexatidões na modelagem devido as condições periódicas utilizadas na direção x pelo uso da FFT e o segundo objetivo é mensurar o impacto da não paralelização da terceira dimensão do escoamento no custo computacional do IMERSPEC MPI. Assim, L_x foi estendido para $L_y/2$ resultando em uma malha 3D com $32 \times 64 \times 2048$ pontos de colocação enquanto os demais parâmetros da simulação se mantiveram iguais.

O custo computacional da malha tridimensional foi igual a 5 dias e 22 *min*, um aumento de aproximadamente 577% em relação à simulação pseudo 3D. Como a malha tridimensional é oito vezes maior do que a sua contraparte pseudo 3D, pode-se esperar, em uma primeira análise ingênua, um aumento proporcional de 800% no custo computacional. Entretanto, essa expectativa não tem fundamento teórico: o custo de uma FFT 3D escala como $N \log N$, e não linearmente com N , de modo que um aumento de oito vezes no tamanho da malha não implica um aumento de oito vezes no custo computacional. Além disso, a mudança no tamanho do problema altera a razão entre computação e comunicação, fator que também afeta o tempo de execução.

Com a intenção de detectar inexatidões causadas pela extensão da terceira dimensão do escoamento, analisa-se a Fig. 37a e 37b que, respectivamente, ilustram distribuição do coeficiente de atrito, C_f , das paredes superior e inferior ao longo de z no meio do comprimento de L_x para a simulação pseudo-3D e a 3D completa. Nota-se que as distribuições são equivalentes entre si para cada parede mesmo que haja mais pontos de colocação na terceira dimensão para a simulação 3D. Esses resultados provam que as condições periódicas modelam o escoamento laminar bem e não impactam negativamente na modelagem do fenômeno numérico-computacional já que os resultados obtidos para ambas as simulações foram equivalentes para as distribuições de C_f e também para as coordenadas z_r , z_s e z_{rs} .

Figura 37 – Comparação da distribuição de C_f para a simulação 3D e a pseudo-3D.



Fonte: Próprio autor.

Em suma, o código IMERSPEC MPI foi verificado atestando a sua exatidão a nível de erro de máquina para MFI coincidente e de quarta ordem para MFI não coincidente. Destaca-se a vulnerabilidade dessa técnica de paralelização para modelagens com Γ e Ω não coincidentes onde, dependendo da quantidade de processos envolvidos na paralelização, a performance da paralelização é comprometida por distribuição de pontos lagrangianos entre os processo. Adicionalmente, o algoritmo também foi validado para o *benchmark* clássico do escoamento laminar sobre um degrau. Atesta-se assim, que o código utilizando MPI apresenta significativa contribuição à metodologia IMERSPEC vide a opção sem paralelização alguma, mas ainda sim tem custo computacional significativo.

5.2 Paralelização CPU/GPU

Nesta seção será considerado o processo de verificação e validação do código computacional IMERSPEC CUDA. Antes de discorrer sobre os resultados obtidos para esse algoritmo, convém comentar de suas particularidades.

Assim como o IMERSPEC MPI, o objetivo principal do código desenvolvido é maximizar a performance da metodologia IMERSPEC entretanto, além do uso da paralelização heterogênea inerente do uso de CPU e GPU no mesmo código foram feitas algumas escolhas em relação a programação do novo código que merecem destaques. Essas escolhas são:

- O novo código foi escrito na linguagem CUDA C/C++;
- Os vetores computacionais das componentes u , v e w tridimensionais foram escritos de forma unidimensional e condensados em único vetor;
- Foram minimizadas completamente a passagem de informação do CPU para o GPU e vice-versa. Priorizou-se que todas as operações matemáticas sejam realizadas na memória GPU e a memória CPU seja usada para entrada/saída de dados do algoritmo;
- Utiliza-se, majoritariamente, blocos de *threads* unidimensionais. Blocos bidimensionais ou tridimensionais não seriam vantajosos para essa aplicação devido a metodologia MPF ser composta de operações globais ponto a ponto. Não há necessidade de comunicação com pontos de colocação vizinhos, o mapeamento dos pontos no *kernel* acontecem pelo índice global independentemente assim da dimensão do bloco de *threads*;
- O código MPI CUDA, conforme será discutido, foi operado em duas placas GPU distintas com performances discrepantes. Dessa forma, não se utilizou memória compartilhada nos *kernels* do código;
- Quando o número de *threads* totais definidos para a execução do código é inferior a malha computacional definida, $N_x \times N_y \times N_z$, cada *thread* processa múltiplos elementos via a estratégia *grid-stride loop* para garantir coalescência de memória.

Mesmo existindo a opção de programar o novo código utilizando CUDA Fortran foi escolhida a nova linguagem por esta ter compatibilidade direta com as novas bibliotecas relacionadas a GPUs e, portanto, não necessitarem da criação de interfaces.

Os componentes de velocidade foram vetorizados e condensados juntos para que as operações relacionadas a FFT sejam aplicadas de forma única, ainda que no contexto de GPU, elas sejam aplicadas de forma paralela, utilizando a quantidade de *threads* disponível na placa de vídeo.

Entende-se que em um código CUDA, um possível gargalo na performance do código desenvolvido é a a passagem de informação entre o CPU e o GPU ou vice-versa. Como uma

possível alternativa tem-se a memória unificada em que a informação é mantida tanto no CPU (*host*) quanto no GPU (*device*) entretanto códigos com esse tipo de memória tem performances inferiores àqueles em que mantêm-se as duas memórias dos equipamentos separadas. Portanto, salienta-se que esta estratégia não foi adotada no código desenvolvido.

Como o primeiro passo em um novo código computacional é sua verificação, as subseções 5.2.1, 5.2.2 e 5.2.3 se relacionam, respectivamente, com os resultados em termos de exatidão das seções 5.1.1, 5.1.2 e 5.1.3. Em seguida, o novo algoritmo também foi validado para o mesmo escoamento laminar, subseção 5.2.5, em relação a subseção 5.1.4.

Destaca-se que as simulações numérico-computacionais realizadas nessa seção, utilizando o código IMERSPEC CUDA ou o código IMERSPEC MPI, foram realizadas em um máquina equipada com um processador Intel Core i7-7700 (4 núcleos físicos e 8 *threads*, arquitetura Kaby Lake) e uma placa de vídeo NVIDIA GeForce GTX 1060 com 6 GB de memória GDDR5 (192-bit bus, 10 *Streaming Multiprocessors*, 1280 CUDA Cores e arquitetura Pascal). A fim de verificar a performance e a dependência do código IMERSPEC CUDA com a placa GPU usada, também foi utilizada a placa NVIDIA A100, disponível no *cluster* LaMCAD⁹⁶. A A100 é baseada na arquitetura Ampere e possui 108 *Streaming Multiprocessors* (SMs) habilitados. A placa conta com 80 GB de memória HBM2e, com largura de banda de até 2039 GB/s, valor substancialmente superior ao da GTX 1060 (GDDR5, 192-bit bus).

5.2.1 Verificação MPF

Assim como realizado na subseção 5.1.1, foram utilizadas as soluções manufaturadas dadas nas Eq. 5.1a, 5.1b, 5.1c e 5.1d no presente processo de verificação do código IMERSPEC CUDA.

A Tabela 6 apresenta os parâmetros de simulação utilizados em que o número de Reynolds, Re , é definido como $U_\infty L / \nu$ para $L = \pi [m]$. Conforme visto na seção 5.1, no código IMERSPEC MPI são utilizados o número de processos independentes ou núcleos como forma de paralelização em contrapartida, a paralelização em GPUs utilizam blocos de *Threads* para essa ação.

Sendo assim, para a execução do código IMERSPEC CUDA sempre deve-se definir mais dois parâmetros: a quantidade de *threads* em cada bloco e a quantidade de blocos totais. A multiplicação entre essas duas variáveis deve ser igual a quantidade de pontos de colocação da malha à serem executadas dentro do código, considerando que cada *thread* faz uma única operação. Dessa forma, adotando o valor da malha numérica como dado de entrada pode-se definir a quantidade de blocos de *threads* como uma consequência da definição do número de *threads* por bloco. Inicialmente, adotou-se 64 *threads* por bloco para as simulações computacionais.

Com os parâmetros definidos, tem-se a curva L_2 para os componentes de velocidade e para a variável de pós-processamento p , Fig. 38, onde ao contrário da verificação do MPF para o

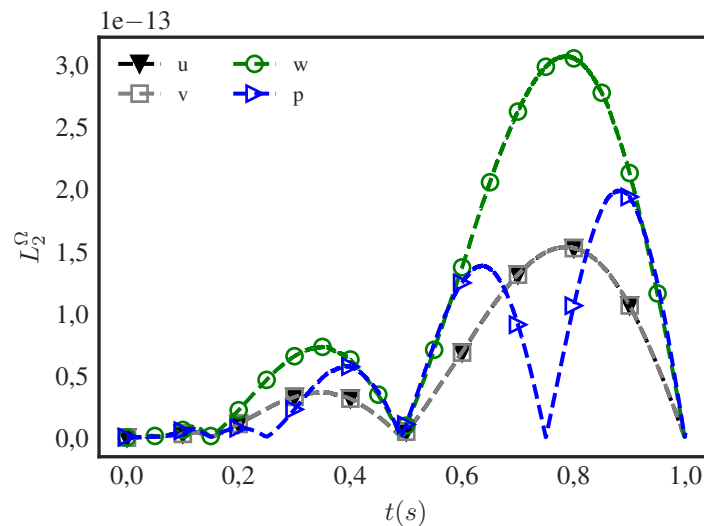
Tabela 6 – Propriedades de simulação relacionadas a verificação IMERSPEC CUDA.

$L_x \times L_y \times L_z$	$N_x \times N_y \times N_z$	$\Delta t(s)$	$t_f(s)$	$U_\infty(m/s)$	Re
$2\pi \times 2\pi \times 2\pi$	$16 \times 16 \times 16$	10^{-4}	1	1	10

Fonte: Próprio Autor.

IMERSPEC MPI, Fig. 22, nota-se o fenômeno de erro de cancelamento catastrófico. As razões por esse erro são as mesmas daquelas apresentadas na verificação do IMERSPEC MPI com fronteira imersa coincidente. Observa-se que o erro L_2 obtido na execução em GPU satura em torno de 10^{-13} , uma ordem de grandeza acima do valor observado na execução em CPU (10^{-14}) Fig. 22. Essa diferença é consistente com o efeito esperado da reordenação das operações de soma na redução paralela executada na GPU, uma vez que a adição em ponto flutuante não é associativa; a magnitude do desvio observado (uma ordem de grandeza) é compatível com o comportamento típico de acumulação de erro de arredondamento em dupla precisão IEEE 754, não indicando problema de exatidão da arquitetura em si, mas sim da ordem de acumulação empregada.

Apesar do erro numérico de cancelamento catastrófico, nota-se que esse erro é presente apenas no cálculo da norma L_2 e, ainda sim, tem um valor desprezível que não impacta em quaisquer modelagens. Atesta-se, assim, que a alta exatidão da metodologia MPF é mantida no código IMERSPEC CUDA já que resultados tão exatos são obtidos para uma malha de apenas 16^3 pontos de colocação, *i.e.*, a metodologia ainda apresenta exatidão de erro de máquina.

Figura 38 – Curva de L_2 para u , v , w e p para a verificação de MPF em IMERSPEC CUDA.

Fonte: Próprio autor.

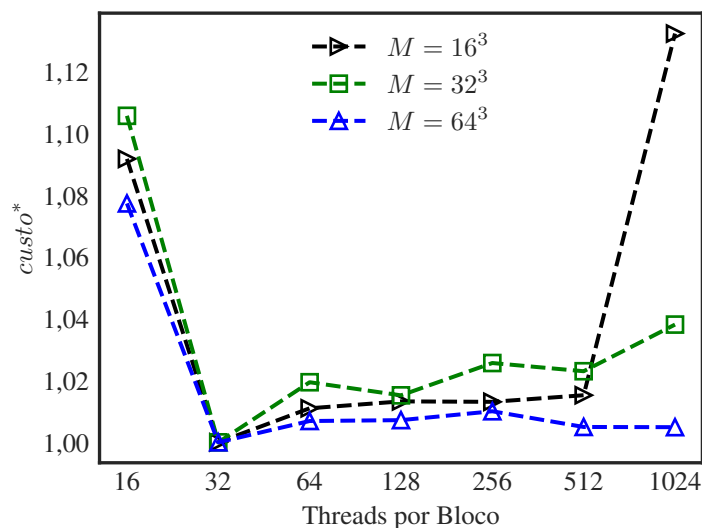
Verificado a exatidão da metodologia IMERSPEC, analisou-se dois parâmetros, em três malhas numéricas diferentes, relacionados a paralelização do código em GPUs: a quantidade de *threads* por bloco e a quantidade de operações matemáticas por *threads*. Em relação a primeira

análise, tem-se que a placa de vídeo utilizada nas simulações suporta blocos de até 1024 *threads*, assim, variou-se de 16 até 1024 *threads* por bloco no estudo realizado. Na Figura 39 é apresentada a relação entre quantidade de *threads* por bloco e o custo computacional adimensionalizado pelo menor custo computacional do grupo de simulações realizadas para as malhas de 16^3 , 32^3 e 64^3 pontos de colocação onde nota-se que o melhor resultado para todas as três malhas é a configuração de 32 *threads* por bloco.

GPUs da NVIDIA se subdividem em um grupo de unidades de processamento que executam diversos blocos simultâneos denominados por multiprocessadores, do inglês, *streaming Multiprocessors*, (SM). Conforme já mencionado, blocos são um conjunto de *threads*, entretanto, o processamento físico de dados dentro dos multiprocessadores ocorre em grupos de apenas 32 *threads* denominados por *warps*. Assim, o bloco pode conter um grupo maior de *threads* mas o processamento físico em si acontece em um múltiplo de 32 *threads*, *warps*. Por fim, tem-se a unidade *thread* em si que representa a menor unidade de execução.

Sendo assim, é recomendado sempre utilizar múltiplos de 32 na definição do número de *threads* por bloco à serem executados a fim de evitar *threads* ociosas e maximizar o paralelismo dentro do código. Conforme visualiza-se a Fig. 39, em blocos de 32 *threads* cada bloco apresentará apenas um *warp* por bloco, simplificando a execução física do código. Assim, infere-se que essa configuração torna mais simples o processo de escalonamento e distribuição dos blocos dentro dos multiprocessadores que compõem essa placa de vídeo.

Figura 39 – Custo computacional por número de *threads* em cada bloco.



Fonte: Próprio autor.

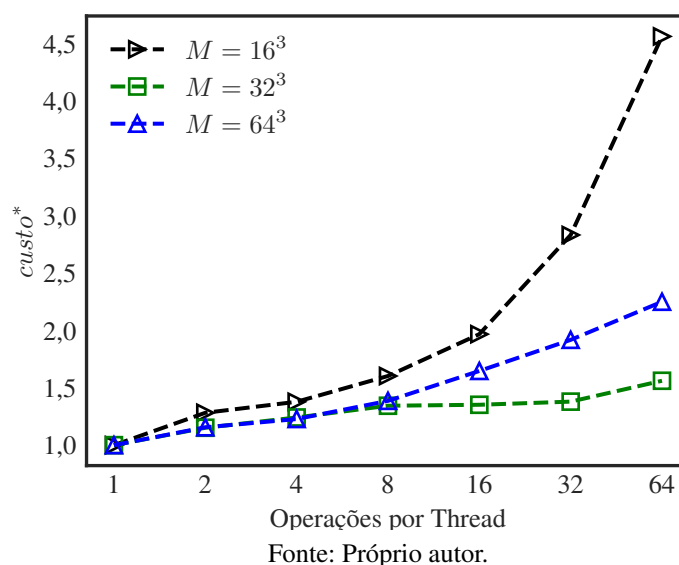
Outra significativa análise é a quantidade de operações matemáticas que cada *thread* realizará na placa GTX 1060. Para esse estudo se preservou, o máximo possível, acesso coalescido de memória dentro do código computacional, *i.e.*, caso mais de uma operação seja realizada dentro de um vetor a próxima operação será realizada em relação a posição adjacente da feita

anteriormente. A Figura 40 ilustra os resultados obtidos para a relação entre a quantidade de operações matemáticas por *thread* e o custo computacional obtido adimensionalizado pelo menor custo absoluto da rodada de simulações realizadas para três malhas numéricas diferentes. Verifica-se que operações unitárias por *thread* são a melhor escolha nas três malhas para essa placa de vídeo. Isso é lógico quando se observa as características físicas do *hardware* usado.

A GTX 1060 possui apenas 10 SMs e, por ser baseada na arquitetura Pascal, dispõe de menos registradores e memória compartilhada por SM, o que limita o número de *warps* residentes simultaneamente em cada SM. Essa limitação é relevante porque a capacidade de manter múltiplos *warps* residentes é o que possibilita o mecanismo de *latency hiding* (esconder a latência). Esse processo atua quando um *warp* fica ocioso aguardando o retorno de uma operação de acesso à memória global. Assim, o escalonador de *warps* do SM pode alternar a execução para outro *warp* já residente e pronto para executar, mantendo as unidades de processamento ocupadas e atuantes. Com menos *warps* residentes disponíveis para essa troca, a GTX 1060 satura sua capacidade de esconder latência com um número menor de blocos por SM. Por esse motivo, para essa placa, aumentar o paralelismo bruto (mais *threads* simultâneas, menos trabalho por *thread*) tende a ser vantajoso ao longo de toda a faixa testada.

Portanto, mesmo que o refinamento de malha cause maior quantidade de blocos a serem processados por chamada de kernel para a execução completa da operação matemática descrita, essa placa de vídeo ainda é mais eficiente fazendo esse gerenciamento de blocos do que realizando mais de uma operação por *thread*.

Figura 40 – Custo computacional por número de operações em cada thread.



Com as análises realizadas, adotou-se a configuração de 32 *threads* por bloco realizando operações matemáticas unitárias em todas as posteriores simulações quando se utilizou a placa GTX 1060.

Após a verificação do código IMERSPEC CUDA na placa GTX 1060, o código foi executado junto a placa A100. A exatidão do código se manteve com precisão de erro de máquina e de acordo com o já relatado, Fig. 38. Entretanto, para o novo *hardware*, tem-se um cenário de paralelização diferente. Para esta placa de vídeo há significativamente maior quantidade de SMs, 108, e, portanto, de CUDA cores. Consequentemente, se torna possível que cada *thread* processe mais de um elemento visto um melhor mecanismo de *latency hiding*.

Assim, para a A100, tem-se dois parâmetros de entrada para a paralelização: a quantidade de *threads* por bloco e a quantidade máxima de blocos de *threads*. Destaca-se que há uma diferença fundamental para a execução do código IMERSPEC CUDA nesta placa de vídeo. Necessariamente sempre haverá mais de um elemento sendo processado por cada *thread*.

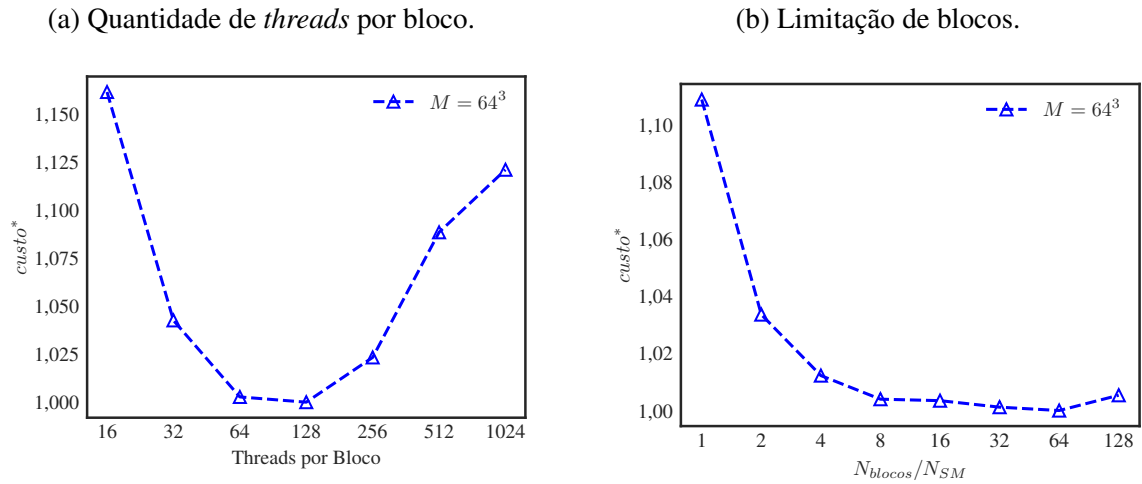
Nas Figuras 41a e 41b ilustram-se o impacto dos dois parâmetros na performance do código em relação ao custo computacional adimensionalizado pelo menor valor do grupo de simulações executadas. Os parâmetros da Tab. 6 também foram implementados para esse grupo de experimentos e destaca-se que para essas análises se considerou apenas a malha de 64^3 pontos de colocação.

Analizando o impacto da quantidade de *threads* por bloco, Fig. 41a, com a quantidade de blocos limitada a $32N_{SM}$, em que N_{SM} é o número de SM da A100, observa-se um valor ótimo de 128 *threads* por bloco. Esse valor difere do observado para a GTX 1060, para a qual o valor ótimo foi de 32 *threads* por bloco, e reflete a maior capacidade da arquitetura Ampere de sustentar *warps* residentes por SM, viabilizada por sua maior quantidade de registradores e pelos limites de arquitetura superiores de *warps* e blocos simultâneos por SM em relação à arquitetura Pascal.

Fixando o valor de 128 *threads* por bloco, variou-se a limitação da quantidade máxima de blocos na execução do código para a A100, Fig. 41b, e observou-se um valor ótimo de $64N_{SM}$. Esse resultado explicita o número maior de blocos residentes por SM para A100 que permite um melhor *latency hiding*, *i.e.*, com mais *warps* residentes disponíveis, o escalonador consegue alternar a execução entre eles enquanto aguarda o retorno de acessos à memória global, mantendo as unidades de processamento ocupadas. Esse benefício começa a saturar a partir de $128N_{SM}$, quando o aumento no número de blocos não compensa mais o overhead de escalonamento associado. A GTX 1060, por sua vez, apresenta limites menores de *warps* residentes por SM, de modo que sua capacidade de *latency hiding* já está saturada com poucos blocos por SM, tornando sempre preferível maximizar o número de *threads* simultâneas em detrimento do trabalho realizado por cada *thread* individualmente.

Quando se compara a performance do código para as duas placas de vídeo, nota-se que a definição dos valores ótimos e até da forma como a paralelização na GPU será feita, com uma *thread* processando mais de um elemento ou não nos *kernels*, é restrita ao modelo da GPU. Assim, para o código IMERSPEC CUDA, não há parâmetros de performance globais ao contrário da paralelização MPI onde tem-se parâmetros de paralelização que independem do

Figura 41 – Análise de parâmetros de performance do IMERSPEC CUDA na placa GPU A100.



Fonte: Próprio autor.

modelo das CPUs usadas.

Definidas as melhores configurações para maximizar o paralelismo do código IMERSPEC CUDA para cada placa. Realizou-se uma comparação entre os custos computacionais da metodologia IMERSPEC paralelizada em GPU e a paralelizada com a biblioteca MPI para as malhas computacionais com 16^3 , 32^3 e 64^3 pontos de colocação. Destaca-se que as simulações utilizando MPI e a placa GTX 1060 foram realizadas com os mesmos parâmetros e realizados na mesma máquina em que o código IMERSPEC MPI foi simulada com 4 núcleos físicos. Adicionalmente, o código IMERSPEC CUDA também foi executado com os mesmos parâmetros das demais simulações junto aos seus respectivas condições de paralelização mais com a placa A100.

Na Tabela 7 apresenta-se os custos computacionais obtidos assim como a relação de ganho computacional dos resultados do código IMERSPEC CUDA em relação ao código IMERSPEC MPI para cada uma das placas de vídeo. É notável que, executando os códigos em uma máquina relativamente acessível, a performance do IMERSPEC CUDA é superior em todas as malhas simuladas e se torna mais expressivo com o refinamento de malha chegando a um ganho de 8,25 vezes mais rápido do que a performance do IMERSPEC MPI para a malha de 64^3 pontos de colocação. Ainda sim, destaca-se que o IMERSPEC MPI pode chegar a um custo computacional ainda menor de acordo com os resultados de *speedup*, Fig. 23, caso seja executado em um *cluster* ou uma máquina com mais núcleos físicos.

Em relação ao código IMERSPEC CUDA executado na placa A100, percebe-se que os valores definidos como ótimos são válidos apenas para malhas computacionais refinadas. Para a performance do código em uma malha de 16^3 pontos de colocação, nota-se um custo computacional superior a paralelização MPI com 4 núcleos. Infere-se que para essas malhas, a execução do código deve ser feita com a operação de um ponto colocante por *thread* analogamente ao que se tem para a placa GTX 1060. Como na presente tese, busca-se a execução do código para malhas

computacionais consideráveis, os valores ótimos para a performance do código nessa placa se mantém.

Salienta-se que apesar das diferenças de ganho computacional das duas placas de vídeo utilizadas, há uma significativa diferença de preço na aquisição de cada uma. A placa GTX 1060 pode ser obtida a partir de qualquer loja de produtos eletrônicos e, no momento de escrita dessa tese, apresenta preço de aproximadamente⁹⁹, R\$ 1.154,01. Por outro lado, a NVIDIA A100 é uma placa de vídeo utilizada em *datacenters* e ambientes corporativos, não tem a facilidade de acesso como a GTX 1060 e apresenta um preço de, aproximadamente¹⁰⁰, R\$ 150.200,00.

Tabela 7 – Custo computacional, em segundos, da verificação de MPF para IMERSPEC MPI e IMERSPEC CUDA.

Malha	MPI	GTX 1060	Ganho GTX 1060	A100	Ganho A100
$16 \times 16 \times 16$	147,22	24,16	6,09	167,96	0,88
$32 \times 32 \times 32$	1048,16	169,52	6,18	170,49	6,15
$64 \times 64 \times 64$	11146,13	1351,25	8,25	209,69	53,16

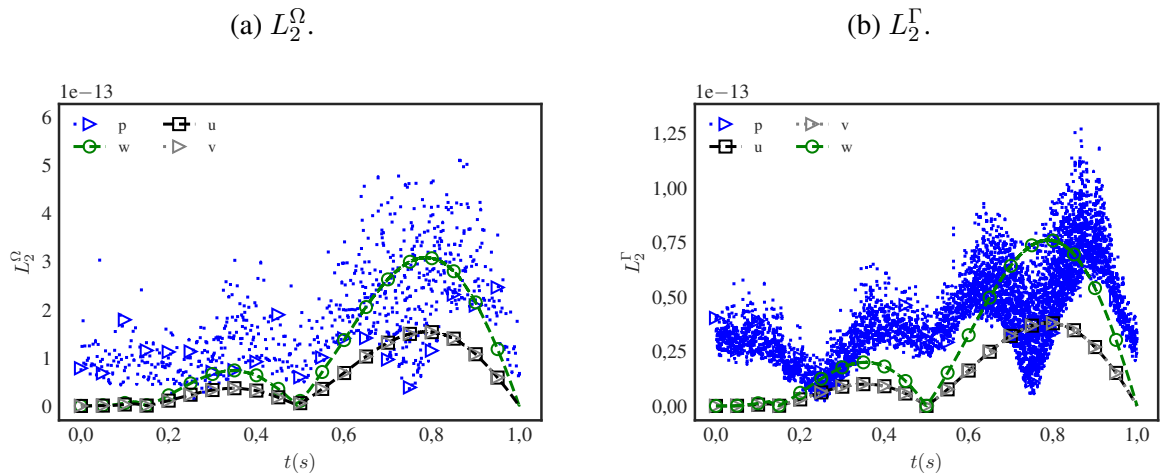
Fonte: Próprio Autor.

5.2.2 Verificação MFI Coincidente

Após às análises relacionadas apenas ao MPF, foi realizada a verificação do IMERSPEC CUDA com fronteira imersa Γ coincidente com pontos no domínio Ω . Conforme utilizado no processo de verificação descrito na subseção 5.1.2, uma casca em formato cúbico com lado igual a $\pi[m]$ foi posicionado no centro do domínio euleriano, Fig. 24. Salienta-se que a fronteira imersa impõe as velocidades das solução contínuas provenientes das Eq. 5.1a, 5.1b e 5.1c através do MDFM.

A partir dos parâmetros da Tab. 6 aplicados no contexto da verificação com a casca cúbica coincidente com o domínio euleriano, tem-se as normas L_2 definidas no domínio euleriano, L_2^Ω , e às determinadas no domínio lagrangiano, L_2^Γ para a simulação com 16^3 pontos de colocação. As Figuras 42a e 42b ilustram, respectivamente, as curvas L_2^Ω e L_2^Γ onde é notável a presença de erros numéricos causados por cancelamento catastrófico em todas as normas.

Analisando as normas L_2^Ω e L_2^Γ da variável de pós-processamento p , percebe-se um comportamento altamente oscilativo em ambas as normas conforme também visto na subseção 5.1.2. Ainda sim, percebe-se aqui a exatidão inferior que a GeForce GTX 1060 obtém em suas operações de dupla precisão visto que a ordem de grandeza dos dois tipos de normas são inferiores em relação às observadas nas Fig. 25a e 25b para todos os componentes de velocidade e também para a variável de pós-processamento p . Ressalta-se, entretanto, que mesmo com a perda de exatidão geral, as normas lagrangianas ainda são mais exatas do que às eulerianas conforme esperado de acordo com a metodologia MDFM/MFI e se mantém a exatidão de erro de máquina da metodologia IMERSPEC com o tipo de fronteira imersa simulada.

Figura 42 – Normas L_2 referentes a verificação com fronteira imersa coincidente de IMERSPEC CUDA.

Fonte: Próprio autor.

A fim de se comparar a performance do IMERSPEC MPI e IMERSPEC CUDA no contexto da presente verificação com Γ e Ω coincidentes realizou-se simulações com diferentes malhas computacionais utilizando os parâmetros da Tab. 6 na mesma máquina. Adicionalmente, também foram feitas simulações com os mesmos parâmetros para a placa A100 no *cluster* LaMCAD. Todas as simulações executadas com IMERSPEC MPI foram realizadas com os quatro núcleos físicos disponíveis. A Tabela 8 apresenta os custos computacionais, em segundos, para todas as simulações junto ao ganho relativo da performance de IMERSPEC CUDA sobre o IMERSPEC MPI para cada placa onde destaca-se o ganho de 24,40 para a simulação executada em GPU com a malha de 128^3 pontos de colocação relativo a GTX1060.

Em relação aos custos computacionais do código IMERSPEC CUDA junto a placa A100 apresentados na Tab. 8, nota-se o mesmo comportamento visto na Tab. 7. Para malhas com poucos pontos de colocação, a estratégia de *gride-stride loop* é maléfica, mas a medida que se tem malhas com maior quantidade de pontos de colocação, a performance do código se torna 271,74 vezes superior ao código MPI.

Tabela 8 – Custo computacional, em segundos, da verificação com fronteira imersa coincidente para IMERSPEC MPI e IMERSPEC CUDA.

Malha	MPI	GTX 1060	Ganho GTX 1060	A100	Ganho A100
$16 \times 16 \times 16$	121,50	26,19	4,64	450,96	0,27
$32 \times 32 \times 32$	1123,39	177,04	6,35	460,00	2,44
$64 \times 64 \times 64$	13728,23	1408,69	9,75	518,54	26,47
$128 \times 128 \times 128$	287547,41	11786,61	24,40	1058,19	271,74

Fonte: Próprio Autor.

Analisando a Tab. 8, nota-se o mesmo comportamento da Tab. 7, *i.e.*, a performance do código do IMERSPEC CUDA é superior para todas as malhas numéricas em relação ao código

IMERSPEC MPI para a placa GTX 1060 e a placa A100 se torna o destaque da disparidade dos custos computacionais entre os códigos assim que as malhas simuladas se tornam mais proeminentes.

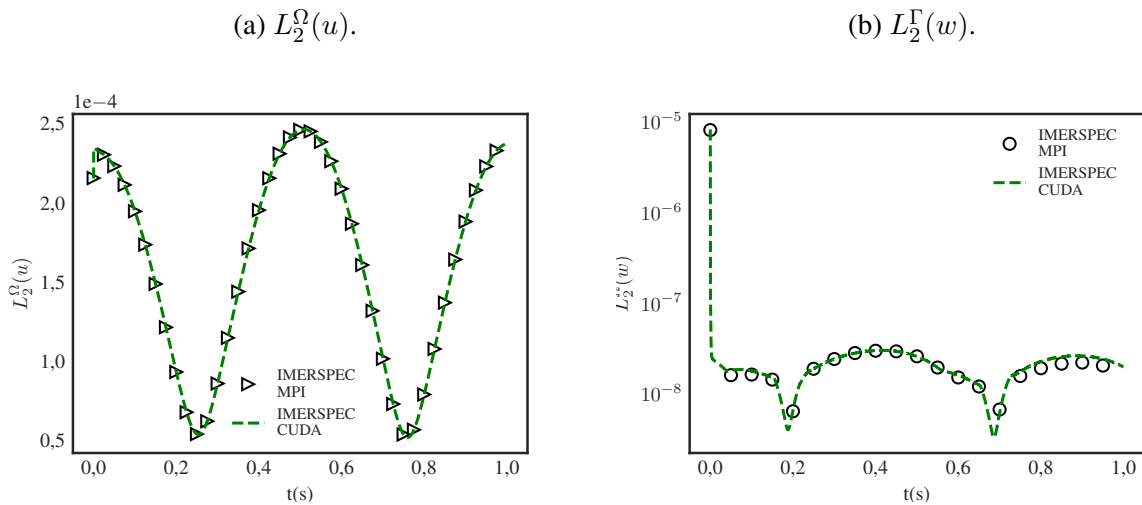
5.2.3 Verificação MFI Não Coincidente

A última verificação numérica do código IMERSPEC CUDA a ser realizada é a verificação do MPF junto a uma fronteira imersa Γ não coincidente. Assim como descrito na subseção 5.1.3, a fronteira imersa não coincidente utilizada é uma casca esférica de diâmetro $\pi [m]$ centralizada no domínio euleriano, Fig. 27. A malha lagrangiana adotada é não uniforme e criada externamente ao código IMERSPEC CUDA através do *software* livre SALOME.

A fim de analisar a exatidão do IMERSPEC CUDA com essa condição de contorno implementada, utilizou-se os parâmetros de simulação da Tab. 6. A partir da simulação realizada, obteve-se curvas L_2 dos domínios Ω e Γ para os componentes de velocidade e a variável de pós-processamento p iguais às curvas obtidas na verificação do código IMERSPEC MPI. A Figura 43a ilustra as curvas L_2^Ω para o componente u da velocidade ao longo do tempo físico simulado e a Fig. 43b apresenta as curvas L_2^Γ para o componente w de velocidade.

Salienta-se que em nenhuma das normas das Fig. 43a e 43b são observados indícios de erro de cancelamento catastrófico. Como os resultados numérico-computacionais e as soluções contínuas não são tão próximas da ordem de erro de máquina, esse fenômeno numérico não ocorre no cálculo das normas. Outro ponto de destaque é a exata correspondência entre as curvas das normas, o que indica que apesar de GPUs não serem tão exatas quanto um CPU em operações de dupla precisão, elas apenas apresentam resultados aquéns em operações de exatidão de erro de máquina.

Figura 43 – Comparação de curvas L_2 de IMERSPEC MPI e IMERSPEC CUDA



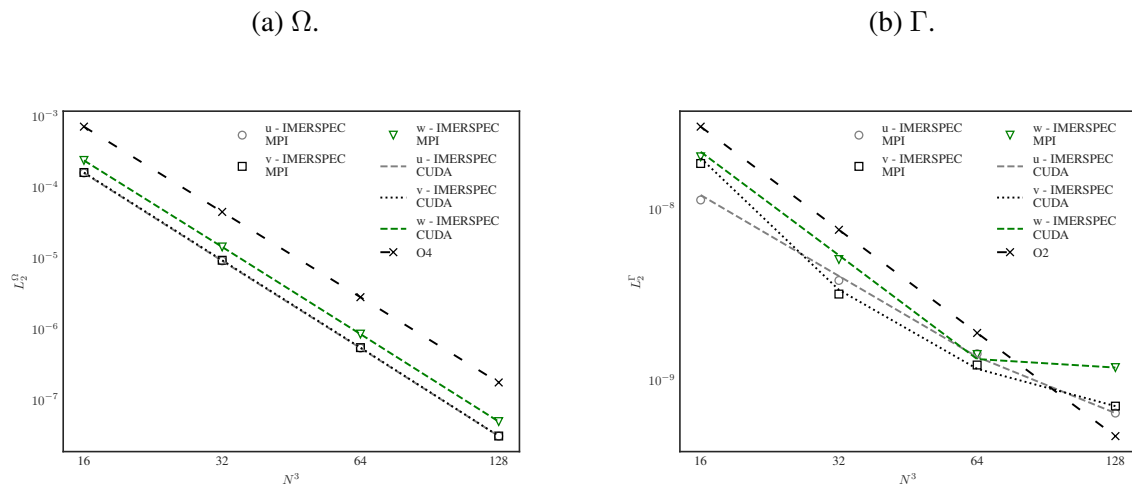
Após a comprovação dos mesmos resultados numéricos para a malha com 16^3 pontos de

colocação para os códigos IMERSPEC MPI e IMERSPEC CUDA, realizou-se novas simulações com os parâmetros da Tab. 6 para malhas computacionais com 32^3 , 64^3 e 128^3 pontos de colocação a fim de comprovar a ordem de convergência esperada nas normas L_2 eulerianas e lagrangianas.

São mostradas às curvas de convergência das normas eulerianas para os componentes de velocidade e pressão, Fig. 44a, obtidos para o presente código e àquelas obtidas na verificação do código IMERSPEC MPI, subseção 5.1.3. Nota-se exata correspondência entre os resultados e, conforme esperado, a metodologia IMERSPEC associada a uma fronteira imersa não coincidente apresenta quarta ordem de convergência.

Em relação às curvas de convergência das normas Lagrangianas (Fig. 44b), são feitas as mesmas comparações entre os códigos. Conclui-se que as normas Lagrangianas convergem seguindo uma curva de segunda ordem, conforme esperado.

Figura 44 – Comparação de curvas de convergência de IMERSPEC MPI e IMERSPEC CUDA.



Fonte: Próprio autor.

Após às análises relacionadas a exatidão da fronteira imersa não coincidente no código IMERSPEC CUDA foram realizadas comparações da performance dos dois códigos desenvolvidos na presente pesquisa. Novamente, o IMERSPEC MPI foi executado sempre paralelizado com quatro processos. Essa configuração é vantajosa para esse código com a configuração de verificação analisada visto que, de acordo com a Fig. 30, é a maior quantidade de processos em que o *speedup* do código é coincidente com o seu respectivo comportamento teórico.

Os custos computacionais obtidos para cada malha computacional simulada em cada um dos códigos e são apresentados na Tab. 9 junto ao ganho de performance do código IMERSPEC CUDA para cada GPU em relação ao IMERSPEC MPI. A paralelização da metodologia IMERSPEC com GPU resulta em um performance de até 20 vezes superior a sua paralelização usando MPI para a placa GTX 1060 enquanto pode chegar a um custo computacional 243,28 vezes menor para a placa A100. No IMERSPEC CUDA, o impacto da aplicação do

MDFM/MFI em pontos lagrangianos localizados em regiões específicas do domínio euleriano não causa a perda de paralelização tão expressiva observada na paralelização do código IMERSPEC MPI, Fig. 30.

Salienta-se que o ganho da placa A100 é positivo mesmo para a menor malha simulada devido a paralelização dos processos de interpolação/distribuição. Outro aspecto significativo é de que a paralelização por GPU obteve ganhos bem próximos aos obtidos na subseção anterior, ao contrário da paralelização MPI que tem performances do código IMERSPEC bem discrepantes para esses dois tipos de simulações. Nota-se assim, mais uma vantagem para o IMERSPEC CUDA executado em qualquer GPU testado.

Tabela 9 – Custo computacional, em segundos, da verificação com fronteira imersa não coincidente para IMERSPEC MPI e IMERSPEC CUDA.

Malha	MPI	GTX 1060	Ganho GTX 1060	A100	Ganho A100
$16 \times 16 \times 16$	724,79	212,99	3,40	522,76	1,39
$32 \times 32 \times 32$	3963,67	641,17	6,18	548,07	7,23
$64 \times 64 \times 64$	31745,80	3615,52	8,78	662,22	47,94
$128 \times 128 \times 128$	257926,31	12762,47	20,21	1060,19	243,28

Fonte: Próprio Autor.

Assim, considerando os resultados provenientes da Fig. 44a e da Tab. 9, pode-se afirmar que a paralelização do IMERSPEC CUDA é superior para esse tipo de fronteira imersa já que mesmo empregando mais processos, por exemplo, em um *cluster*, a performance do IMERSPEC MPI seria inferior pela limitação do MDFM/MFI em regiões específicas do domínio Ω . Indicando assim, um possível ganho ainda mais expressivo do que os já identificados na Tab. 9.

5.2.4 Verificação MIED/MFI

Conforme mencionado na subseção 4.1.3, na presente tese foi proposta uma nova metodologia para a modelagem de uma fronteira imersa não coincidente exclusivamente no IMERSPEC CUDA. Essa nova metodologia foi implementada apenas no código CUDA por dependência de operação em GPU da biblioteca NUFFT.

A verificação de MIED foi realizada de forma análoga à realizada nas subseções 5.1.3 e 5.2.3. Assim, a fronteira não coincidente em Γ também é uma casca em formato esférico posicionado no meio de Ω , Fig. 27, em que as suas malhas foram feitas no *software* livre externo SALOME. Os parâmetros de simulação apresentados na Tab. 6 também foram utilizados no primeiro experimento virtual.

A Figura 45a apresenta o campo de velocidade v/U_∞ resultante da simulação realizada com a metodologia MIED em uma malha de 16^3 pontos de colocação em um plano xz em $y = L$. Nota-se visíveis inexatidões na região euleriana em que se implementou a casca esférica. Isso

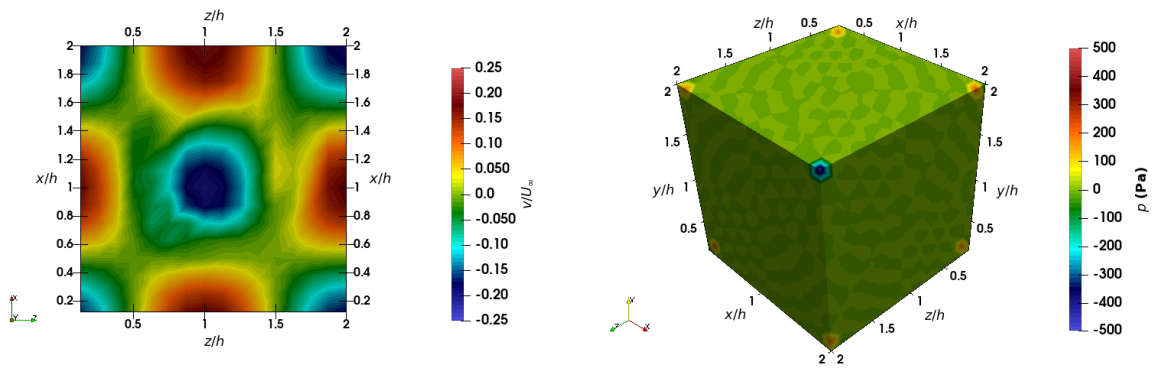
demonstra, qualitativamente, que a metodologia proposta não apresenta uma exatidão tão grande quanto o MDFM.

Outro aspecto significativo relativos aos resultados numérico-computacionais obtidos segundo essa metodologia está relacionado a variável de pós-processamento p . A pressão até então sempre foi determinada, após a solução da Eq. 4.32, pela Eq. 4.37, entretanto, utilizando a metodologia MIED obtém-se a pressão com significativas inconsistências conforme visto na Fig. 45b. Nota-se significativos valores absolutos de pressão nas “quinas” do domínio euleriano, comportamento não esperado.

Figura 45 – Resultados Computacionais para MIED com 16^3 pontos de colocação.

(a) Velocidade v no plano xz em $y = L$.

(b) Campo p .



Fonte: Próprio autor.

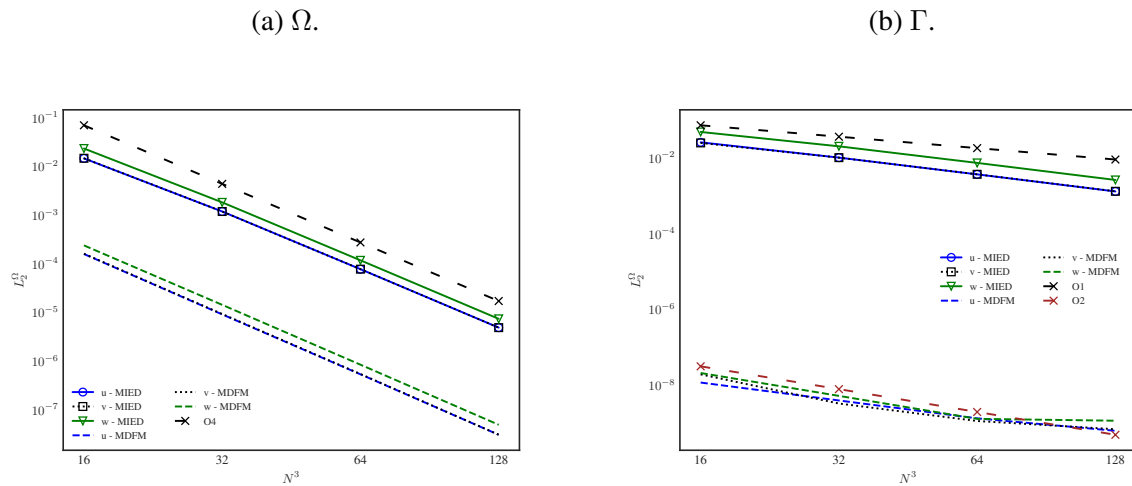
A partir dos resultados observados para a malha com 16^3 pontos de colocação, realizou-se novas simulações com a metodologia proposta em malhas com 32^3 , 64^3 e 128^3 pontos de colocação a fim de se determinar as suas curvas de convergência para a norma L_2 .

As curvas de convergência das normas L_2 obtidas para os componentes da velocidade para MIED e MDFM em Ω são apresentadas na Fig. 46a. Observa-se que tanto as normas de MIED quanto de MDFM possuem quarta ordem de convergência numérica, entretanto, é explícito o quanto o MDFM é mais exato do que o MIED em que o grupo de curvas das normas MIED se distanciam em cerca de duas ordens de grandeza do grupo de normas MDFM.

Quanto à comparação da convergência das normas L_2 de Γ , Fig. 46b, verifica-se que as normas associadas ao MIED convergem, aproximadamente, seguindo uma tendência linear (primeira ordem) enquanto as do MDFM exibem convergência de segunda ordem, na maior parte do refinamento de malha. Destaca-se que também são observada uma diferença de três ordens de grandeza entre os dois conjuntos de retas convergentes.

Após examinar a exatidão da metodologia MIED em comparação com a MDFM, analisou-se os custos computacionais das duas metodologias aplicadas no IMERSPEC CUDA. A Tabela 10 apresenta os custos computacionais das malhas simuladas aplicando MIED e MDFM. Reconhece-se que MIED apresenta menor custo computacional em todas as malhas simuladas, entretanto, o ganho computacional deste método tende a reduzir a medida que ocorre o refinamento de malha.

Figura 46 – Comparação de curvas de convergência de MDFM e MIED



Fonte: Próprio autor.

A metodologia MIED é baseada no uso da biblioteca NUFFT que realiza transformadas rápidas de Fourier não uniformes através da convolução do sinal não uniforme com funções de domínio compacto. Pelos resultados da Tab. 10 percebe-se que a convolução continua sendo uma operação matemática de alto custo mesmo com o seu domínio compacto e a medida que a malha fica mais grosseira as operações envolvidas nas transformadas não uniformes se tornam ainda mais significantes.

Tabela 10 – Comparativo de MDFM e MIED em relação aos seus custos computacionais, em segundos.

Malha	MDFM	MIED	Ganho
$16 \times 16 \times 16$	212,99	90,31	2,36
$32 \times 32 \times 32$	641,17	288,79	2,22
$64 \times 64 \times 64$	3615,52	2052,93	1,76
$128 \times 128 \times 128$	12762,47	16267,39	0,78

Fonte: Próprio Autor.

O processo de interpolação e distribuição é comum tanto na abordagem segundo MIED quanto MDFM mas esses processos ocorrem globalmente em MIED e localmente em MDFM. Buscou-se exatidão espectral e um menor custo computacional com o uso da NUFFT junto ao IMERSPEC, entretanto, o processo de interpolação e distribuição globais resultaram em soluções numérico-computacionais significativamente mais inexatas e com um custo computacional próximo ao método consolidado no IMERSPEC. Assim, entende-se que o MDFM continua sendo a abordagem mais indicada para a modelagem da fronteira imersa não coincidente para o cenário avaliado, a partir dos resultados da formulação MIED testada com os parâmetros de NUFFT adotados.

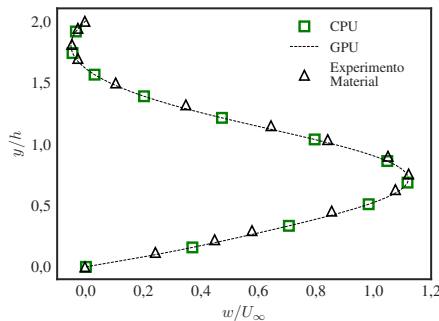
5.2.5 Escoamento Laminar sobre um Degrau

O processo de validação do código IMERSPEC CUDA foi análogo ao discutido para o código IMERSPEC MPI, subseção 5.1.4. Entretanto, não foi necessário realizar as mesmas análises de malha pseudo tridimensionais. Como o objetivo desta subseção é apenas validar o código através desse escoamento laminar. Simulou-se a malha computacional de $32 \times 64 \times 2048$ pontos de colocação com o novo código nas duas GPUs disponíveis, GTX 1060 e A100 a fim de se determinar se a exatidão da metodologia se mantém a mesma e se houve redução de custo computacional com a nova implementação da técnica de paralelização.

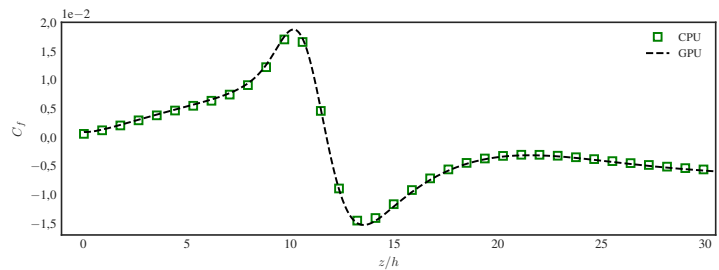
As Figuras 47a e 47b ilustram, respectivamente, a comparação de perfis verticais em $z = 14H$ para os dois códigos IMERSPEC MPI e CUDA para a modelagem do escoamento laminar sobre o degrau com $Re = 400$ assim como o resultado do trabalho material de referência¹⁸ e a distribuição do coeficiente de atrito C_f da parede inferior para os dois códigos. As figuras demonstram que a exatidão é a mesma para os dois códigos. Destaca-se que é apresentado apenas um resultado relativo ao código IMERSPEC CUDA mesmo que o código tenha sido executado em duas placas de vídeo visto que os seus resultados também foram idênticos.

Figura 47 – Comparação da exatidão dos códigos IMERSPEC MPI e CUDA

(a) Perfis verticais em $14H$.



(b) Distribuição de C_f na parede inferior.



Fonte: Próprio autor.

Em relação ao custo computacional dos diferentes códigos, relembra-se conforme apresentado na subseção 5.1.4 que a malha computacional havia sido simulada com 64 processos e foi obtido um custo computacional de 5 dias e 22 min. O código IMERSPEC CUDA modelando o mesmo fenômeno pelo mesmo tempo físico apresentou custo computacional igual a: 2 dias, 16 h e 8 min para a execução na placa GTX 1060 e 4 h e 8 min na placa A100. Esses custos computacionais representam um ganho de, respectivamente, 1,88 e 29,12 vezes para cada placa.

Salienta-se que nessa modelagem não são necessários processos de interpolação e distribuição visto que Γ e Ω são coincidentes, *i.e.*, a paralelização MPI apresenta a sua melhor performance e ainda foram usados 64 processos. Mesmo nesse cenário a paralelização heterogênea do IMERSPEC CUDA teve uma performance superior enquanto proporciona a mesma exatidão dos resultados computacionais. Portanto, após a verificação e validação dos dois códigos,

gos, continuou-se os desenvolvimentos da presente tese apenas com o IMERSPEC CUDA que apresenta mesma exatidão numérica e melhor performance.

Nesse capítulo foi realizado os processos de verificação numérica e validação física da metodologia IMERSPEC para os dois códigos, com técnicas de paralelização distintas, escritos na presente pesquisa. Notou-se que a paralelização MPI apresenta ligeira exatidão superior quando os resultados estão na ordem de erro de máquina entretanto na verificação do caso extremo de MFI com Ω e Γ não coincidentes, a própria paralelização MPI perde sua eficácia devido aos processos de interpolação/distribuição locais na malha computacional. A validação física para o código IMERSPEC MPI ainda demonstrou que mesmo a paralelização sendo feita em apenas duas das três dimensões do escoamento, há impacto na performance da malha e ainda atestou-se a exatidão espectral da abordagem IMERSPEC em um *benchmark* clássico.

Em seguida foi verificado e validado o código IMERSPEC CUDA onde descobriu-se que a execução do código deve ser calibrada particularmente a cada GPU, não há uma configuração global de paralelização deste caso. Ainda sim, para as duas placas de vídeo utilizadas, notou-se exatidão comparável ao código MPI e performance geral superior.

Adicionalmente, foi testado o método MFI MIED para substituir o MDFM na esperança de tornar toda a metodologia espectral em que os resultados provaram que a melhor alternativa de exatidão e performance continua sendo o MDFM. Ao final do capítulo, se concluiu que a modelagem de escoamentos turbulentos seria explorada apenas no código IMERSPEC CUDA e este será o tema a ser explorado no próximo capítulo.

6 RESULTADOS: MODELAGEM DE ESCOAMENTOS TURBULENTOS

Neste capítulo serão explorados os resultados da presente tese relacionados a modelagem de escoamentos turbulentos. Conforme discutido no capítulo 3, na presente tese foi utilizada a abordagem LES. Essa abordagem foi utilizada na modelagem do escoamento turbulento sobre o degrau. Na primeira parte do capítulo são discutidos alterações na estrutura do código realizadas devido a modelagem de interesse. Em seguida, são analisadas as condições de topo do domínio físico, necessárias para a correta representação do caso de referência. Por fim, são discutidos exclusivamente os modelos de fechamento funcionais submalha utilizados: Smagorinsky clássico com funções de Van Driest, WALE e Dinâmico de Germano-Lilly^{1,2}.

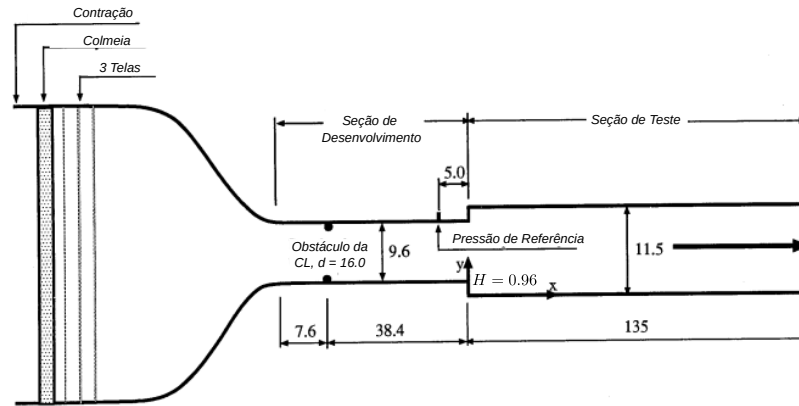
6.1 Trabalhos de Referência

Novamente escolheu-se o escoamento sobre um degrau para o caso de referência. Essa escolha foi feita devido a complexidade desse tipo de escoamento. Apesar de possuir uma geometria simples de modelagem, esse fenômeno físico apresentam características de escoamentos cisalhantes livres e parietais. Fenômenos como deslocamento e recolamento da camada limite e interação dessas regiões com estruturas turbilhonares estão presentes para esse clássico caso de referência.

O trabalho de referência material utilizado na modelagem do escoamento turbulento sobre degrau foi desenvolvido por Jovic; Driver¹⁶ em um túnel de vento de circuito aberto com número de Reynolds definido a partir da altura do degrau, $Re_H = 5000$. Os dados materiais do estudo, velocidades médias e flutuações, foram obtidos por meio de velocimetria a *laser Doppler* de dois feixes com dois componentes de fibra óptica, do inglês *Laser Doppler Velocimetry* (LDV) com exatidão de $\pm 2\%$ para a velocidade média e $\pm 15\%$ para as flutuações. Os coeficientes de atrito, C_f , foram quantificados através de um interferômetro a laser de escoamento de óleo, do inglês *Oil-Flow Laser Interferometer* em que a incerteza de medições dessa variável foi $\pm 0,0005$.

O esquema do experimento material é visto na Fig. 48. Nota-se que o escoamento é alinhado e ordenado pela região da contração da entrada do túnel de vento onde há a colmeia e as três telas. Na seção de desenvolvimento, a intensidade da turbulência na corrente livre é inferior a 1% e utiliza-se fios próximos as paredes inferior e superior para perturbar a camada limite a fim de simular uma camada limite turbulenta no experimento material. Após a seção de desenvolvimento, tem-se a expansão brusca característica do escoamento modelado com razão de expansão, $E_r = 1,2$.

Figura 48 – Esquema do Experimento Material.



Fonte: Adaptado de Jovic; Driver¹⁶.

Destaca-se que, no experimento material, o escoamento sobre o degrau foi feito em um canal com paredes levementes inclinadas para evitar o efeito de bloqueio causado pelo desenvolvimento das camadas limites. Destaca-se também o fato de que o degrau simulado materialmente tem forma simétrica, *i.e.*, ele é modelado na parede inferior e superior da seção de teste com as mesmas condições.

Alternativamente ao trabalho material, Le; Moin; Kim¹⁷, através de uma simulação numérica direta (DNS), modelaram o mesmo caso do experimento material discutido com algumas modificações. Neste trabalho, os autores não modelaram toda a seção de teste, conforme visto na Fig. 48, e sim apenas a metade da seção, modelando apenas um degrau. Com essa alteração, foi implementado a condição de simetria no topo do domínio computacional e adotou-se $Re_H = 5100$. Assim, o domínio computacional usado pelos autores foi de $30H \times 6H \times 4H$ em que o desenvolvimento do escoamento ocorreu na direção x e o degrau foi posicionado após $10H$ do *inlet* do domínio.

O DNS foi realizado a partir de um algoritmo com malha deslocada, na qual a pressão é definida no centro da célula computacional e as componentes de velocidade nas faces da célula. Espaçamentos de malha uniformes foram adotados nas direções do escoamento (x) e na sua direção homogênea (z). Um total de 768 células computacionais foram utilizadas na direção x e 64 células na direção z . Na direção vertical, perpendicular a parede, uma distribuição de malha não uniforme foi empregada, com espaçamentos refinados próximos à parede inferior e ao degrau. O número total de células computacionais na direção vertical é 192, das quais 70 estão posicionadas dentro do degrau ($y < h$). Os espaçamentos de malha nas três direções, em unidades de parede, são $\Delta x^+ \approx 10$, $\Delta y_{min}^+ \approx 0,3$, $\Delta y_{max}^+ \approx 31$ e $\Delta z^+ \approx 15$, respectivamente, baseados na velocidade de atrito da camada limite de entrada, u_{τ_0} .

Le; Moin; Kim¹⁷ validaram seus resultados computacionais com os dados materiais do

trabalho de Jovic; Driver¹⁶. Dessa forma, ao longo do presente capítulo, serão utilizados dados de ambos os estudos para analisar a modelagem LES, utilizando o código IMERSPEC CUDA, do fenômeno físico turbulento.

6.2 Modelagem Turbulenta e a Metodologia IMERSPEC

Um dos principais resultados do capítulo 5 foi a implementação de uma forma antissimétrica para o termo difusivo, esperava-se com isso a eliminação das ondas $2h$ na modelagem laminar do degrau, o que ocorreu, mas acima de tudo que essa forma fosse o suficiente para a modelagem de um fenômeno físico turbulento sem a presença de erros relativos a *aliasing*. Entretanto, infelizmente, a forma antissimétrica do termo difusivo não foi capaz de eliminar a contaminação dos números de onda devido ao fenômeno de *aliasing*, embora houvesse a sua minimização, ao longo do tempo físico modelado a contaminação ocorria e toda a simulação se tornava ruidosa sem qualquer reprodutibilidade de fenômeno turbulento de interesse.

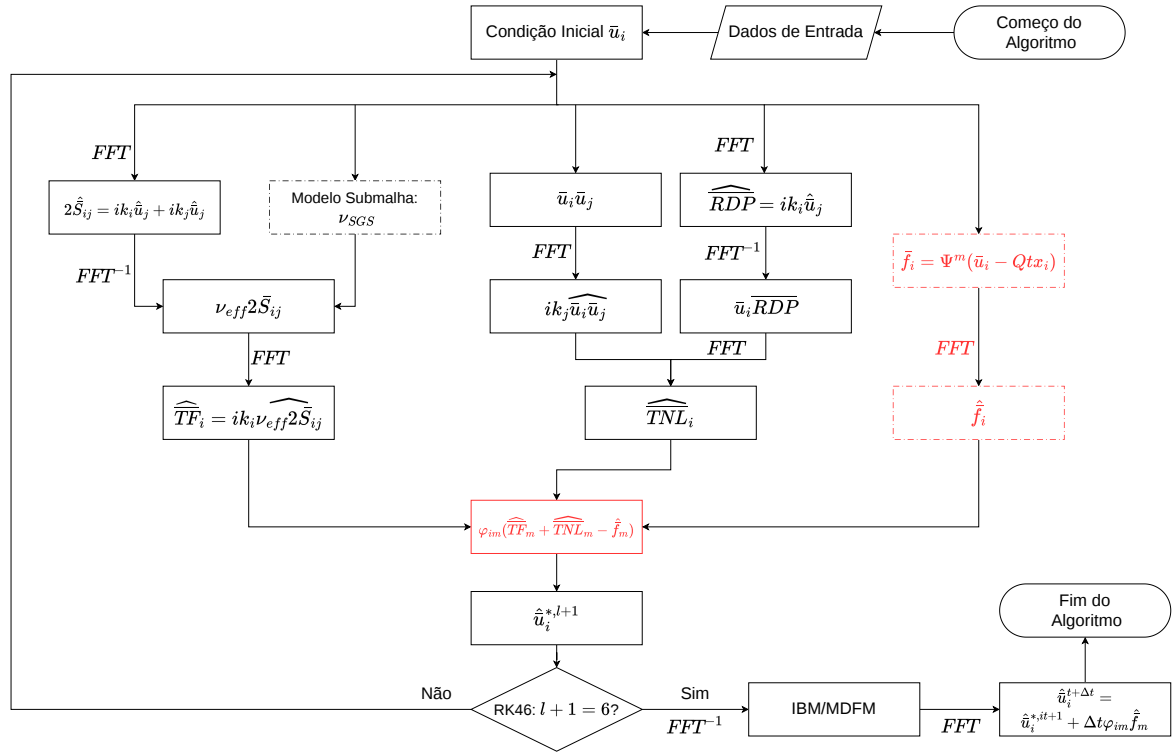
Sendo assim, a forma antissimétrica do termo difusivo foi descartada e adotou-se a forma do termo difusivo usual da metodologia IMERSPEC, Eq. 4.50. A forma antissimétrica do termo advectivo, por outro lado, embora reduza o erro de *aliasing* e conserve energia discreta²⁸, mostrou-se insuficiente para evitar o fenômeno de *aliasing* para escoamentos turbulentos. Conforme observado durante o desenvolvimento do código IMERSPEC-GPU, simulações com $Re \approx 5100$ apresentaram instabilidade numérica associada ao *aliasing*, cujo espectro de energia próximo ao número de onda de corte k_c é não desprezível nesse regime. Esse comportamento é consistente com a análise de Boyd⁶³, que demonstra que produtos de funções contém frequências superiores às de cada termo isolado, exigindo, em teoria, uma malha duas vezes mais refinada em cada direção para sua representação completa. A adoção do procedimento de *dealiasing* por truncamento espectral, em cada produto de duas funções, tornou-se, portanto, necessária para garantir a estabilidade e a acurácia das simulações em regimes turbulentos mais intensos.

Adicionalmente, determinou-se, que as ondas $2h$, presentes na modelagem do escoamento laminar sobre o degrau, surgiam devido a imposição da zona de forçagem justaposta a implementação das paredes pelo MDFM. Assim, apesar de amplamente utilização da zona de forçagem para aumentar a exatidão da condição de entrada do domínio físico na modelagem IMERSPEC^{52,58,60,61}, este artifício computacional foi descartado na modelagem do presente fenômeno físico turbulento.

Assim, sem a zona de forçagem, precisava-se de uma forma de aumentar a exatidão da condição de entrada sem o MDFM. A alternativa adotada foi o acoplamento total da zona de buffer no método de avanço temporal de Runge-Kutta de 4ª Ordem com 6 passos. Em trabalho anteriores, conforme explicitado no fluxograma da Fig. 20, a zona de buffer é imposta apenas uma vez por avanço temporal. Na nova implementação computacional adotada, para cada avanço temporal, essa correção é feita 6 vezes conforme explicitado e destacado em vermelho na Fig.

49.

Figura 49 – Fluxograma atualizado da metodologia IMERSPEC para modelagem de escoamentos turbulentos.



Fonte: Próprio autor.

Para que esse acoplamento seja válido, deve-se calibrar o coeficiente de amortecimento Ψ de modo que este resulte em um termo fonte $\bar{f}_i$ não desprezível durante o avanço temporal. Schlatter; Adams; Kleiser¹⁰¹ demonstrou, por análise de estabilidade linear temporal, que a condição $\Psi_{max} \Delta t \leq 2,78$ deve ser satisfeita para esquemas de Runge-Kutta de 4ª ordem. Considerando esse limite, adotou-se $\Psi^m = \Psi 0,1/\Delta t$, valor que garante exatidão na imposição da condição de entrada do domínio físico e, ao mesmo tempo, evita que o coeficiente de amortecimento da zona *buffer* seja excessivamente elevado, o que introduziria descontinuidades numéricas e o consequente fenômeno de Gibbs na solução computacional.

Com o fluxograma do IMERSPEC atualizado, foi definido um esquema para a modelagem do escoamento turbulento sobre o degrau com a definição da zona de *buffer*, condições de contorno impostas via MDFM, e regiões sem modelagem imposta a fim de preservar as condições de periodicidade inerentes ao uso de cuFFT presentes na fronteira do domínio computacional.

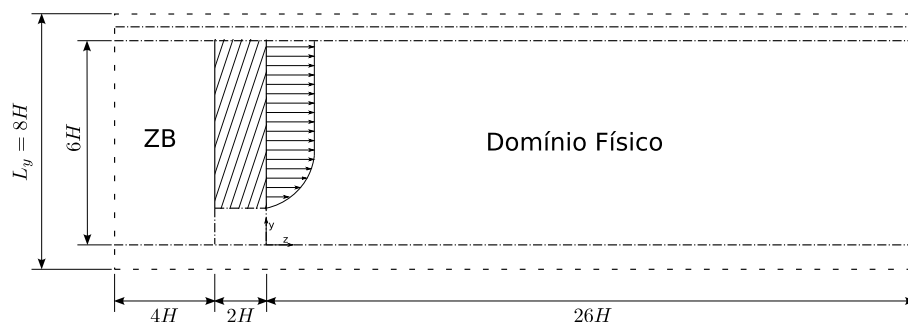
Na Figura 50 é ilustrado o esquema para a modelagem proposta. As condições impostas via MDFM são destacadas pela linha com traço e ponto, a linha tracejada indica onde as condições de periodicidade estão presentes. Destaca-se que o degrau é imposto via uma casca entre a zona de *buffer* e o domínio físico, isto é feito para que a turbulência sintética inserida na zona de *buffer* via ruído possa se desenvolver, um comportamento análogo a seção de desenvolvimento

no experimento material, Fig. 48. A região listrada corresponde a essa zona de desenvolvimento da turbulência sintética.

Destaca-se que as condições dessa modelagem, impostas na fronteira imersa, são do tipo Γ e Ω coincidentes e foram implementadas de acordo com o fluxograma da Fig. 9. Para as condições de não deslizamento, $\bar{U}_i^{IM} = 0$ e as demais condições são determinadas de acordo com a sua respectiva representação matemática.

Conforme discutido, para modelar o escoamento turbulento é necessário a inserção de ruído numérico. A verificação numérica do código, subseção 5.2.2, demonstrou a significativa exatidão do código. Dessa forma, caso não haja a imposição de ruído numérico, o escoamento modelado é ainda perturbado pelo degrau geométrico mas as instabilidades de Kelvin-Helmholtz não se tridimensionalizam devido a ausência de perturbações que, por exemplo, na modelagem de volumes finitos, estão inerentemente presentes visto a exatidão numérico-computacional menor desse método quando comparado ao MPF. As perturbações comentadas aqui estão presentes mesmo em experimentos materiais, como os desenvolvidos por Jovic; Driver¹⁶, estas são um efeito combinado sobre a simulação material de fenômenos como: escoamento desalinhado por sujidades nas colmeias e telas, vibração estrutural do túnel de vento devido a rotação do motor do ventilador, ruído da rede elétrica inserido nas respostas dos equipamentos de instrumentação, erros aleatórios devido a variações de propriedades do fluido providas das variações locais de temperatura e umidade, perturbações devido as rugosidades das paredes do túnel de vento, posicionamento irregular dos instrumentos de medida (variação nos ângulos de posicionamento do laser), entre outras.

Figura 50 – Esquema de modelagem do Escoamento Turbulento.



Fonte: Próprio autor.

Assim, no IMERSPEC é necessária a inserção de ruído numérico e para isso adotou-se o ruído de Smirnov; Shi; Celik³⁸ denominado por *Random Flow Generation* (RFG). Adotou-se esse ruído pelos resultados satisfatórios obtidos com esse algoritmo na metodologia IMERSPEC no trabalho de Moreira²³, a sua facilidade de implementação e migração para a linguagem CUDA C e baixo custo computacional quando comparado com outros métodos de geração de turbulência sintética. O ruído foi implementado utilizando $N_e = 500$ modos de Fourier. As escalas de comprimento de turbulência adotadas para as direções x , y e z são $0,3H$, $0,1H$ e

$0,6H$, com tempo característico de $0,6H/U_\infty$.

O ruído é imposto exclusivamente na região acima do degrau, para $1H < y \leq 1,5H$, através de uma função de amortecimento linear que garante transição suave entre a região de ruído e o escoamento livre, no limite superior da região de imposição. As flutuações são adicionadas ao campo de velocidade forçados na zona de *buffer* com fatores de escalonamento adimensionais distintos por componente de velocidade, 0,6, 0,5 e 1,0 para as componentes u' , v' e w' , respectivamente, de modo a reproduzir a anisotropia das flutuações turbulentas esperadas na região de entrada do escoamento sobre o degrau.

Salienta-se que esse escalonamento por componente de velocidade é independente da intensidade turbulenta global I_{turb} , adotada como 3% da velocidade de referência U_∞ . As flutuações do ruído são sobrepostas aos valores médios dos perfis de velocidade obtidos da DNS de Le; Moin; Kim¹⁷ usados como condição de entrada do domínio físico. Os parâmetros relacionados à inserção de ruído foram adotados em todas as simulações do presente capítulo e seus valores foram determinados empiricamente durante o processo de modelagem, buscando reproduzir a anisotropia das intensidades turbulentas reportadas por Jovic; Driver¹⁶ e Le; Moin; Kim¹⁷ na região de entrada do domínio.

Além da inserção de ruído, a Tab. 11 explicita alguns dos parâmetros adotados para os experimentos virtuais do capítulo. Esclarece-se que, ao contrário do trabalho DNS de referência, a direção x foi adotada como a direção homogênea e a direção z como a direção de desenvolvimento do escoamento. L_x , L_y e L_z são os tamanhos do domínio computacional total em que nota-se que todos eles são múltiplos de 2. Esse parâmetro é um resultado implícito da escolha de pontos de colocação, também múltiplos de 2, escolhidos para que a *cuFFT* tenha a melhor performance computacional possível. A condição de estabilidade de *Courant-Friedrichs-Lewy* é representada por *CFL*. O tempo físico adimensional simulado é definido como $t_a = 450U_\infty/H$ enquanto o tempo adimensional para início do cálculo de estatísticas foi determinado como $t_s = 150U_\infty/H$. O número de Reynolds, Re_H , é definido como $Re_H = \frac{HU_\infty}{\nu}$ em que H é a altura do degrau, U_∞ é a velocidade característica do escoamento e ν a viscosidade molecular do fluido newtoniano. Por fim, o critério de parada do MDFM para os experimentos virtuais é representado como ϵ na Tab. 11.

Tabela 11 – Propriedades das simulações do escoamento sobre degrau.

$L_x \times L_y \times L_z$	<i>CFL</i>	t_a	t_s	Re_H	$U_\infty(m/s)$	ϵ	$H(m)$
$4H \times 8H \times 32H$	0,4	900	300	5100	1	10^{-4}	0,5

Fonte: Próprio Autor.

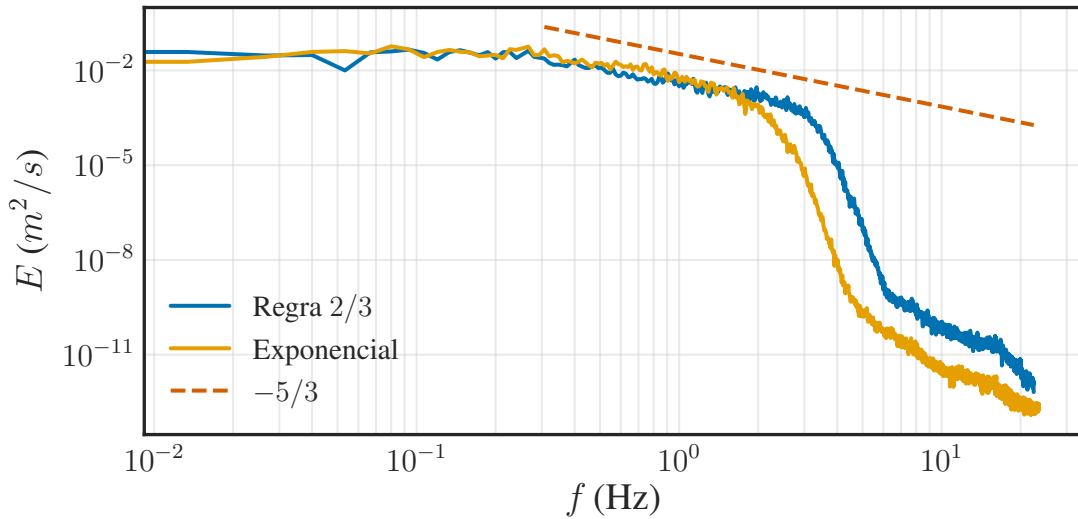
Duas malhas computacionais uniformes e distintas foram usadas ao longo das análises no presente capítulo: $32 \times 128 \times 256$ pontos de colocação com $\Delta x^+ = 30,9$, $\Delta y^+ = 15,5$ e $\Delta z^+ = 30,9$ e $64 \times 256 \times 512$ pontos de colocação com $\Delta x^+ = 15,5$, $\Delta y^+ = 7,7$ e $\Delta z^+ = 15,5$. Nota-se que os parâmetros Δx^+ , direção homogênea, e Δz^+ , direção do desenvolvimento

do escoamento, da malha mais refinada são próximos aos valores respectivos do trabalho computacional de referência¹⁷ enquanto Δy^+ é maior.

Após a apresentação dos parâmetros gerais adotados nas simulações, é necessário definir qual a forma de *dealising* a ser aplicado no IMERSPEC esquematizado na Fig. 49. Conforme descrito na subseção 4.3.4, foram testados duas formas distintas de *dealising*, a regra clássica de 2/3 que corresponde a um filtro passa baixa e um filtro exponencial.

Considerando o modelo funcional de fechamento submalha WALE, Fig. 50, condições de contorno no topo do domínio físico iguais às condições de simetria adotadas por Le; Moin; Kim¹⁷ e, ainda, uma malha computacional $32 \times 128 \times 256$ pontos de colocação, realizou-se duas simulações. Nessa análise, variou-se apenas o processo de *dealising* aplicado. As densidades espectrais de energia cinética específica turbulenta, $E(x_i, f)$, determinadas através de uma sonda em $(2H, H, 10H)$, são exibidas na Fig. 51.

Figura 51 – $E(x_i, f)$ para os distintos processos de *dealising*.



Fonte: Próprio autor.

Nota-se que a regra clássica de 2/3 apresenta uma queda mais súbita no espectro, enquanto o processo de *dealising* realizado pelo filtro exponencial ocorre de forma mais suave. Sabendo que o fenômeno de Gibbs, um erro espectral severo, surge de variações abruptas e descontinuidades no domínio de Fourier, optou-se pelo processo de *dealising* via filtro exponencial para as demais análises da presente tese. Admite-se, contudo, que essa escolha acarreta uma ligeira atenuação das estruturas turbilhonares de comprimento característico intermediário, uma vez que o decaimento suave do filtro avança sobre a região inercial do espectro de energia, conforme visto na Fig. 51.

Destaca-se que ao utilizar o procedimento de *dealising*, que é um processo de filtragem específico, o termo advectivo modelado na abordagem LES resultante é $\overline{\bar{u}_i \bar{u}_j}$ e não, apenas $\bar{u}_i \bar{u}_j$.

conforme visto na formulação matemática da abordagem LES, capítulo 3, e isso é uma limitação computacional a fim de preservar a estabilidade computacional das simulações.

6.3 Condições de Topo

As condições de topo do domínio físico, linhas tracejadas na Fig. 50 também devem ser discutidas. Considerando o trabalho de Le; Moin; Kim¹⁷, tem-se a opção de utilizar condições de simetria nessa casca a fim de modelar o fenômeno físico. Outra opção pode ser explorada dentro do IMERSPEC é a condição de transpiração explorada na modelagem satisfatória de uma camada limite laminar bidimensional através da metodologia IMERSPEC no trabalho de de Freitas; da Silveira Neto; Mariano⁶². A condição de transpiração é similar a condição de simetria em dois dos 3 componentes de velocidade que apresentam derivada perpendicular nula, entretanto ao invés de utilizar $v = 0$ conforme a condição de simetria, na condição de transpiração é aplicada a expressão $U_\infty \frac{\partial \langle \delta^* \rangle}{\partial z}$ em que $\langle \delta^* \rangle$ é a espessura média de deslocamento da camada limite do escoamento. Ressalta-se que a inclinação das paredes da seção de teste para evitar a bloqueio no experimento material realizadas por Jovic; Driver¹⁶ possuem o mesmo efeito no experimento material que a condição de transpiração para v no experimento virtual.

Outra possível forma de modelagem do escoamento turbulento sobre o degrau pelo IMERSPEC é a simulação do canal completo ao invés de meia região explicitada na Fig. 50, para esse caso tem-se um canal simétrico com o degrau implementado onde as condições de topo do domínio físico também são condições de não deslizamento. Destaca-se que para essa modelagem, o Ly assumiu o valor de $16H$ e o domínio físico tem altura igual $14H$, mantendo a mesma quantidade de pontos das malhas apresentadas na seção passada, tem-se um Δy^+ igual a metade do incremento adimensional anterior para uma malha, necessariamente, uniforme.

Assim, as três possibilidades de condições de topo do domínio físico podem ser resumidas pela Tab. 12 em que a condição “canal” representa a modelagem do canal simétrico completo. Considerando uma malha de $32 \times 128 \times 256$ pontos de colocação, foi simulada cada uma das condições de contorno do domínio físico discutidas com o modelo funcional submalha Smagorinsky dinâmico. Considerou-se esse modelo de fechamento pela expectativa de maior exatidão numérico-computacional.

Tabela 12 – Possíveis condições de contorno no topo do domínio físico.

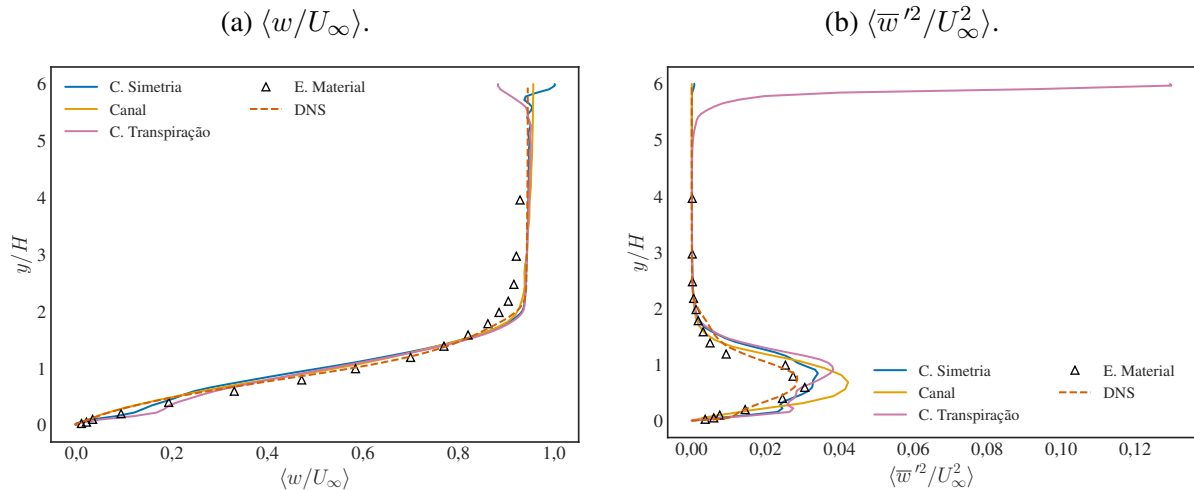
	u	v	w
C. Simetria	$\frac{\partial u}{\partial y} = 0$	0	$\frac{\partial w}{\partial y} = 0$
C. Transpiração	$\frac{\partial u}{\partial y} = 0$	$U_\infty \frac{\partial \langle \delta^* \rangle}{\partial z}$	$\frac{\partial w}{\partial y} = 0$
Canal	0	0	0

Fonte: Próprio Autor.

A fim de determinar qual estratégia de modelagem é a mais adequada para a reprodutibilidade do fenômeno físico, analisam-se perfis verticais em $x = 2H$ e $z = 6H$ da velocidade média $\langle w/U_\infty \rangle$, Fig. 52a, e da intensidade turbulenta $\overline{w'^2}/U_\infty^2$, Fig. 52b, comparados com os dados de referência. Os perfis de velocidade média apresentam um gradiente abrupto para as condições de contorno do tipo simetria e transpiração próximo a $y = 6H$, altura do domínio físico, enquanto a modelagem do canal completo mantém um comportamento suave nessa região, embora sem concordância com os dados experimentais ou com a simulação DNS, provavelmente em decorrência do espaçamento de malha Δy grosseiro. Destaca-se que as discrepâncias introduzidas pelas condições de simetria e transpiração parecem restritas à sua região de aplicação, não se propagando para o restante do perfil.

Por outro lado, ainda analisando a Fig. 52a, nota-se ainda discrepâncias entre as três alternativas de modelagem na região próxima a parede, $y \leq H$, em que as condições de contorno do tipo simetria e transpiração proporcionam valores mais próximos dos resultados materiais enquanto a modelagem do canal completo apresentou resultados próximos à simulação de Le; Moin; Kim¹⁷. De toda forma, todos os valores computacionais se distanciam do perfil de velocidade médio material na região inercial do escoamento, $y \geq 2H$.

Figura 52 – Perfis Verticais em $x = 2H$ e $z = 6H$.



Fonte: Próprio autor.

Analisando a intensidade turbulenta $\langle \overline{w'^2}/U_\infty^2 \rangle$, Fig. 52b, nota-se que a condição de simetria foi a que apresentou melhor aproximação com os dados de referência enquanto a modelagem do “canal” apresentou os piores resultados locais. A condição de transpiração embora apresente resultados intermediários para $y \leq 2H$, apresenta uma discrepância significativa da intensidade turbulenta na altura do domínio físico.

Considerando os perfis verticais para as duas grandezas ilustrados nas Fig. 52, conclui-se que a melhor condição de contorno para a modelagem do escoamento turbulento sobre o degrau é a condição de simetria. Embora esta introduza gradiente abrupto localmente em $y = 6H$, as discrepâncias não se propagam para o restante do perfil. Destaca-se ainda que, dada a altura do

domínio físico, não há risco de reflexão de estruturas turbilhonares decorrentes da condição de simetria.

Em relação à modelagem do canal completo, embora apresente comportamento suave na região $y = 6H$, deve-se considerar que o domínio computacional teve sua altura total dobrada, aspecto crítico no contexto do IMERSPEC, onde parte do domínio já é inerentemente perdida devido às restrições de periodicidade da *cuFFT*. Nessa abordagem, simula-se um domínio computacional maior para que apenas uma fração tenha validade física para o fenômeno de interesse, comprometendo a fidelidade da modelagem pela resolução de malha efetivamente disponível para o domínio físico, enquanto grande parte do domínio simulado não contribui para a representação do fenômeno físico.

Dessa forma, a partir dessa comparação, utilizou-se apenas as condições de simetria no topo do domínio físico, considerando um domínio computacional de $4H \times 8H \times 32H$, na análise dos modelos de fechamento funcionais submalha analisados na presente tese.

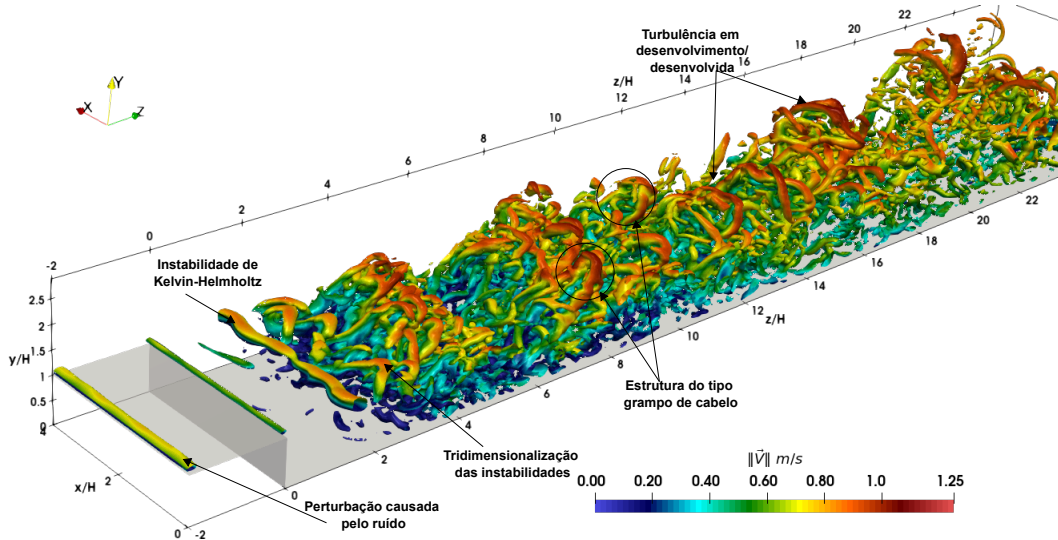
6.4 Modelo Submalha de Smagorinsky Clássico com Van Driest

Definidos os aspectos relacionados a modelagem do fenômeno físico, convém analisar o resultados computacionais do modelo de fechamento de Smagorinsky clássico junto com as funções *ad-hoc* de Van Driest, implementado na metodologia IMERSPEC. O escoamento turbulento sobre o degrau modelado é apresentado na Fig. 53 para a malha computacional de $64 \times 256 \times 512$ pontos de colocação em $t_a = 900$ com as estruturas turbilhonares representadas pelo critério Q com isovalor igual a 2,75 e coloridas pela magnitude da velocidade. O desenvolvimento temporal da Fig. 53 para o modelo submalha Smagorinsky com Van Driest é apresentado no vídeo *MP4_SMGK.mp4*¹⁰².

Através da Fig. 53 é visto o desenvolvimento do escoamento turbulento ao longo do domínio físico. Na região imediatamente a montante do degrau, $z < 0$, observa-se a perturbação introduzida pelo ruído sintético RFG, que desencadeia as instabilidades de Kelvin-Helmholtz na camada cisalhante separada, $0 \leq z \leq 4H$. Essas estruturas bidimensionais sofrem instabilidades secundárias que promovem a tridimensionalização do escoamento, $4H \leq z \leq 8H$, culminando no surgimento de estruturas do tipo grampo de cabelo (*hairpin vortices*) típicas de escoamentos turbulentos com parede, $8H \leq z \leq 14H$. Para $z > 14H$, o escoamento apresenta turbulência em desenvolvimento, com estruturas de múltiplas escalas distribuídas de forma progressivamente mais homogênea, evidenciando o processo de recuperação do escoamento após o recolamento. A região de baixa atividade turbulenta próxima à parede inferior para $z \leq 6H$ corresponde à zona de recirculação característica do escoamento sobre degrau.

A vista lateral dos campos da magnitude da vorticidade, $\|\vec{\omega}\|$ e da viscosidade efetiva

Figura 53 – Critério Q (2,75) em $t_a = 900$ para malha de $64 \times 256 \times 512$ pontos de colocação com o modelo submalha de Smagorinsky com Van Driest.



Fonte: Próprio autor.

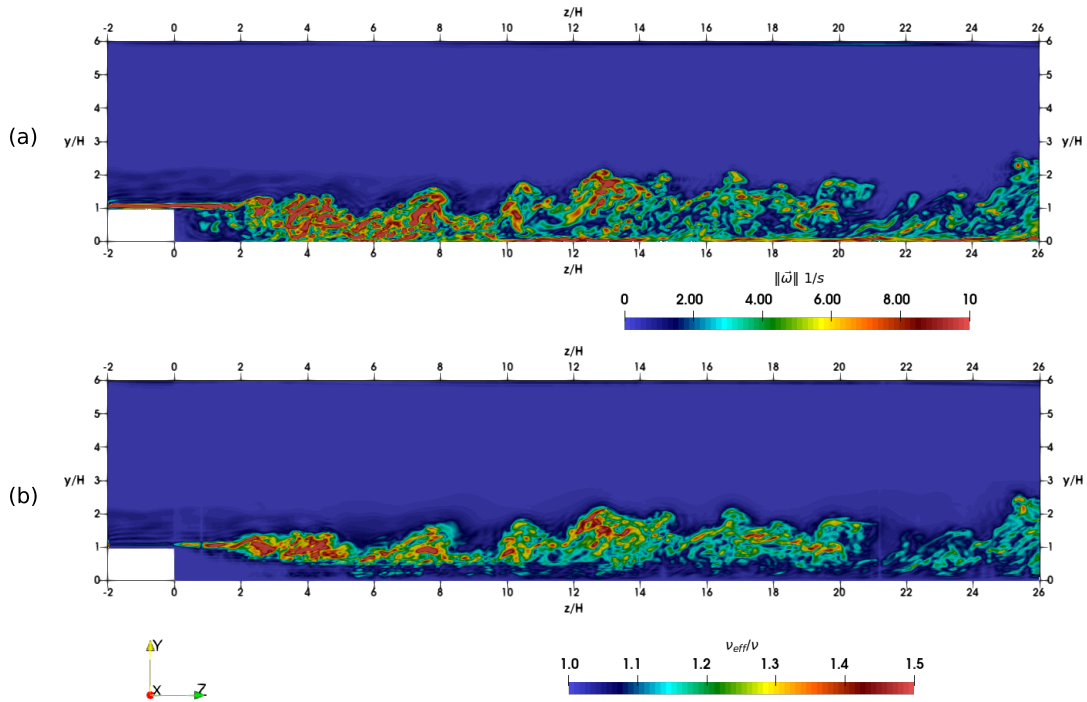
adimensionalizada, ν_{eff}/ν , são apresentados na Fig. 54 em $t_a = 900$ para a malha de $64 \times 256 \times 512$ pontos de colocação. A análise qualitativa de $\|\vec{\omega}\|$ indicam que a escolha das condições de contorno do topo do domínio físico como as condições de simetria não causaram a reflexão de estruturas turbilhonares na modelagem do fenômeno físico visto que as estruturas se desenvolvem apenas em regiões próximas às paredes ou intermediárias no domínio computacional, $y \leq 3H$.

Em relação ao campo ν_{eff}/ν , também apresentado na Fig. 54, notam-se linhas verticais com amortecimento máximo em $z \approx H$ e $z \approx 21H$. Essas linhas originam-se de gradientes abruptos na tensão de cisalhamento na parede τ_w , decorrentes da imposição das condições de contorno pelo MDFM. Caso τ_w fosse calculado a partir da derivada espectral global do campo de velocidade, o campo de viscosidade adimensional apresentaria um número ainda maior de linhas verticais, uma vez que a derivada espectral é um operador global, sensível a todas as frequências do domínio, incluindo as discontinuidades introduzidas pelo MDFM nas fronteiras imersas.

Para contornar esse problema, utilizou-se a aproximação por diferenças finitas de segunda ordem na determinação de τ_w , necessário para o cálculo de y^+ da Eq. 3.15. Diferentemente da derivada espectral, o operador de diferenças finitas é local, utiliza apenas os pontos vizinhos imediatos, sendo, portanto, menos sensível às oscilações não físicas provenientes de outras regiões do domínio. Dessa forma, o valor de τ_w calculado por diferenças finitas é mais representativo do comportamento físico local da tensão de cisalhamento na parede, reduzindo a ocorrência de linhas verticais espúrias no campo ν_{eff}/ν .

O código IMERSPEC CUDA foi estruturado para o processamento de ponto a ponto em operações globais devido a sua resolução espectral. O cálculo de τ_w através de diferenças finitas, força que uma parte desse modelo submalha atue localmente. Com base nisso, para os *kernels* relacionados a essa implementação utilizou-se blocos de *threads* bidimensionais com o valor

Figura 54 – Vista lateral dos campos de $\|\vec{\omega}\|$ (a) e ν_{eff}/ν (b) relativos ao modelo de fechamento submalha de Smagorinsky com Funções de amortecimento de Van Driest em $t_a = 900$ com $64 \times 256 \times 512$ pontos de colocação.



Fonte: Próprio autor.

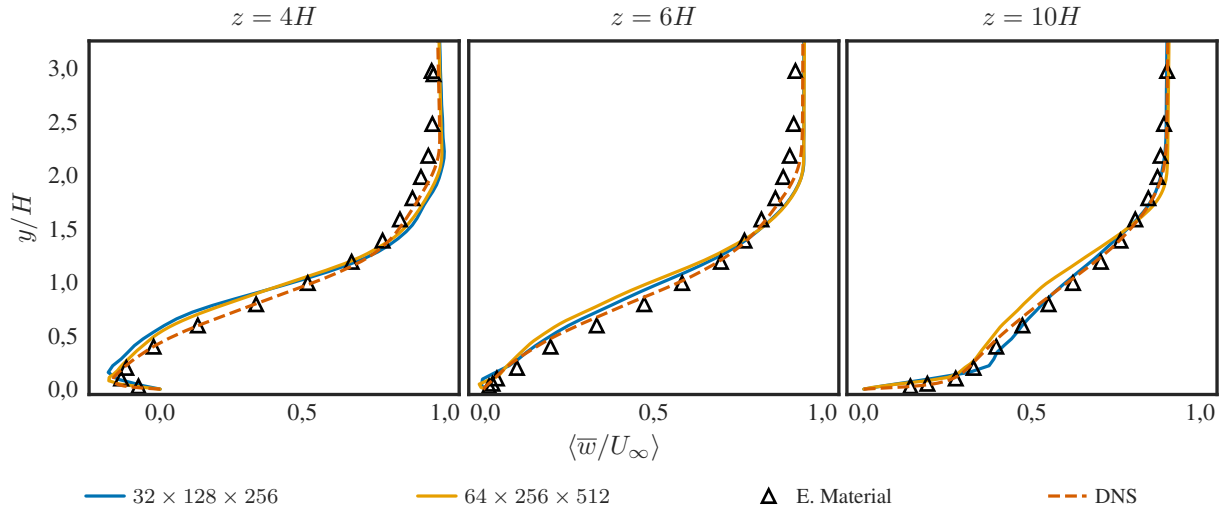
ótimo de *threads* previamente definido para cada placa de vídeo. Salienta-se, entretanto, que mais otimizações relacionadas a implementações de processos locais podem ser necessários caso mais *kernels* similares sejam utilizados em outras modelagens.

Ainda considerando a vista lateral de ν_{eff}/ν , Fig. 54, percebe-se o efeito do amortecimento global das funções de Van Driest, há um amortecimento abrupto e artificial na região próxima a parede $y \leq 0,5H$ onde há uma significativa redução dos valores de ν_{eff} . Esse comportamento qualitativo evidencia o caráter *ad-hoc* desse amortecimento.

Perfis verticais das diferentes malhas computacionais referente a velocidade média na direção do desenvolvimento do escoamento ao longo de $x = 2H$ são apresentados em 3 posições z distintas: $4H$, $6H$ e $10H$ na Fig. 55 junto aos respectivos perfis de referência. Para os perfis das malhas computacionais, observa-se, em geral, boa aproximação dos resultados com os trabalhos de referência, sobretudo ao trabalho DNS de Le; Moin; Kim¹⁷.

Entretanto, os resultados computacionais das duas malhas simuladas, para esse modelo de fechamento, ainda não apresentam recolamento da camada limite em $6H$ conforme os resultados materiais. Adicionalmente, identifica-se que o refinamento de malha na região de

Figura 55 – Perfis verticais de $\langle \bar{w}/U_\infty \rangle$ para o modelo de Smagorinsky com funções de Van Driest.



Fonte: Próprio autor.

$0,5H \leq y \leq 1,5H$ no perfil vertical referente a $10H$ piorou o perfil computado em relação aos resultados de Jovic; Driver¹⁶.

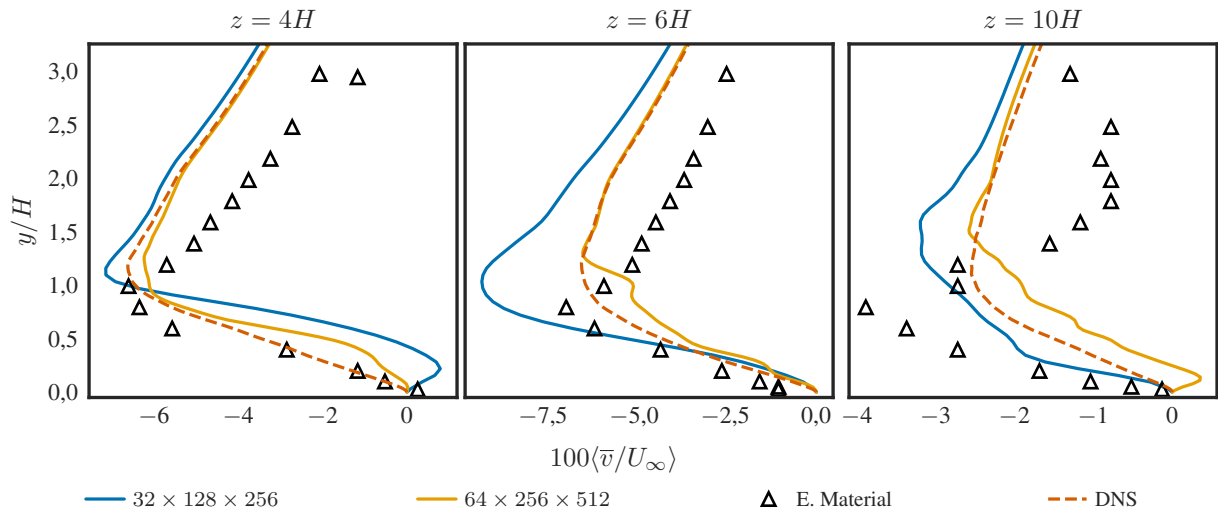
A componente vertical média, perpendicular às paredes das simulações, também é analisada nas mesmas posições, conforme visto na Fig. 56. Embora seja necessário destacar que o valor dessa componente de velocidade média é duas ordens de grandeza menor do que $\langle \bar{w}/U_\infty \rangle$, é explícito que os valores dos perfis de velocidade provenientes dos ensaios materiais não são os mesmos do trabalho virtual, dito DNS, especialmente para o perfil de vertical em $z = 10H$.

Ainda assim, analisando os resultados computacionais das duas malhas utilizadas nas simulações, tem-se resultados com tendência a convergência dos resultados numéricos de referência a medida que há o refinamento global da malha simulada. As regiões imprecisas dos perfis médios ocorrem em $y \leq 1,5H$ e se tornam mais significativas a medida que caminham a jusante do degrau. Destaca-se que essas regiões são as regiões próximas onde as funções *ad-hoc* de Van Driest não são desprezíveis.

Após a análise dos perfis verticais médios da velocidade, tem-se, para as duas malhas computacionais, os perfis da componente do tensor de Boussinesq-Reynolds, $\overline{w'^2}/U_\infty^2$, correspondente à tensão normal na direção z das estruturas turbilhonares resolvidas, e são comparados com os dados de referência de Le; Moin; Kim¹⁷ e Jovic; Driver¹⁶, Fig. 57. Novamente, os perfis demonstram um efeito acumulativo de imprecisão a medida que as posições se afastam do degrau, evidenciando erros devido ao caráter *ad-hoc* de C_s .

O refinamento de malha ainda proporciona uma melhor reprodutibilidade do fenômeno físico e destaca-se a região $1H \leq y \leq 2H$ em que os perfis $z = 4H$ e $z = 6H$ evidenciam uma

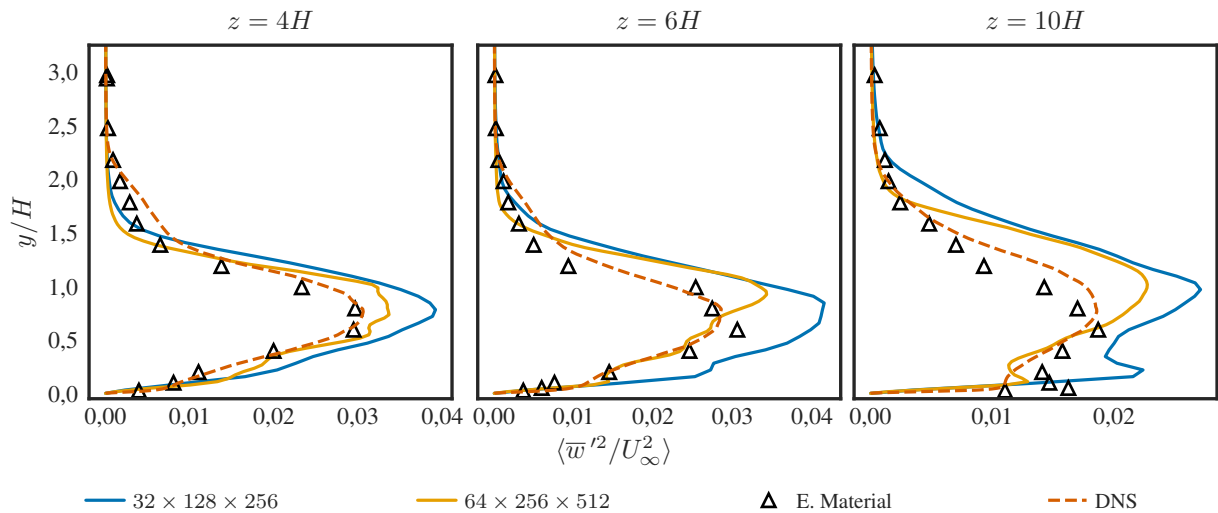
Figura 56 – Perfis Verticais de $\langle \bar{v}/U_\infty \rangle$ para o modelo de Smagorinsky com funções de Van Driest.



Fonte: Próprio autor.

tendência a aproximação maior dos resultados do IMERSPEC com abordagem LES em relação aos resultados de Jovic; Driver¹⁶ do que os resultados numéricos de Le; Moin; Kim¹⁷.

Figura 57 – Componente $\langle \bar{w}'^2/U_\infty^2 \rangle$ do tensor de Boussinesq-Reynolds para o modelo de Smagorinsky com funções de Van Driest.

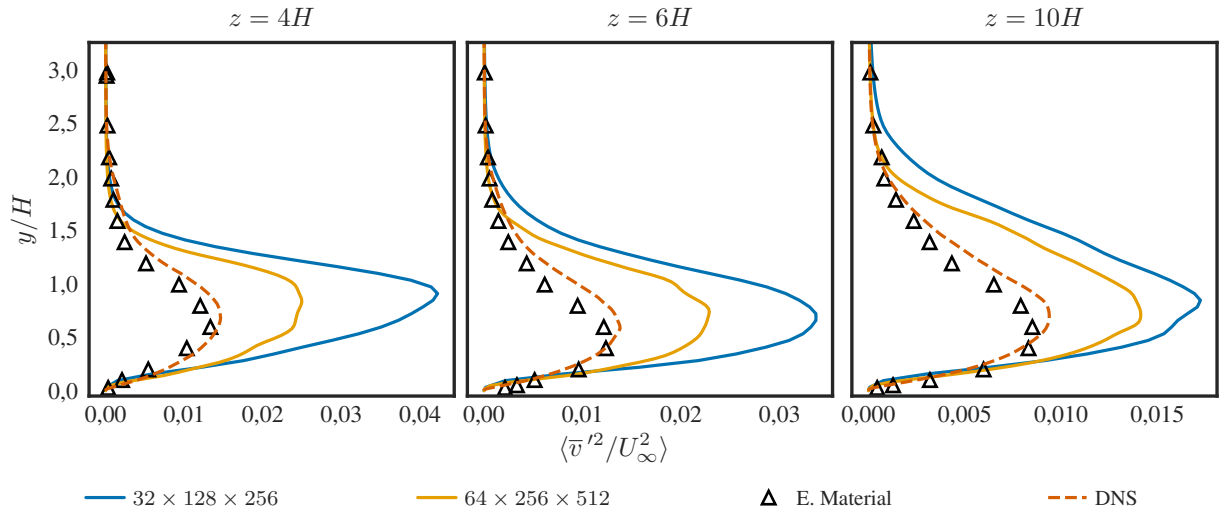


Fonte: Próprio autor.

Analogamente a Fig. 56, os perfis da componente do tensor de Boussinesq-Reynolds, $\langle \bar{v}'^2/U_\infty^2 \rangle$, correspondente à tensão normal na direção y das estruturas turbilhonares resolvidas são apresentados, para as malhas computadas, na Fig. 58. A figura evidencia uma tendência aos resultados de referência, mas também apresenta uma significativa diferença entre os resultados computados para as duas malhas à quaisquer resultados dos trabalhos consultados.

Torna-se claro, através da análise da Fig. 58, que um efeito mais anisotrópico na direção y durante a inserção do ruído numérico pode tornar os resultados computacionais mais próximos aos trabalhos de referência para esse modelo funcional de fechamento submalha.

Figura 58 – Componente $\langle \bar{v}'^2 / U_\infty^2 \rangle$ do tensor de Boussinesq-Reynolds para o modelo de Smagorinsky com funções de Van Driest.

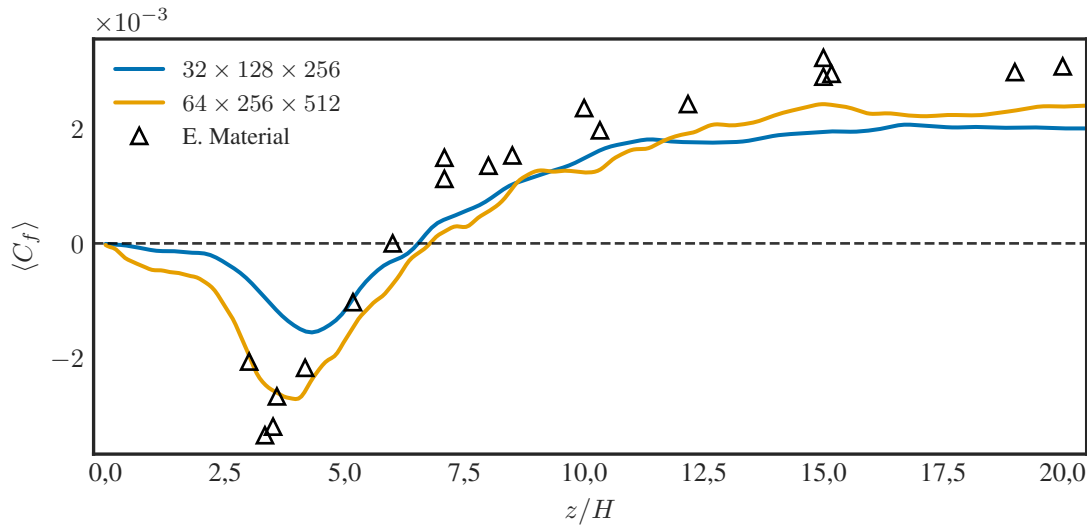


Fonte: Próprio autor.

A determinação do ponto de recolamento da camada limite do escoamento modelado pode ser feita a partir da distribuição do coeficiente de atrito médio, $\langle C_f \rangle$. As distribuições dessa grandeza para as diferentes malhas computacionais e os valores referentes ao trabalho material são apresentadas na Fig. 59. Destaca-se que a distribuição desse coeficiente médio foi feita em pós processamento, a partir dos campos médios de velocidade. Nenhuma das malhas computacionais apresentam distribuições de $\langle C_f \rangle$ exatamente iguais aos valores materiais, entretanto, nota-se uma tendência a convergência dos valores materiais para refinamentos sucessivos de malha.

Quanto ao ponto de recolamento da camada limite, os dados materiais demonstram um ponto igual a $6H$ enquanto os valores computacionais obtidos indicam um ponto igual a $6,56H$ e $6,78H$, respectivamente, para a malha de $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos de colocação. Assim, tem-se um aumento de, aproximadamente, 9% para 13% de erro na determinação do ponto de recolamento em relação ao refinamento global de malha. Isso indica uma tendência à convergência de um valor diferente do resultado material, causada provavelmente devido a hipótese usada para a definição do invariante utilizado no modelo submalha, e ao mesmo tempo, o fato do coeficiente C_s ser estático ao longo do experimento virtual.

A fim de avaliar a qualidade física das simulações e o efeito do refinamento de malha na resolução das estruturas turbilhonares, foram determinadas as densidades espectrais de energia cinética turbulenta $E(x_i, f)$ no ponto $(2H, H, 10H)$ para as duas malhas computacionais, com frequências de amostragem de 37 Hz e 45 Hz para $32 \times 128 \times 256$ e $64 \times 256 \times 512$, respectivamente. Os espectros obtidos são apresentados na Fig. 60.

Figura 59 – $\langle C_f \rangle$ para o modelo de Smagorinsky com funções de Van Driest.

Fonte: Próprio autor.

Ambas as malhas apresentam região inercial com decaimento compatível com a lei de Kolmogorov⁸¹, confirmando que o escoamento modelado reproduz as características físicas esperadas de turbulência desenvolvida. O efeito do refinamento de malha é evidente: a malha com $64 \times 256 \times 512$ pontos de colocação apresenta região inercial mais extensa, com frequência de corte deslocada de $f \approx 2 \text{ Hz}$ para $f \approx 4 \text{ Hz}$, corroborando a expectativa de que malhas mais refinadas em LES resolvem diretamente mais estruturas turbilhonares e modelam menos pelo SGS.

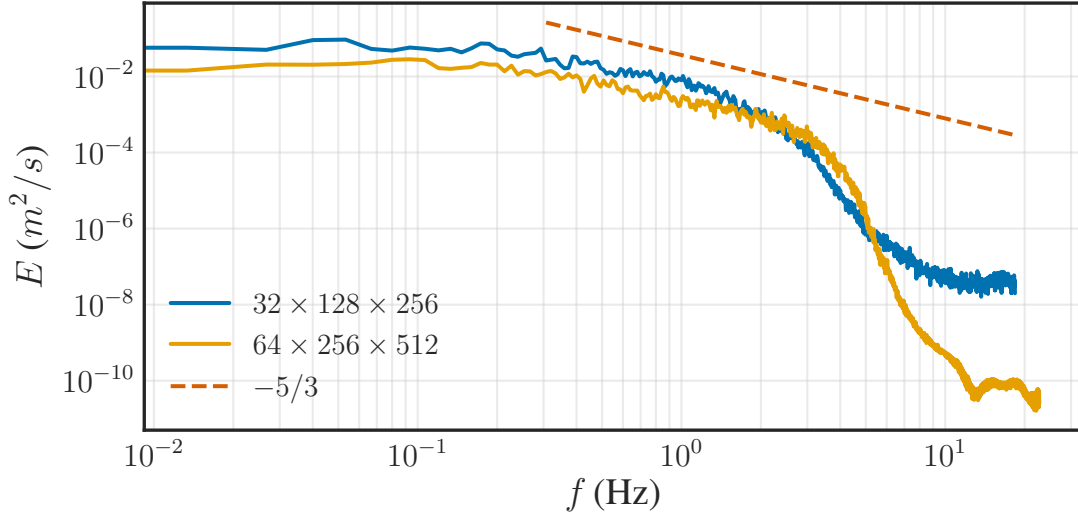
Nas altas frequências, o decaimento abrupto além da região inercial evidencia a atuação do modelo SGS e do processo de *dealiasing* na transformação viscosas das estruturas não resolvidas. A malha mais refinada apresenta esse decaimento em frequências significativamente mais altas, confirmando que a abordagem LES se deslocou para a direita, conforme esperado. Adicionalmente, a malha de $64 \times 256 \times 512$ pontos de colocação apresenta menor nível energético nas baixas frequências, indicando menor acúmulo de energia nas grandes estruturas e melhor resolução da cascata energética em comparação à malha mais grosseira.

Destaca-se, por fim, que o custo computacional da malha mais refinada executada na GPU A100 foi de, aproximadamente, 66 h e 15 min para a modelagem do fenômeno físico utilizando o modelo funcional de fechamento submalha de Smagorinsky com funções de amortecimento de Van Driest.

6.5 Modelo Submalha WALE

Os resultados das duas malhas computacionais, $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos de colocação, referentes ao modelo submalha WALE são considerados nesta seção. A Figura 61

Figura 60 – Espectros com Modelo Clássico de Smagorinsky.



Fonte: Próprio autor.

ilustra as vistas laterais dos campos da magnitude de vorticidade, $\|\vec{\omega}\|$, e a viscosidade efetiva adimensionalizada, ν_{eff}/ν , em $t_a = 900$ para o modelo de fechamento de turbulência em análise.

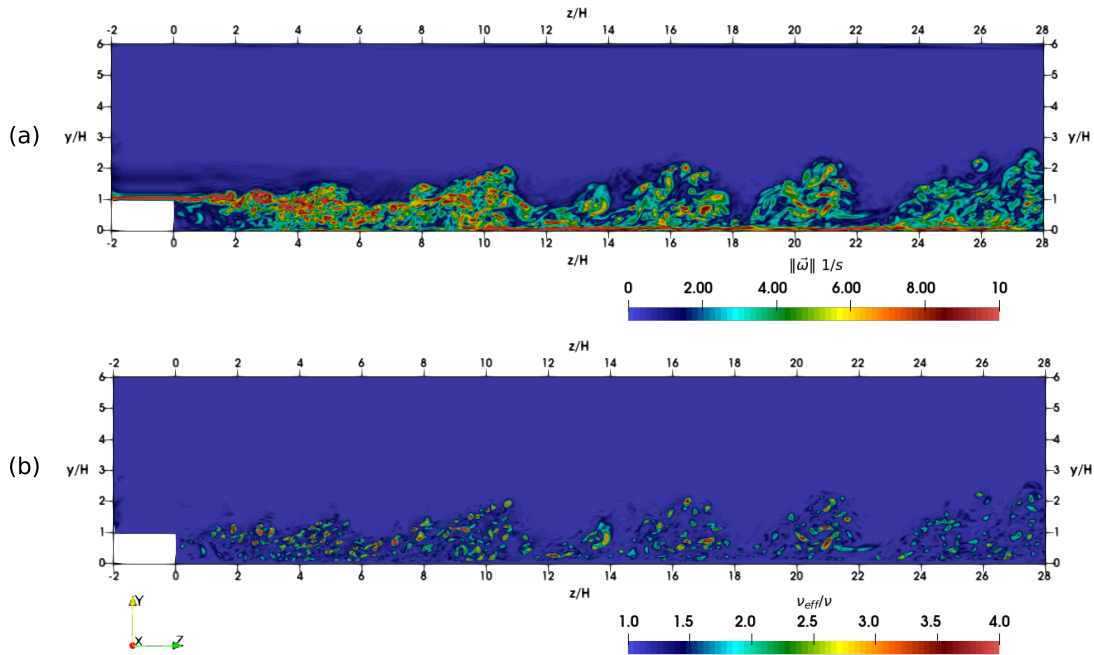
O desenvolvimento temporal da simulação com o modelo WALE e a malha com $64 \times 256 \times 512$ pontos de colocação representado pelo critério Q com isovalor de 2,75 colorido pela magnitude da velocidade é apresentado através do vídeo *MP4_WALE.mp4*¹⁰². A análise qualitativa do vídeo mostra um comportamento similar ao apresentado para o modelo de Smagorinsky clássico abordado na seção 6.4, entretanto, as estruturas turbilhonares parecem estar mais definidas do que o modelo submalha anterior.

Em relação a magnitude da vorticidade instantânea vista na Fig. 61, pode-se afirmar que qualitativamente o modelo WALE mantém o comportamento esperado de uma simulação LES, originando um escoamento com estruturas turbilhonares similar ao obtido com o modelo da seção anterior.

O campo lateral de ν_{eff}/ν apresentado na Fig. 61 difere significativamente do mesmo campo observado na Fig. 54 para o modelo de Smagorinsky com amortecimento de Van Driest. O modelo WALE apresenta ν_{SGS} mais intensa, com valor máximo aproximadamente 2,6 vezes superior ao obtido com o modelo de Smagorinsky clássico. Comparando os campos de vorticidade e de viscosidade adimensional, nota-se que o modelo submalha apresenta maior atividade nas regiões de intensa interação entre deformação e rotação, (tensor $\bar{G}_{ij}^d$), correspondentes ao cerne das estruturas turbilhonares. Isso está em acordo com Sagaut²⁸ que afirma que $\nu_{SGS} \rightarrow 0$ quando o escoamento é bidimensional.

Destaca-se ainda que, próximo à parede, a viscosidade adimensional apresenta $\nu_{eff}/\nu = 1$, confirmando que $\nu_{SGS} \rightarrow 0$ nessa região, comportamento assintótico $\mathcal{O}((y^+)^3)$ correto do modelo

Figura 61 – Vista lateral de $\|\vec{\omega}\|$ (a) e ν_{eff}/ν (b) relativos ao modelo de fechamento submalha WALE em $t_a = 900$ com $64 \times 256 \times 512$ pontos de colocação.



Fonte: Próprio autor.

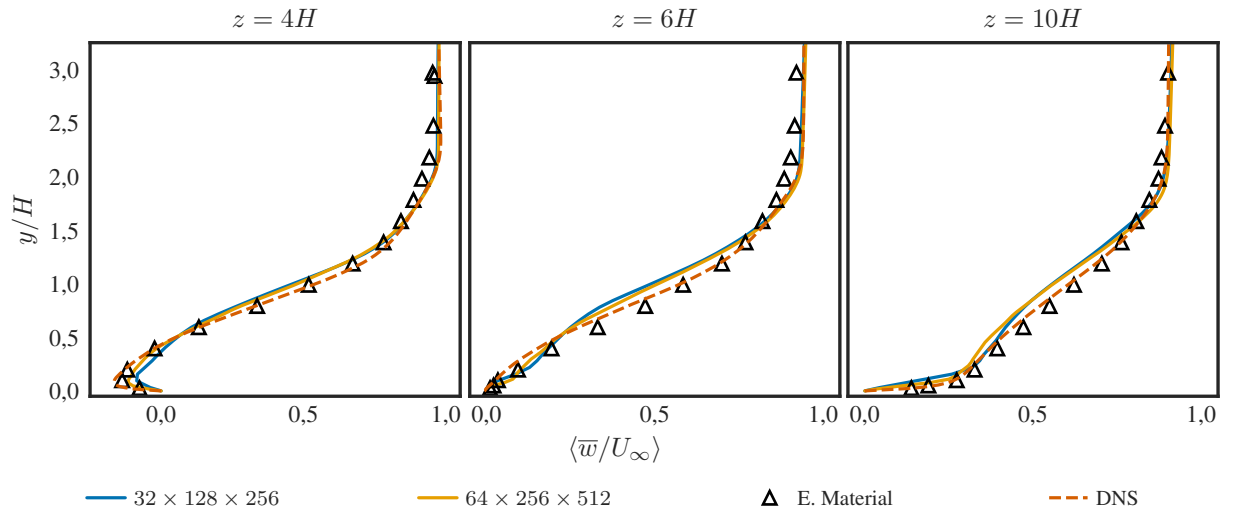
WALE, obtido sem necessidade de funções de amortecimento empíricas ou procedimentos dinâmicos³⁰.

Quando se analisa os perfis verticais médios de velocidade, $\langle \bar{w}/U_\infty \rangle$, dos resultados computacionais em relação aos respectivos resultados materiais e numéricos de referência em $x = 2H$ e $z = 4H$, $z = 6H$ e $z = 10H$, Fig. 62, constata-se que os resultados gerais são visualmente similares aos do modelo de Smagorinsky com uma significativa diferença em $z = 6H$, nenhuma das malhas simuladas apresenta recirculação nessa região, similar às referências.

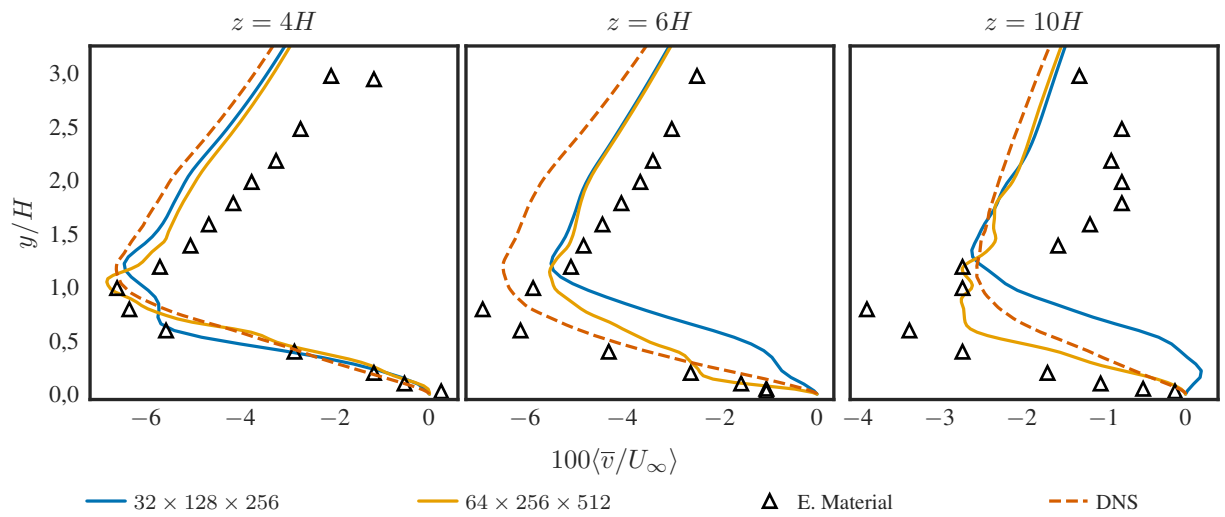
Após a análise do componente da velocidade média na direção do escoamento, analisa-se a componente média perpendicular a parede, $\langle \bar{v}/U_\infty \rangle$, nas mesmas condições, Fig. 63. Através da figura, nota-se uma concordância substancial entre os valores das duas malhas com os valores do trabalho dito DNS.

Em contrapartida, os perfis verticais de $\langle \bar{v}/U_\infty \rangle$ das malhas executadas se diferem dos resultados de referência em $z = 6H$ e $z = 10H$ ainda que pode-se falar que chegou-se a valores próximos, principalmente tendo em conta que os valores dos gráficos são multiplicados por uma fator de 100.

Na Figura 64 apresenta-se os perfis verticais do componente $\langle \bar{w}'^2/U_\infty^2 \rangle$ do tensor médio de Bussinesq-Reynolds nas regiões de análisadas. Nota-se aqui mais uma vez a superioridade dos resultados desse modelo submalha em relação ao modelo de Smagorinsky. Os perfis da malha mais refinada, $64 \times 256 \times 512$ pontos de colocação, estão expressivamente e mais próximos aos

Figura 62 – Perfis Verticais de $\langle \bar{w}/U_\infty \rangle$ para o modelo WALE.

Fonte: Próprio autor.

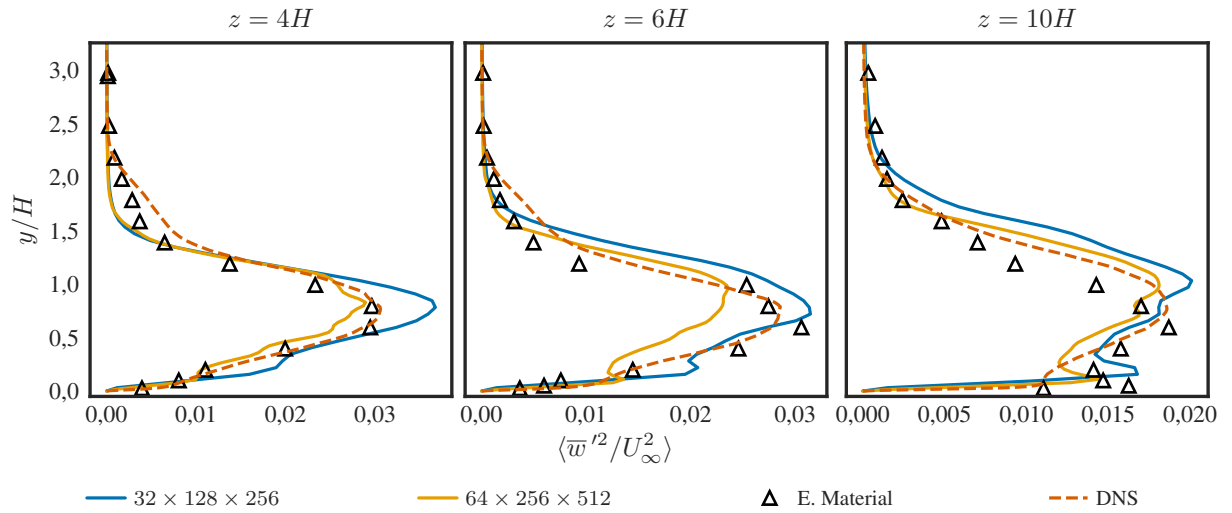
Figura 63 – Perfis Verticais de $\langle \bar{v}/U_\infty \rangle$ para o modelo WALE.

Fonte: Próprio autor.

trabalhos de referências em dois dos três perfis, $z = 4H$ e $z = 10H$.

Para o perfil médio em $6H$, nota-se que o refinamento de malha proporcionou resultados subestimados em relação as referências. Destaca-se que essa é a situação oposta aos resultados do modelo considerado na seção anterior, Fig. 57, onde a malha mais refinada apresentou resultados superestimados na região $H \leq y \leq 1,5H$.

Figura 64 – Perfis verticais de $\langle \overline{w}^2 / U_\infty^2 \rangle$ para o modelo WALE.



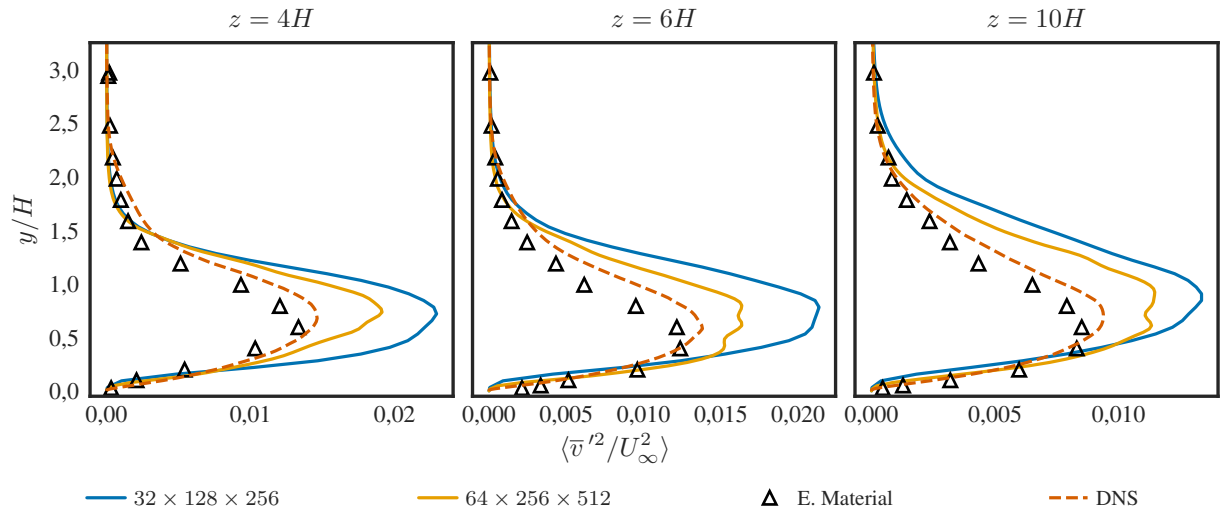
Fonte: Próprio autor.

Os resultados para os perfis médios do componente $\langle \overline{v}^2 / U_\infty^2 \rangle$ do tensor de Bussinesq-Reynolds são vistos na Fig. 65 para as malhas simuladas com o modelo WALE. Os resultados para as três posições z tem um mesmo comportamento de resultados superdimensionados anteriores em relação as referências com uma aproximação melhor com o refinamento de malha global. Salienta-se, entretanto, que os valores apresentados para o modelo submalha considerado estão mais próximos das referências dos que os resultados do modelo de Smagorinsky, mais uma vez destacando os resultados melhores do modelo WALE.

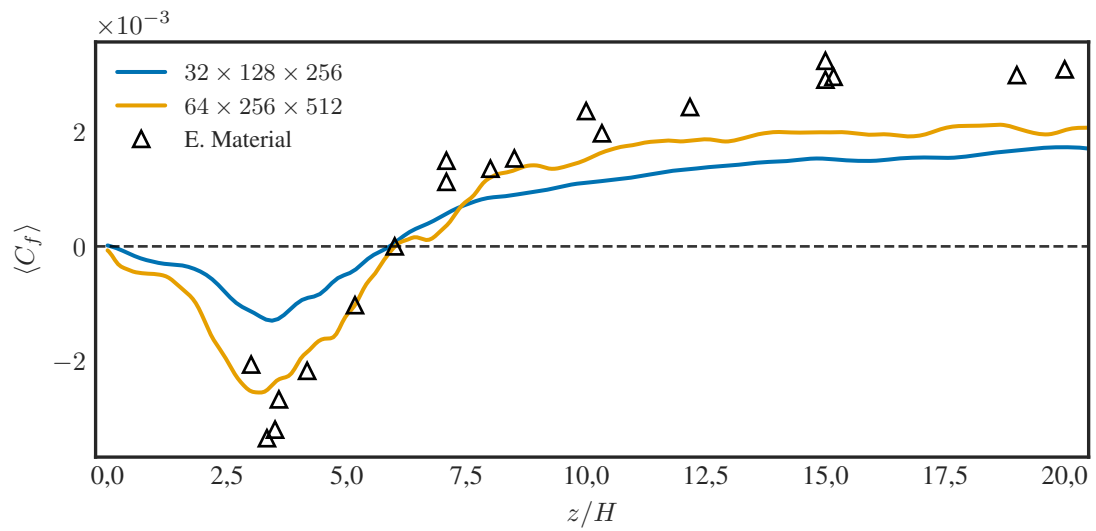
Em relação a determinação do ponto de recolamento da camada limite em cada malha simulada, isso pode ser feito a partir das distribuições do coeficiente médio de atrito, $\langle C_f \rangle$. A Figura 66 ilustra as distribuições de cada malha simulada junto aos resultados materiais. Nota-se em ambas as malhas simuladas, pontos de recolamento próximos ao valor material. A malha de $32 \times 128 \times 256$ pontos de colocação apresentou um ponto de recolamento igual a $5,94H$ enquanto a malha mais refinada de $64 \times 256 \times 512$ pontos de colocação teve o recolamento em $6,03H$.

Considerando o valor proveniente da malha mais refinada, tem-se um erro de 0,5%. Mais uma vez nota-se a significativa exatidão desse modelo submalha com fenômenos relacionados a presença de parede no escoamento turbulento considerado.

O comportamento LES nas duas malhas simuladas pode ser confirmado através das

Figura 65 – Perfis verticais de $\langle \bar{v}'^2 / U_\infty^2 \rangle$ para o modelo WALE.

Fonte: Próprio autor.

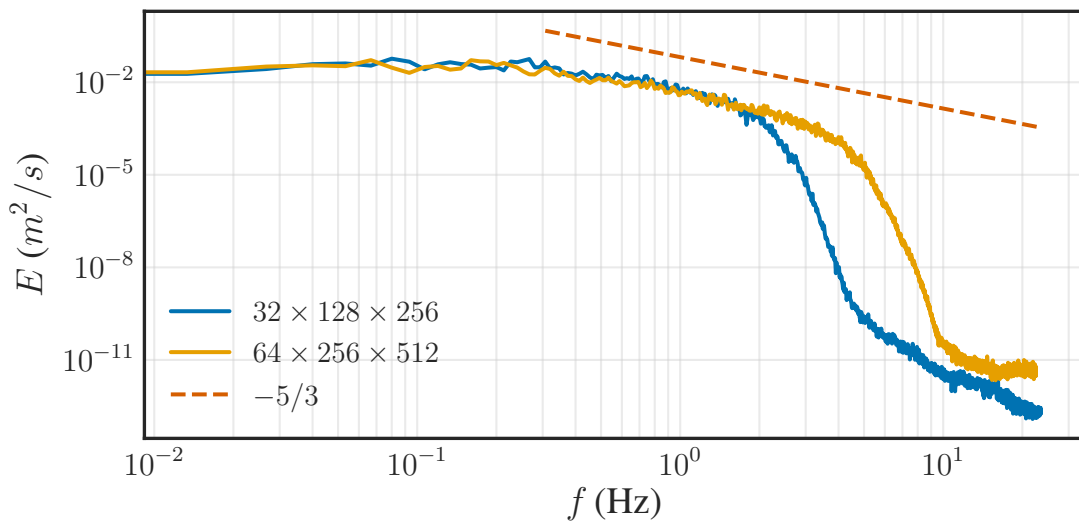
Figura 66 – Distribuições de $\langle C_f \rangle$ para o modelo WALE.

Fonte: Próprio autor.

densidades espectrais de energia cinética turbulenta $E(x_i, f)$, Fig. 67, determinadas a partir de uma sonda no ponto $(2H, H, 10H)$ com frequências de amostragem de 47,1 Hz e 45,0 Hz para, respectivamente, as malhas $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos de colocação.

Ambas as simulações com malhas distintas apresentam o mesmo nível energético nas baixas frequências, comportamento esperado em LES, onde as grandes estruturas são resolvidas diretamente e independem do modelo SGS adotado. Ambos os espectros apresentam região inercial com decaimento compatível com a lei de Kolmogorov $E(f) \propto f^{-5/3}$, confirmando o caráter físico das simulações. O efeito do refinamento de malha é evidente: a malha $64 \times 256 \times 512$ apresenta região inercial mais extensa, com frequência de corte deslocada para frequências mais altas em relação à malha $32 \times 128 \times 256$, confirmando que o filtro LES está associado ao espaçamento de malha Δ , e não ao modelo SGS, e que a malha mais refinada resolve diretamente uma faixa mais ampla da cascata energética.

Figura 67 – $E(x_i, f)$ para as diferentes malhas computacionais com modelo WALE.



Fonte: Próprio autor.

Em relação ao custo computacional do modelo WALE na malha mais refinada considerando a sua execução na GPU A100, obteve-se um valor igual a 39 h e 54 min. Esse valor é inferior ao modelo clássico de Smagorinsky, portanto, o modelo WALE apresenta maior exatidão com um custo computacional inferior.

6.6 Modelo Submalha Dinâmico de Germano-Lilly

Nesta seção serão avaliados os resultados relacionados ao modelo funcional de fechamento submalha dinâmico Germano-Lilly, proposto por Germano *et al.*¹ e Lilly². O primeiro destaque desse modelo submalha é a escolha da natureza do filtro teste. Em métodos pseudo-

espectrais com *dealiasing*, o filtro de truncamento corresponde ao corte espectral de Fourier, e nesse caso o filtro regular e o filtro de truncamento comutam exatamente^{Winckelmans2001}.

Seguindo essa consideração, foi adotado inicialmente o filtro exponencial com $\tilde{\Delta} = 2\Delta$ como filtro teste. Essa escolha, no entanto, gerou um campo de viscosidade efetiva ν_{eff} ruidoso, associado às oscilações numéricas decorrentes do corte abrupto do filtro exponencial próximo ao número de onda de corte.

Embora Sagaut²⁸ recomende que os filtros sejam da mesma natureza no procedimento dinâmico, a característica particular da combinação pseudoespectral com MDFM no IMERSPEC exigiu a adoção do filtro gaussiano como filtro teste. O filtro gaussiano, com $\tilde{\Delta} = 2\Delta$, apresenta decaimento suave no espaço espectral, $\hat{G}(k) = \exp(-k^2 \tilde{\Delta}^2/24)$, sem corte abrupto, evitando oscilações numéricas associadas ao filtro exponencial²⁸. Embora essa escolha constitua uma aproximação ao procedimento dinâmico estrito, sua validade é confirmada a posteriori pelos resultados apresentados nesta seção.

Com os parâmetros de simulação, Tab. 11, a condição de topo de simetria, o processo de *dealiasing* por filtragem exponencial simulou-se as duas malhas computacionais, $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos de colocação, para o modelo dinâmico de Germano. A vista lateral da magnitude da vorticidade, $\|\vec{\omega}\|$, coeficiente dinâmico do modelo, C_d e a viscosidade efetiva adimensionalizada, ν_{eff}/ν são apresentadas na Fig 68 para a malha mais refinada no tempo físico adimensional igual a 900.

Analisando a magnitude da vorticidade dos resultados computacionais, tem-se um resultado qualitativo próximo aos obtido para os demais modelos submalha. O desenvolvimento temporal da simulação com o modelo dinâmico de Germano-Lilly para a malha com $64 \times 256 \times 512$ pontos de colocação representado pelo critério Q com isovalor de 2,75 colorido pela magnitude da velocidade é apresentado através do vídeo *MP4_DynSMGK.mp4*¹⁰². A análise qualitativa do vídeo mostra novamente um comportamento similar ao apresentado para os modelos submalhas anteriores com destaque para estruturas turbilhonares mais definidas do que os observados para o modelo submalha de Smagorinsky clássico.

Por outro lado, quando se analisa o campo bidimensional C_d nota-se uma variação do coeficiente do modelo submalha de acordo com a física do escoamento modelado, comportamento dinâmico, não visto para o modelo de Smagorinsky ou WALE. Destaca-se que o campo C_d é bidimensional por uma consequência direta dos processos de média realizados na direção homogênea do escoamento, *i.e.*, direção x a fim de se ter estabilidade computacional.

O coeficiente dinâmico C_d apresenta oscilações em duas regiões distintas do domínio físico: adjacente à zona de *buffer* e próximo à condição de contorno de topo. Esse comportamento oscilatório decorre de dois mecanismos relacionados.

O primeiro está associado aos gradientes abruptos de velocidade introduzidos nessas regiões, pelo termo fonte do *buffer* na direção z e pela condição de simetria no topo do domínio,

que geram energia em altos números de onda no espectro. O segundo mecanismo está relacionado ao filtro teste gaussiano, aplicado globalmente no espaço espectral: ao filtrar o campo sobre regiões com gradientes abruptos, naturalmente, o C_d calculado nessas interfaces torna-se também ruidoso.

Destaca-se que a média nas direções homogêneas, empregada na estabilização do coeficiente dinâmico², não elimina completamente essas oscilações nas fronteiras do domínio, onde a homogeneidade estatística é localmente comprometida pela imposição das condições de contorno. Entretanto, as oscilações permanecem restritas às regiões de fronteira, como evidenciado na Fig. 68. Através da figura, é claro que o campo de C_d na região de interesse do escoamento turbulento sobre o degrau ($y \leq 3H$, $2H \leq z \leq 22H$), se comporta conforme o esperado para a modelagem da física envolvida.

Salienta-se que realizando uma média espacial do campo bidimensional C_d , chega-se a um valor, aproximadamente, igual a 0,0087. Sabendo-se que os coeficientes dinâmicos do presente modelo de fechamento e o coeficiente estático do modelo de Smagorinsky clássico se relacionam por uma potência de 2, *i.e.*, $C_d = C_s^2$. O valor médio de C_d equivale a um valor de C_s igual, a aproximadamente, 0,093. Um valor muito próximo ao adotado para o modelo clássico de Smagorinsky com funções de Van Driest explorado na seção 6.4.

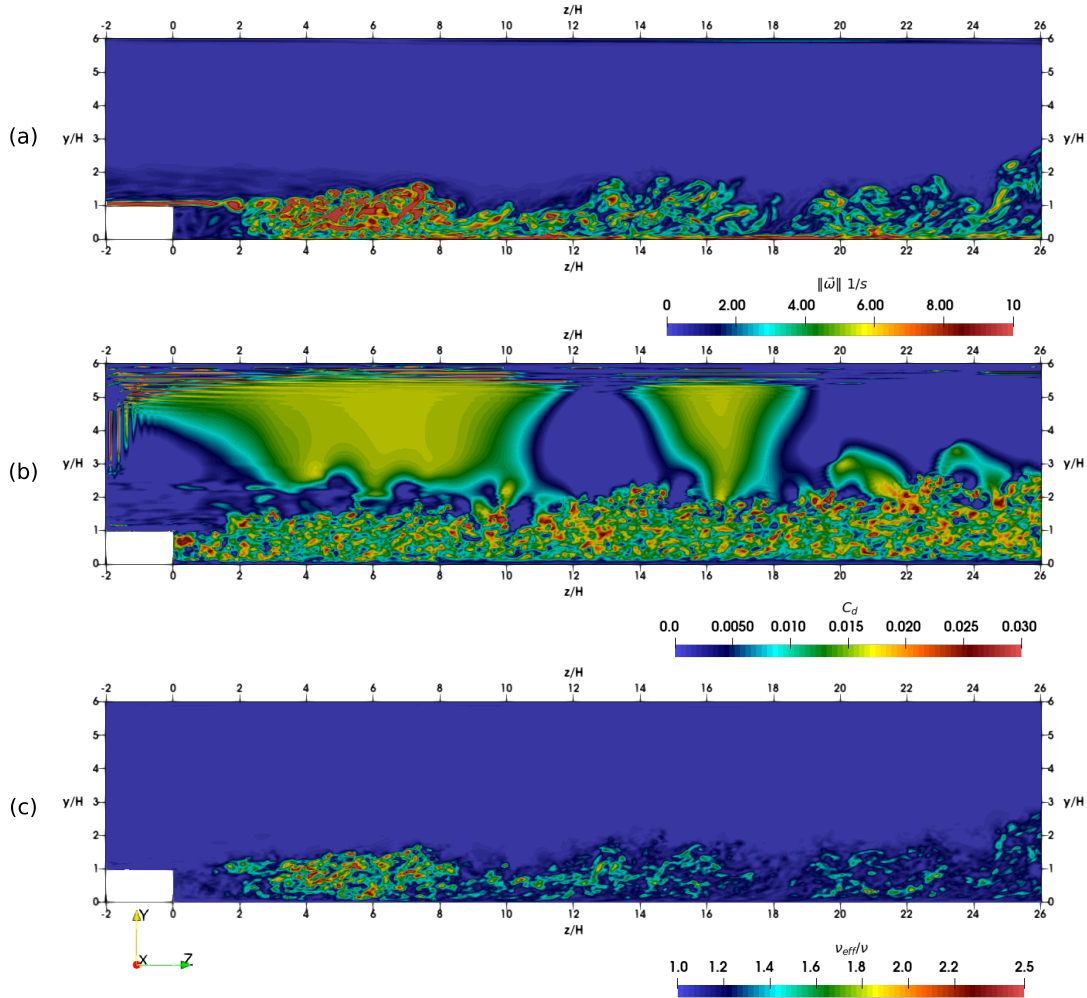
Após a análise do campo de C_d , tem-se a viscosidade efetiva adimensionalizada, ν_{eff}/ν , também presente na Fig. 68. Apenas nas regiões onde existe significativa vorticidade há valores de ν_{eff} diferentes de ν . Isso pode causar dúvida, visto que o campo de C_d apresenta valores oscilatórios próximos às regiões em que se aplica condições de contorno e valores diferentes de zero em regiões inerciais do escoamento. Entretanto, deve-se lembrar que o modelo dinâmico de Germano modela ν_{SGS} a partir do produto do comprimento característico da malha, C_d e a magnitude do tensor taxa de deformação. Nas regiões inerciais do escoamento ou àquelas onde notou-se comportamento oscilatório de C_d , tem-se ν_{SGS} desprezível devido a $\|\bar{S}\|$ desprezível. Mais uma vez, destaca-se que embora não desejável, o comportamento oscilatório visto para C_d nesse modelo submalha, não compromete a modelagem física do fenômeno.

Perfis verticais da velocidade média $\langle \bar{w}/U_\infty \rangle$ para as diferentes malhas computacionais simuladas em $x = 2H$ juntos aos perfis de referência são vistos na Fig. 69 em $z = 4H$, $z = 6H$ e $z = 10H$. Os resultados computacionais, em geral, mostram boa concordância com os resultados de referência nos 3 perfis verticais e, acima de tudo, para ambas as malhas simuladas. Porém, observa-se uma tendência a aproximação dos resultados numéricos de Le; Moin; Kim¹⁷ maior do que os materiais de Jovic; Driver¹⁶.

A Figura 70 é analisada em seguida, apresentando os perfis médios $\langle \bar{v}/U_\infty \rangle$, nas mesmas posições consideradas para a Fig. 69. As três posições indicam concordância do refinamento de malha para com os resultados ditos DNS, especialmente para a região $y \geq H$.

Por outro lado, para as regiões com $y \leq H$, principalmente em $z = 4H$ e $z = 6H$ tem-se

Figura 68 – Vista lateral de $\|\vec{\omega}\|$ (a), C_d (b) e ν_{eff}/ν (c) relativos ao modelo de fechamento submalha de Smagorinsky dinâmico em $t_a = 900$ com $64 \times 256 \times 512$ pontos de colocação.

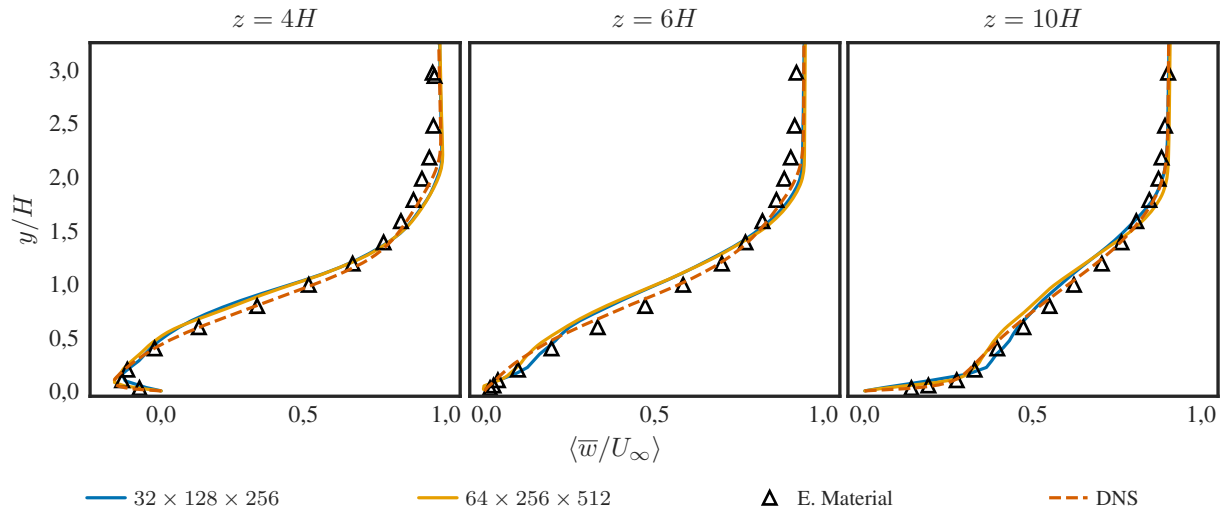


Fonte: Próprio autor.

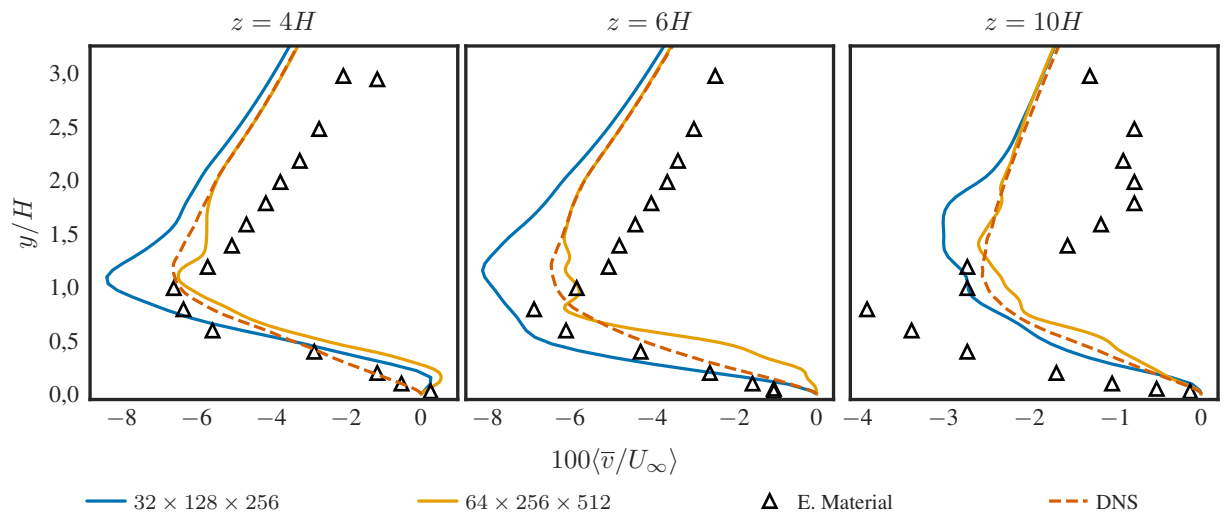
uma tendência a resultados superestimados em relação aos resultados de Le; Moin; Kim¹⁷. A partir desse comportamento e as vistas laterais da Fig. 68, infere-se que esse modelo submalha dinâmico apresenta melhor exatidão quando já ocorreu o recolamento da camada limite, ou seja, $z \geq 6H$.

Após a análise dos componentes da velocidade média, tem-se os perfis verticais da componente do tensor de Boussinesq-Reynolds $\langle \overline{w'^2}/U_\infty^2 \rangle$. Nota-se, em geral, ligeira melhor concordância dos resultados computacionais com o modelo dinâmico de Germano do que os respectivos perfis apresentados na Fig. 57 para o modelo de Smagorinsky, mas não tão bons quanto os resultado do modelo WALE.

De toda forma, os efeitos de refinamento de malha para a intensidade turbulenta na direção z são visualmente próximos aos obtidos utilizando o modelo de Smagorinsky clássico, mesmo que esse resultado computacional tenha sido obtido com um custo computacional superior

Figura 69 – Perfis Verticais de $\langle \bar{w}/U_\infty \rangle$ para o modelo dinâmico de Smagorinsky.

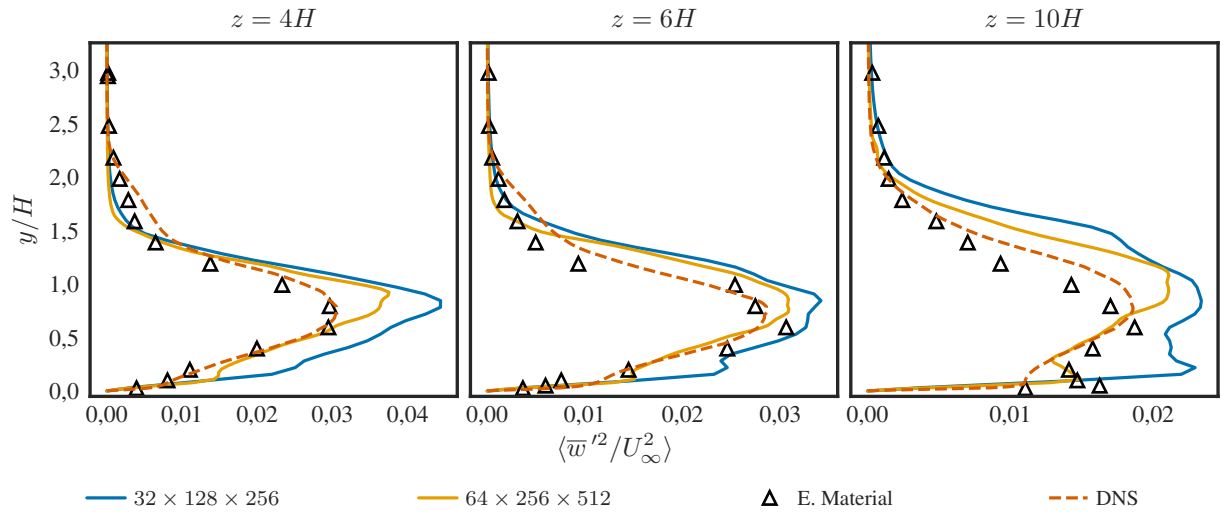
Fonte: Próprio autor.

Figura 70 – Perfis Verticais de $\langle \bar{v}/U_\infty \rangle$ para o modelo dinâmico de Smagorinsky.

Fonte: Próprio autor.

ao modelo clássico.

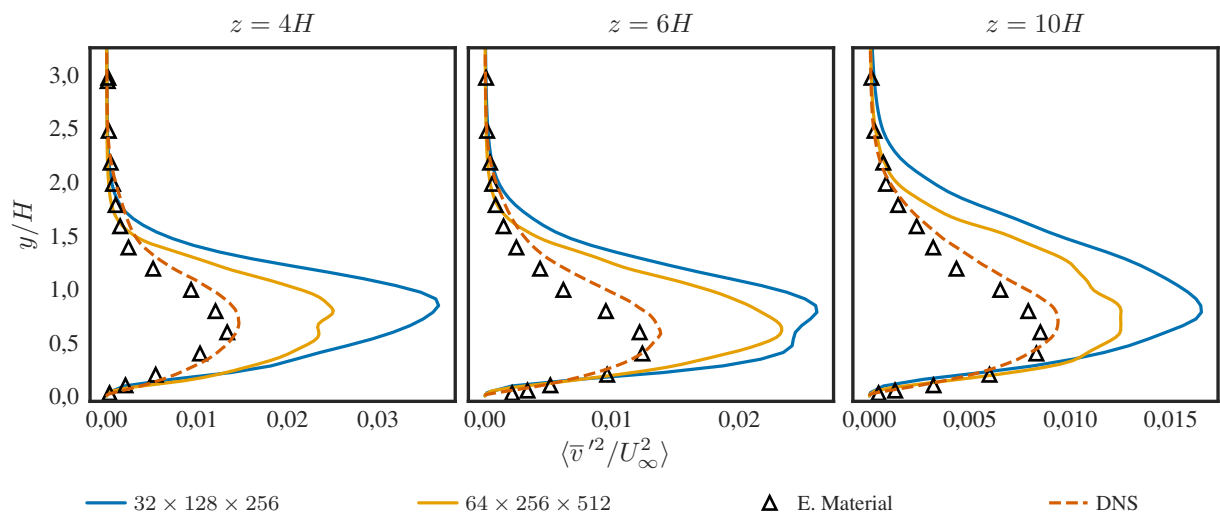
Figura 71 – Perfis verticais de $\langle \bar{w}'^2 / U_\infty^2 \rangle$ para o modelo dinâmico de Smagorinsky.



Fonte: Próprio autor.

Considerando o componente médio do tensor de Boussinesq-Reynolds $\langle \bar{v}'^2 / U_\infty^2 \rangle$ para os mesmos perfis verticais, tem-se a Fig. 72. Os componentes obtidos computacionalmente ainda são bem discrepantes em relação a qualquer dos resultados de referência. Salienta-se, entretanto, que a malha grosseira, apresenta resultados mais atenuados do que a sua contraparte para o modelo de Smagorinsky clássico, Fig. 58, e piores do que o modelo WALE, Fig. 65.

Figura 72 – Perfis verticais de $\langle \bar{v}'^2 / U_\infty^2 \rangle$ para o modelo dinâmico de Smagorinsky.



Fonte: Próprio autor.

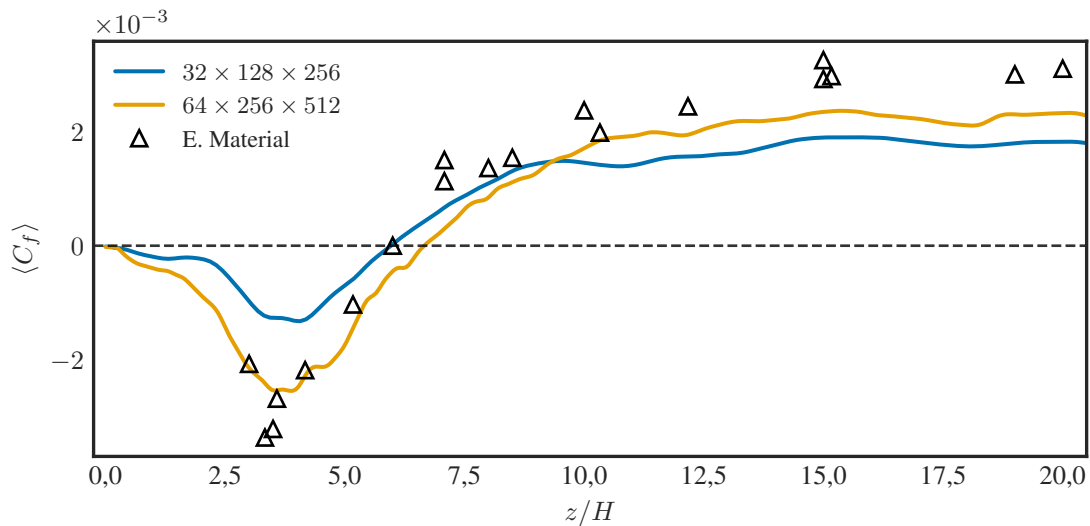
Novamente, o ponto de recolamento de cada uma das malhas simuladas para o presente modelo de fechamento da turbulência é determinado através da distribuição do coeficiente médio

de atrito, $\langle C_f \rangle$. A distribuição dessa grandeza média para as diferentes malhas simuladas e a distribuição dos resultados materiais são apresentados na Fig. 73. Nota-se que mesmo com o refinamento de malha, os resultados computacionais não são exatamente iguais aos materiais, comportamento análogo ao visto nas Fig. 59 e 66.

Quanto ao ponto de recolamento, identifica-se os valores de $5.94H$ e $6.66H$ para, respectivamente, a malha de $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos de colocação. Quando se compara os resultados numéricos com o material de $6H$, identifica-se um aumento do erro para 11% a medida que houve o refinamento de malha. Sugere-se novamente uma tendência para um valor do ponto de recolamento de camada limite diferente do material a partir da convergência de malha. Entretanto, vale salientar que o modelo funcional de fechamento submalha de Smagorinsky dinâmico consegue reduzir a porcentagem do erro em relação ao modelo clássico mesmo que não seja tão exato quanto o modelo WALE que apresentou erro de 0,5%.

Essa redução de erro em relação ao modelo submalha de Smagorinsky é uma consequência direta do procedimento dinâmico do modelo visto que os coeficientes são determinados para o comportamento local do escoamento. Ainda sim, não se obteve o valor material através das simulações. Infere-se que a não modelagem do *backscatter*, aliados a clipagem e médias na direção homogênea limitou ao potencial teórico máximo desse modelo submalha mesmo que essas ações foram realizadas para tornar o modelo computacional possível.

Figura 73 – Distribuições de $\langle C_f \rangle$ para o modelo de Smagorinsky dinâmico.



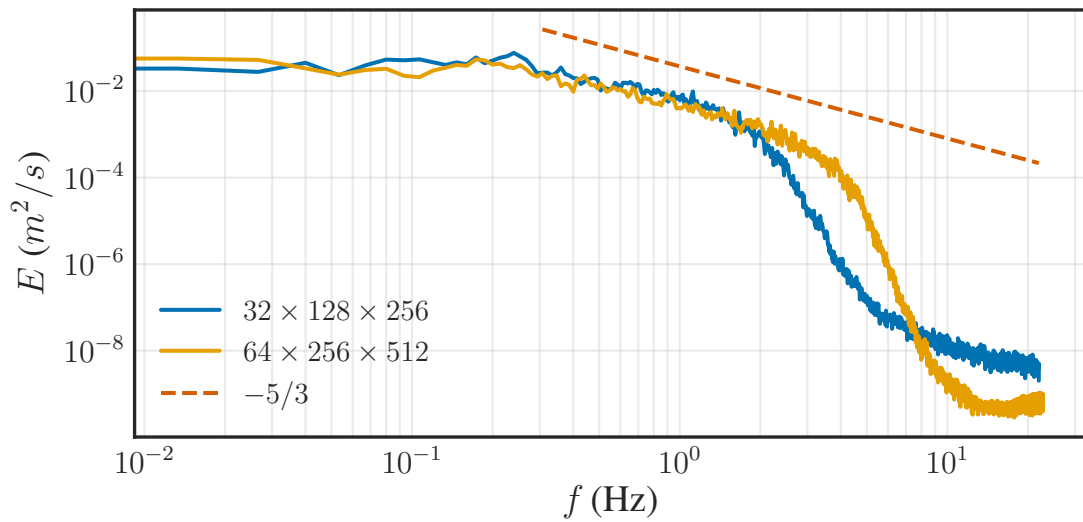
Fonte: Próprio autor.

Novamente, foram determinadas a distribuição das densidades espectrais de energia cinética turbulenta, $E(x_i, f)$, a partir de uma sonda em $(2H, H, 10H)$ para as duas malhas computacionais simuladas com o modelo dinâmico de Germano-Lilly, Fig. 74. As frequências de amostragem de cada malha foram, respectivamente, 43,9 e 45,5 Hz.

Análoga aos espectros para o modelo de Smagorinsky clássico e WALE, as regiões iner-

ciais seguindo a lei de Kolmogorov também são vistas na Fig. 74 em que a malha mais refinada apresenta região inercial maior visto que resolve mais estruturas turbilhonares. Diferentemente do efeito de refinamento de malha da seção 6.4, a malha mais grosseira e a refinada apresentam o mesmo nível energético nas baixas frequências indicando que não há acúmulo de energia nas grandes estruturas.

Figura 74 – $E(x_i, f)$ para as diferentes malhas computacionais com Modelagem Dinâmica de Smagorinsky.



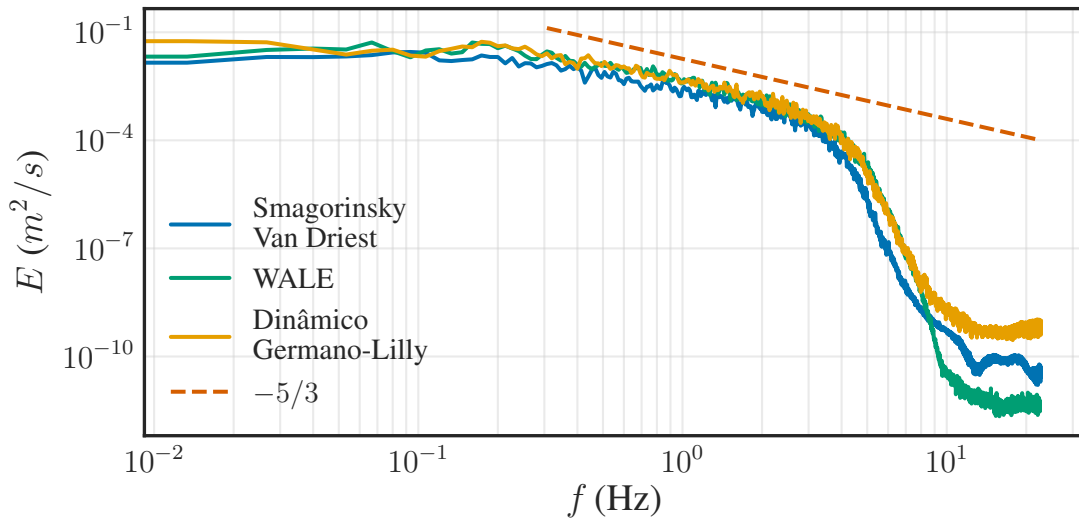
Fonte: Próprio autor.

Por outro lado, quando se compara a densidade espectral de energia cinética turbulenta dos modelos de fechamento implementados no presente trabalho, a partir da sonda na mesma posição e considerando a malha $64 \times 256 \times 512$ pontos de colocação, obtém-se a Fig. 75.

Os três espectros são praticamente equivalentes nas baixas e médias frequências, resultado que merece atenção sob duas perspectivas. Do ponto de vista da malha, a equivalência dos espectros confirma que o filtro LES é determinado pelo espaçamento de malha Δ , e não pelo modelo SGS, uma vez que os três modelos apresentam frequência de corte idêntica. Isso é fisicamente consistente com a teoria LES, onde a malha define a separação entre escalas resolvidas e modeladas²⁸.

Do ponto de vista dos modelos, a equivalência espectral indica que, na região de escoamento analisada, os três modelos produzem níveis de transformação viscosa de submalha comparáveis nas frequências energeticamente dominantes. Para o modelo de Smagorinsky clássico, esse comportamento é esperado quando a constante C_s está adequadamente calibrada, de modo que C_d calculado pelo procedimento de Germano-Lilly converge para valores próximos a C_s^2 fato verificado pela comparação do valor médio espacial de C_d com C_s visto anteriormente nessa seção.

Figura 75 – Comparação de $E(x_i, f)$ utilizando a malha com $64 \times 256 \times 512$ pontos de colocação para os modelos submalha implementados.



Fonte: Próprio autor.

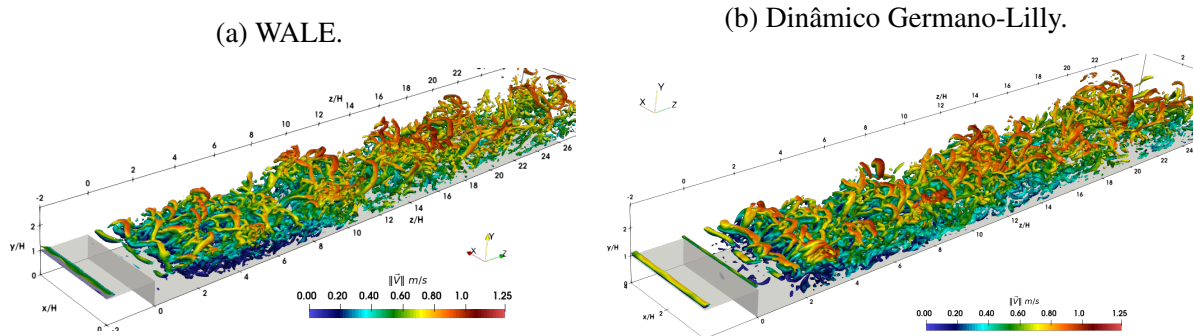
Diferenças entre os modelos tornam-se visíveis nas altas frequências ($f > 5$ Hz). O modelo WALE apresenta o decaimento mais abrupto nessa região, indicando maior transformação viscosa submalha nas frequências próximas ao corte espectral, comportamento consistente com a maior atividade de ν_{SGS} nas regiões de intensa interação entre deformação e rotação, capturadas pelo tensor $\bar{G}_{ij}^d$. O modelo dinâmico de Germano-Lilly apresenta decaimento ligeiramente mais pronunciado do que o Smagorinsky clássico, consistente com o ajuste local de C_d em regiões de maior atividade turbulenta.

A equivalência espectral na região de altas frequências, combinada com as diferenças observadas no ponto de recolamento ($z_R/H = 6,78, 6,66$ e $6,03$ para Smagorinsky clássico, dinâmico de Germano-Lilly e WALE, respectivamente), sugere que a distinção entre os modelos manifesta-se principalmente na dinâmica da camada cisalhante separada próxima ao degrau, região sensível ao comportamento do modelo SGS na parede, e não na estrutura espectral do escoamento desenvolvido a jusante. Nesse contexto, o resultado do modelo WALE ($z_R/H = 6,03$) é significativo pela proximidade ao valor material de referência $z_R/H = 6,0$, demonstrando a vantagem do comportamento assintótico correto $\mathcal{O}(y^{+3})$ desse modelo na predição da dinâmica de separação e recolamento.

O comportamento semelhante nas baixas frequências explicitado na Fig. 75 para os modelos submalha WALE e dinâmico de Germano-Lilly pode ser visto qualitativamente na Fig. 76. Nessas figuras são representados os dois escoamentos através do critério Q com isovalor igual a 2,75, coloridas pela magnitude da velocidade dos respectivos escoamentos, e calculadas para a malha de $64 \times 256 \times 512$ pontos de colocação em $t_a = 900$. Apesar de não se ter resultados exatamente iguais, os escoamentos modelados são turbulentos, percebe-se que os dois campos de

velocidade são semelhantes com estruturas turbilhonares mais bem definidas do que as estruturas da Fig. 53 para o modelo submalha de Smagorinsky com funções de Van Driest.

Figura 76 – Critério Q (2,75) em $t_a = 900$ de dois modelos submalha com a malha de $64 \times 256 \times 512$ pontos de colocação.



Fonte: Próprio autor.

Considerando a execução do código IMERSPEC CUDA na GPU A100, teve-se um custo computacional igual a 71 h e 45 min para o modelo submalha dinâmico de Germano-Lilly. Assim, esse modelo apresentou um custo computacional 8% maior do que o modelo clássico de Smagorinsky apresentando resultados computacionais comparáveis e piores do que o modelo WALE que apresentou o menor custo computacional dos modelos implementados.

No presente capítulo foi apresentado as modificações necessárias ao código para a correta modelagem de escoamentos turbulentos pela abordagem LES junto a metodologia IMERSPEC e os respectivos resultados das implementações de modelos funcionais de fechamento submalha no código IMERSPEC CUDA. Constatou-se que a forma de discretização antissimétrica do termo difusivo não foi suficiente para a eliminação de fenômeno de *aliasing* em modelagens de escoamentos turbulentos e assim foi necessário a introdução de processos de *dealiasing* em todo o código computacional. Constatou-se que entre a regra clássica de 2/3 e um processo de *dealiasing* por filtragem exponencial, esta última alternativa proporcionaria os melhores resultados por ser uma filtragem mais suave, desejável em um contexto de uso do MPF.

Adicionalmente, analisou-se qual seria a melhor condição de topo do domínio físico para proporcionar a melhor reprodutibilidade de fenômeno físico turbulento. A condição de simetria foi a que apresentou o melhor custo benefício das condições alternativas. Em seguida, foi feita as análises e discussões do modelos submalha implementados: Smagorinsky clássico com funções de amortecimento, WALE e dinâmico de Germano-Lilly.

Ao analisar os resultados dos dois modelos com trabalhos de referência, material e DNS, para os modelos implementados, constatou-se que o modelo WALE é uma alternativa melhor. Este modelo apresentou os resultados computacionais mais exatos em relação aos outros modelo com menor custo computacional. Ao final do capítulo, nota-se que apesar de não se ter correspondência dos resultados computacionais com os trabalhos de referência, principalmente os materiais, a metodologia IMERSPEC foi capaz de reproduzir com significativa fidelidade um

escoamento complexo como o escoamento turbulento sobre o degrau, o primeiro escoamento turbulento tridimensional com a presença de paredes a ser modelado por essa metodologia. No próximo capítulo, serão exploradas as conclusões da presente tese.

7 CONCLUSÕES

O presente trabalho teve como objetivo geral melhorar a funcionalidade e a performance da metodologia IMERSPEC tridimensional, ao mesmo tempo que se propôs a modelagem de escoamentos tridimensionais, laminares e turbulentos, com a presença de paredes. Assim, após a apresentação e discussão dos resultados, pode-se afirmar que o objetivo geral foi alcançado. Um escoamento complexo como o escoamento sobre um degrau foi adequadamente modelado com uma metodologia de alta ordem. Em termos de performance, para a modelagem laminar utilizando uma paralelização GPGPU, conseguiu-se uma redução no custo computacional de, aproximadamente, vinte e nove vezes menor em relação a um código paralelizado por MPI em duas das três direções com 64 processos concomitantes. A performance do código IMERSPEC CUDA fica ainda mais expressiva ao se considerar um custo computacional de 71 *h* e 45 *min* referente a uma malha computacional de $64 \times 256 \times 512$ pontos colocantes executada na GPU A100 para modelagem de 450 *s* físicos por uma abordagem LES empregando-se um modelo funcional de fechamento de turbulência submalha complexo como o dinâmico de Germano-Lilly. Em outras palavras, obtém-se resultados de uma simulação de alta ordem com uma abordagem LES em menos de 3 dias.

Por meio do objetivo geral, portanto, foi possível responder afirmativamente ao questionamento sobre se o IMERSPEC seria capaz de modelar escoamentos turbulentos e tridimensionais na presença de paredes. A presente tese vai além ao demonstrar um custo computacional inferior a três dias para a modelagem do *benchmark* turbulento, para todos os modelos submalha apresentados.

Entretanto, para que o objetivo geral fosse alcançado, os objetivos específicos foram vitais. Conseguiu-se, satisfatoriamente, desenvolver dois códigos computacionais com distintas formas de paralelização. O IMERSPEC MPI foi desenvolvido explorando a paralelização através da biblioteca MPI que distribui a carga computacional entre núcleos com memória independentes entre si. Por outro lado, o IMERSPEC CUDA foi criado sob a perspectiva da paralelização heterogênea entre CPU e GPU com os cálculos computacionais sendo realizados exclusivamente na placa de vídeo em *threads*, aproveitando assim o ambiente massivamente paralelo do dispositivo.

Ambos os códigos computacionais foram verificados computacionalmente através do método de soluções manufaturadas. O IMERSPEC MPI apresentou exatidão espectral na verificação pura do MPF e na do MPF/fronteira imersa coincidente, mesmo que nessa última verificação tenha sido constatada erros de cancelamento catastrófico devido a alta exatidão numérica do IMERSPEC. Em relação às mesmas verificações numéricas realizadas com o IMERSPEC CUDA identificou-se uma exatidão ainda espectral com precisão de erro de máquina, entretanto, menor quando comparada com os resultados do código IMERSPEC MPI devido a limitações

de operações de dupla precisão realizadas na placa de vídeo utilizada. A limitação da placa de vídeo só é identificada para resultados com precisão de erro de máquina.

Adicionalmente, identificou-se que o código IMERSPEC-GPU requer parâmetros específicos de configuração, número de *threads* por bloco e número de pontos processados por *thread*, dependentes da arquitetura de cada GPU utilizada, não sendo portáteis entre diferentes modelos de hardware. Por exemplo, a GPU GTX 1060 apresentou configuração ótima de 32 *threads* por bloco com processamento de um ponto por *thread*, enquanto a GPU A100 obteve melhor desempenho com $64N_{SM}$ *threads* totais, onde N_{SM} é o número de SM da GPU, organizados em blocos de 128 *threads* para essa placa. Constatou-se ainda que essa diferença entre as configurações ótimas de cada GPU está associada à quantidade de SMs de cada hardware, aliada ao fenômeno de ocultação de latência (*latency hiding*), que favorece configurações com maior número de *threads* ativas em GPUs com maior paralelismo de hardware.

Comparando os custos computacionais dos dois códigos para as verificações comentadas, notou-se uma performance superior do IMERSPEC CUDA, entretanto, reconheceu-se um possível potencial maximizador da performance do IMERSPEC MPI em um *cluster* com dezenas de processos.

Em relação a verificação numérica para fronteira não coincidente, ambos os códigos apresentaram a mesma exatidão, quarta ordem de convergência. Em relação aos custos computacionais, nota-se que a posição dos pontos lagrangianos limita a paralelização do código IMERSPEC MPI enquanto tal limitação não é vista no código IMERSPEC CUDA. O Código que utiliza GPU apresenta uma ligeira perda de performance por necessitar de comunicação entre *threads* para realização da interpolação/distribuição inerentes ao MDFM/MFI mas ainda sim performa melhor que o IMERSPEC MPI em todas as malhas simuladas com tendência a ter ganhos de velocidade computacional ainda maiores à medida que malhas com mais pontos colocantes são utilizadas.

A modelagem do escoamento laminar sobre o degrau, um *benchmark* clássico, foi implementada como forma de validação física nos dois códigos computacionais. No contexto MPI, as simulações do escoamento laminar sobre degrau, inicialmente sob uma perspectiva pseudo-3D, foram comparadas com um trabalho material onde determinou-se a malha com $4 \times 64 \times 2048$ pontos colocantes como aquela com melhor custo benefício. Essa malha apresentou custo computacional razoável quando comparado a um refinamento de malha global posterior e um erro significativo de 13,7%. Ao se estender a malha para um estudo 3D identificou-se exatamente a mesma exatidão dos resultados da malha pseudo-3D, indicando que a direção homogênea do escoamento e não paralelizada não impacta negativamente o IMERSPEC MPI.

No contexto da validação IMERSPEC CUDA, analisou-se o custo computacional de uma malha de $32 \times 64 \times 2048$ pontos colocantes para os dois códigos escritos na presente tese, notou-se a superior performance da paralelização GPGPU mantendo a mesma exatidão na reprodutibilidade do fenômeno físico.

Nessas modelagens teve-se que modelar a interação do escoamento com superfícies sólidas. Esse tipo de interação apresentava, até então, ondas oscilantes indesejadas periódicas a essas superfícies. Esse fenômeno numérico foi identificado e pareceu ser solucionado através de uma nova forma de discretização do termo difusivo que minimizou o problema de *aliasing*. Sendo assim, ao final da análise de verificação e validação dos dois códigos, a técnica de paralelização MPI em duas das três direções do escoamento foi descartada e posteriores desenvolvimentos para a modelagem turbulenta se concentraram no código com paralelização CPU/GPU.

Exclusivamente para o IMERSPEC CUDA, foi proposta uma metodologia alternativa ao MDFM, o MIED. Nessa abordagem buscava-se uma exatidão espectral na modelagem de malhas lagrangianas não coincidentes com as eulerianas através da realização dos processos de interpolação e distribuição realizados com a transformada de Fourier não uniforme (NUFFT). Os resultados numérico-computacionais indicam uma exatidão ainda menor e uma performance próxima a do MDFM com uma tendência a se igualarem a medida que ocorre sucessivos refinamentos de malha.

A nova forma de discretização do termo difusivo foi eficaz na modelagem dos escoamentos laminares mas ineficiente em um contexto turbulento com abordagem LES. A forma antissimétrica não elimina completamente o fenômeno numérico de *aliasing*, ela apenas o minimiza. Os altos números de onda não são desprezíveis em um escoamento turbulento, e assim, mesmo minimizados os demais números de onda são contaminados. Com esse resultado e considerando que a nova forma proposta teve maior custo computacional, retornou-se a forma do termo difusivo adotada previamente no IMERSPEC.

Em relação a eliminação de *aliasing*, no IMERSPEC aplicou-se processos de *dealising*, filtragem, ao longo de todo produto entre duas funções ao longo de todo o algoritmo. Isso impacta diretamente a abordagem LES por que traduz na computação equivalente do termo advectivo filtrado como $\overline{\bar{u}_i \bar{u}_j}$ ao invés de $\bar{u}_i \bar{u}_j$ conforme proposto por Germano²⁵.

Comparou-se dois processos de *dealising* distintos, a filtragem pela regra clássica de 2/3 e por um filtro exponencial. No contexto do MPF presente no IMERSPEC que pode apresentar o fenômeno de Gibbs caso haja mudanças bruscas no escoamento modelado, optou-se pelo filtro exponencial em sucessivas análises.

Também foi descoberto que as ondas $2h$ eram formadas a partir do contato próximo da zona de forçagem do domínio complementar com as condições de contorno de não deslizamento. Logo, se reestruturou o domínio complementar tornando obsoleta a zona de forçagem na modelagem do fenômeno. Isso foi feito acoplando diretamente a zona de forçagem dentro do RK46 com o ajuste dos valores máximos de Ψ mas mantendo a sua forma.

As condições de topo do domínio físico também foram analisadas. Examinou-se três tipos de condições distintas: a condição de simetria adotada no trabalho computacional de referência; a condição de transpiração adota com sucesso na modelagem da camada limite bidimensional

junto a metodologia IMERSPEC; e a condição de não deslizamento que exigiu a modelagem de todo o canal relativo ao trabalho experimental material de referência. Das três possibilidades, comparando os seus respectivos perfis de velocidade média e de intensidade turbulenta na direção de desenvolvimento do escoamento, a condição de simetria foi àquela que apresentou melhor custo benefício e por isso foi utilizada nas análises subsequentes.

Por fim, os modelos submalha implementados: Smagorinsky clássico, WALE e Dinâmico de Germano-Lilly foram analisados. Destaca-se que o modelo de Smagorinsky clássico necessitou de uma modificação *ad-hoc* devida à sua inexatidão próximo às paredes do escoamento, as funções de amortecimento de Van Driest.

Duas malhas computacionais foram executadas para cada modelo, $32 \times 128 \times 256$ e $64 \times 256 \times 512$ pontos colocantes, em que notou-se significativa exatidão dos valores de $\langle \bar{w}/U_\infty \rangle$ e parcial para $\bar{w}'^2/U_\infty^2$. Os componentes $\langle \bar{v}/U_\infty \rangle$ se aproximaram dos resultados numéricos, ditos DNS mas sem total congruência com os resultados materiais, mas foram discrepantes dos resultados materiais, sobretudo em $z = 10H$. O componente médio do tensor de Boussinesq-Reynolds para $\bar{v}'^2/U_\infty^2$ apresentou uma tendência de convergência aos valores de referência mas se mantiveram superestimados.

De forma geral, os modelos submalha de Smagorinsky e Dinâmico de Germano-Lilly apresentaram resultados computacionais próximos, entretanto, o modelo submalha dinâmico apresentou exatidão ligeiramente melhor. Os melhores resultados computacionais para as duas malhas, foram obtidos com o modelo WALE. O ponto de recolamento para a malha mais refinada é um exemplo disso. O modelo de Smagorinsky clássico com Van Driest apresentou um erro de 13%, o modelo de dinâmico estimou o ponto de recolamento para a mesma malha com um erro de 11% enquanto o modelo WALE teve um erro de 0,5%. Com o contexto de que o modelo submalha dinâmico apresenta um custo computacional maior e não apresenta os melhores resultados, constatou-se que o modelo WALE é a melhor escolha para o escoamento modelado junto a metodologia IMERSPEC visto que apresenta o menor custo computacional dos três com a melhor exatidão.

Salienta-se, entretanto, que o comportamento computacional próximo entre os modelos de Smagorinsky e dinâmico de Germano-Lilly é justificado principalmente pelo coeficiente dinâmico médio que é equivalente ao valor *ad-hoc* utilizado no modelo de Smagorinsky clássico.

7.1 Futuros Trabalhos

A partir dos resultados da presente tese, espera-se continuar o desenvolvimento da metodologia IMERSPEC. Em termos de performance, o próximo passo é a junção da filosofia de paralelização MPI junto com a paralelização GPU/CPU. Isso pode ser feito adaptando o código IMERSPEC CUDA para o uso de MPI CUDA-aware em que a computação do código é distribuída entre GPUs.

Outro possível avanço na metodologia do código é a modelagem da equação da energia no novo código. Trabalhos anteriores já realizaram essa implementação em um abordagem do IMERSPEC 2D⁵³ mas ainda não se implementou de forma tridimensional. Espera-se que com essa implementação o código torne-se uma ferramenta computacional mais completa para a modelagem de fenômenos físicos menos idealizados.

Seguindo a mesma lógica, pretende-se incluir modelos de interação fluido-estrutura na formulação tridimensional do IMERSPEC desenvolvido no presente trabalho. Esse modelos já foram também implementado em um contexto 2D e pretende-se agora a completa tridimensionalização de sua modelagem.

Por fim, outro ponto de melhoria a ser explorado em futuros trabalhos é a fronteira imersa. Enquanto o MPF que compõe o IMERSPEC apresenta exatidão de erro de máquina, o MFI/MDFM é um método de baixa ordem. Esperou-se que tornando toda a metodologia híbrida IMERSPEC em um espectral, MIED, a exatidão global fosse melhorada mas isso não ocorreu. Assim, deve-se buscar uma nova forma de melhoria da exatidão desse método. A fronteira imersa por interface discreta pode ser uma alternativa e são necessários testes para verificar sua compatibilidade com o MPF.

REFERÊNCIAS

- 1 GERMANO, Massimo; PIOMELLI, Ugo; MOIN, Parviz; CABOT, William H. A Dynamic Subgrid-Scale Eddy Viscosity Model. **Physics of Fluids A: Fluid Dynamics**, v. 3, n. 7, p. 1760–1765, jul. 1991. ISSN 0899-8213. DOI: <https://doi.org/10.1063/1.857955>. Citado nas p. ix, x, 9, 11, 13, 33, 37, 38, 40, 103, 124.
- 2 LILLY, D. K. A Proposed Modification of the Germano Subgrid-Scale Closure Method. **Physics of Fluids A: Fluid Dynamics**, v. 4, n. 3, p. 633–635, mar. 1992. ISSN 0899-8213. DOI: <https://doi.org/10.1063/1.858280>. Citado nas p. ix, x, 9, 11, 13, 33, 38, 40, 103, 124, 126.
- 3 REYNOLDS, Osborne. XXIX. An Experimental Investigation of the Circumstances Which Determine Whether the Motion of Water Shall Be Direct or Sinuous, and of the Law of Resistance in Parallel Channels. **Philosophical Transactions of the Royal Society of London**, n. 174, p. 935–982, dez. 1883. ISSN 0261-0523, 2053-9223. DOI: <https://doi.org/10.1098/rstl.1883.0029>. Citado na p. 1.
- 4 KIM, John; LEONARD, Anthony. The Early Days and Rise of Turbulence Simulation. **Annual Review of Fluid Mechanics**, v. 56, n. 1, p. 21–44, jan. 2024. ISSN 0066-4189, 1545-4479. DOI: <https://doi.org/10.1146/annurev-fluid-120821-025116>. Citado nas p. 1, 3.
- 5 DURAISAMY, Karthik; IACCARINO, Gianluca; XIAO, Heng. Turbulence Modeling in the Age of Data. **Annual Review of Fluid Mechanics**, Annual Reviews, v. 51, Volume 51, 2019, p. 357–377, jan. 2019. ISSN 0066-4189, 1545-4479. DOI: <https://doi.org/10.1146/annurev-fluid-010518-040547>. Citado na p. 1.
- 6 DURBIN, Paul A. Some Recent Developments in Turbulence Closure Modeling. **Annual Review of Fluid Mechanics**, Annual Reviews, v. 50, Volume 50, 2018, p. 77–103, jan. 2018. ISSN 0066-4189, 1545-4479. DOI: <https://doi.org/10.1146/annurev-fluid-122316-045020>. Citado nas p. 2, 4.
- 7 ORSZAG, Steven A. Numerical Simulation of Incompressible Flows within Simple Boundaries: Accuracy. **Journal of Fluid Mechanics**, v. 49, n. 1, p. 75–112, set. 1971. ISSN 0022-1120, 1469-7645. DOI: <https://doi.org/10.1017/S0022112071001940>. Citado na p. 3.
- 8 FERZIGER, Joel H; PERIĆ, Milovan; STREET, Robert L. **Computational Methods for Fluid Dynamics**. Switzerland AG: springer, 2019. ISBN 978-3-319-99693-6. DOI: <https://doi.org/10.1007/978-3-319-99693-6>. Citado na p. 3.

- 9 PESKIN, Charles S. Flow Patterns around Heart Valves: A Numerical Method. **Journal of Computational Physics**, v. 10, n. 2, p. 252–271, 1972. ISSN 0021-9991. DOI: [https://doi.org/10.1016/0021-9991\(72\)90065-4](https://doi.org/10.1016/0021-9991(72)90065-4). Citado nas p. 3, 14.
- 10 MARIANO, Felipe Pamplona. **Simulação de Escoamentos Não-Periódicos Utilizando as Metodologias Pseudo-Espectral e Da Fronteira Imersa Acopladas**. 2007. Diss. (Mestrado) – Universidade Federal de Uberlândia, Uberlândia. Citado nas p. 4, 17, 32, 65, 66.
- 11 MOREIRA, Leonardo de Queiroz. **Simulação de Grandes Escalas de Jatos Periódicos Temporais Utilizando a Metodologia Psedo-Espectral de Fourier**. 2007. Diss. (Mestrado) – Universidade Federal de Uberlândia. Citado nas p. 4, 17, 19.
- 12 VILLELA, Mariana Fernandes dos Santos. **Modelagem Matemática de Escoamentos Bifásicos Usando o Método Espectral de Fourier**. Mar. 2011. Diss. (Mestrado) – Universidade Federal de Uberlandia (UFU). Citado nas p. 4, 18.
- 13 KINOSHITA, Denise; DA SILVEIRA NETO, Aristeu; MARIANO, Felipe Pamplona; DA SILVA, Renato Aparecido Pimentel; SERFATY, Ricardo. A Novel Immersed Boundary/Fourier Pseudospectral Method for Flows with Thermal Effects. **Numerical Heat Transfer, Part B: Fundamentals**, Informa UK Limited, v. 69, n. 4, p. 312–333, mar. 2016. DOI: <https://doi.org/10.1080/10407790.2015.1104199>. Citado nas p. 4, 19.
- 14 NASCIMENTO, Andreia Aoyagui. **Métodos Pseudoespectral de Fourier e Fronteira Imersa Aplicados a Escoamentos Simplificados de Engenharia de Perfuração**. 2016. Tese (Doutorado) – Universidade Federal de Uberlândia. Citado nas p. 4, 19, 49, 60, 61.
- 15 CHEN, Lin; ASAI, Keisuke; NONOMURA, Taku; XI, Guannan; LIU, Tianshu. A Review of Backward-Facing Step (BFS) Flow Mechanisms, Heat Transfer and Control. **Thermal Science and Engineering Progress**, v. 6, p. 194–216, jun. 2018. ISSN 2451-9049. DOI: <https://doi.org/10.1016/j.tsep.2018.04.004>. Citado nas p. 5, 13.
- 16 JOVIC, Srba; DRIVER, David M. **Backward-Facing Step Measurements at Low Reynolds Number, $Re (Sub h) = 5000$** . CA, USA, 1994. Citado nas p. 5, 11, 12, 103–105, 107, 108, 110, 115, 116, 126.
- 17 LE, Hung; MOIN, Parviz; KIM, John. Direct Numerical Simulation of Turbulent Flow over a Backward-Facing Step. **Journal of Fluid Mechanics**, v. 330, p. 349–374, jan. 1997. ISSN 0022-1120, 1469-7645. DOI: <https://doi.org/10.1017/S0022112096003941>. Citado nas p. 5, 12, 104, 108–111, 114–116, 126, 127.
- 18 LEE, T; MATEESCU, D. Experimental and Numerical Investigation of 2-D Backward-Facing Step Flow. **Journal of fluids and structures**, Elsevier, v. 12, n. 6, p. 703–716, 1998. Citado nas p. 5, 12, 83, 84, 101.

- 19 AIDER, Jean-Luc; DANET, Alexandra. Large-Eddy Simulation Study of Upstream Boundary Conditions Influence upon a Backward-Facing Step Flow. **Comptes Rendus Mécanique**, v. 334, n. 7, p. 447–453, jul. 2006. ISSN 1631-0721. DOI: <https://doi.org/10.1016/j.crme.2006.05.004>. Citado nas p. 5, 12, 13.
- 20 ROUIZI, Y.; FAVENNEC, Y.; VENTURA, J.; PETIT, D. Numerical Model Reduction of 2D Steady Incompressible Laminar Flows: Application on the Flow over a Backward-Facing Step. **Journal of Computational Physics**, v. 228, n. 6, p. 2239–2255, abr. 2009. ISSN 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2008.12.001>. Citado na p. 5.
- 21 JÜRGENS, W.; KALTENBACH, H. -J. The Effect of Sweep on the Forced Transitional Flow over a Backward-Facing Step. **Computers & Fluids**, v. 59, p. 1–10, abr. 2012. ISSN 0045-7930. DOI: <https://doi.org/10.1016/j.compfluid.2012.01.022>. Citado nas p. 5, 13.
- 22 KANCHI, H.; SENGUPTA, K.; MASHAYEK, F. Effect of Turbulent Inflow Boundary Condition in LES of Flow over a Backward-Facing Step Using Spectral Element Method. **International Journal of Heat and Mass Transfer**, v. 62, p. 782–793, jul. 2013. ISSN 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2013.03.038>. Citado nas p. 5, 13.
- 23 MOREIRA, Leonardo de Queiroz. **Modelagem Matemática de Jatos Em Desenvolvimento Espacial Usando Metodologia Pseudoespectral de Fourier**. 2011. Tese (Doutorado) – Universidade Federal de Uberlândia. Citado nas p. 5, 13, 18, 19, 44, 45, 49, 60, 61, 70, 83, 107.
- 24 LÉONARD, A. Energy Cascade in Large-Eddy Simulations of Turbulent Fluid Flows. **Advances in Geophysics**, 1974. DOI: [https://doi.org/10.1016/S0065-2687\(08\)60464-1](https://doi.org/10.1016/S0065-2687(08)60464-1). Citado na p. 8.
- 25 GERMANO, M. A Proposal for a Redefinition of the Turbulent Stresses in the Filtered Navier–Stokes Equations. **The Physics of Fluids**, v. 29, n. 7, p. 2323–2324, jul. 1986. ISSN 0031-9171. DOI: <https://doi.org/10.1063/1.865568>. Citado nas p. 8, 30, 137.
- 26 SILVEIRA NETO, Aristeu. **Escoamentos Turbulentos. Análise Física e Modelagem Teórica**. Uberlândia: Composer, 2020. ISBN 978-65-990365-1-4. Citado nas p. 8, 26, 29–34, 53.
- 27 SMAGORINSKY, J. GENERAL CIRCULATION EXPERIMENTS WITH THE PRIMITIVE EQUATIONS: I. THE BASIC EXPERIMENT*. **Monthly Weather Review**, v. 91, n. 3, p. 99–164, mar. 1963. ISSN 0027-0644, 1520-0493. DOI: [https://doi.org/10.1175/1520-0493\(1963\)091<0099:GCEWTP>2.3.CO;2](https://doi.org/10.1175/1520-0493(1963)091<0099:GCEWTP>2.3.CO;2). Citado nas p. 8, 33.
- 28 SAGAUT, Pierre. **Large Eddy Simulation for Incompressible Flows**. 3. ed. Berlin/Heidelberg: Springer-Verlag, 2006. (Scientific Computation). ISBN

- 978-3-540-26344-9. DOI: <https://doi.org/10.1007/b137536>. Citado nas p. 9, 28–31, 33, 34, 37–40, 105, 119, 125, 131.
- 29 VAN DRIEST, E. R. On Turbulent Flow Near a Wall. **Journal of the Aeronautical Sciences**, v. 23, n. 11, p. 1007–1011, nov. 1956. ISSN 1936-9956. DOI: <https://doi.org/10.2514/8.3713>. Citado nas p. 9, 11, 34.
- 30 NICOUD, F.; DUCROS, F. Subgrid-Scale Stress Modelling Based on the Square of the Velocity Gradient Tensor. **Flow, Turbulence and Combustion**, v. 62, n. 3, p. 183–200, set. 1999. ISSN 1386-6184, 1573-1987. DOI: <https://doi.org/10.1023/A:1009995426001>. Citado nas p. 9, 11, 35, 36, 120.
- 31 MENEVEAU, Charles; LUND, Thomas; CABOT, W. A Lagrangian Dynamic Subgrid-Scale Model of Turbulence. **Journal of Fluid Mechanics**, 1996. DOI: <https://doi.org/10.1017/S0022112096007379>. Citado na p. 9.
- 32 VREMAN, A.W. An Eddy-Viscosity Subgrid-Scale Model for Turbulent Shear Flow: Algebraic Theory and Applications, 2004. DOI: <https://doi.org/10.1063/1.1785131>. Citado na p. 9.
- 33 NICOUD, Franck; TODA, Hubert Baya; CABRIT, Olivier; BOSE, Sanjeeb; LEE, Jungil. Using Singular Values to Build a Subgrid-Scale Model for Large Eddy Simulations, 2011. DOI: <https://doi.org/10.1063/1.3623274>. Citado na p. 9.
- 34 LIU, Shewen; MENEVEAU, Charles; KATZ, Joseph. On the Properties of Similarity Subgrid-Scale Models as Deduced from Measurements in a Turbulent Jet. **Journal of Fluid Mechanics**, 1994. DOI: <https://doi.org/10.1017/S0022112094002296>. Citado na p. 10.
- 35 SILVIS, Maurits H.; REMMERSWAAL, Ronald A.; VERSTAPPEN, Roel. Physical Consistency of Subgrid-Scale Models for Large-Eddy Simulation of Incompressible Turbulent Flows. **Physics of Fluids**, 2017. DOI: <https://doi.org/10.1063/1.4974093>. Citado na p. 11.
- 36 VERSTAPPEN, Roel. When Does Eddy Viscosity Damp Subfilter Scales Sufficiently? **J. Sci. Comput.**, 2011. DOI: <https://doi.org/10.1007/S10915-011-9504-4>. Citado na p. 11.
- 37 ROZEMA, Wybe; BAE, H. Jane; MOIN, Parviz; VERSTAPPEN, Roel. Minimum-Dissipation Models for Large-Eddy Simulation, 2015. DOI: <https://doi.org/10.1063/1.4928700>. Citado na p. 11.
- 38 SMIRNOV, A.; SHI, S.; CELIK, I. Random Flow Generation Technique for Large Eddy Simulations and Particle-Dynamics Modeling. **Journal of Fluids Engineering**, v. 123, n. 2, p. 359–371, jun. 2001. ISSN 0098-2202, 1528-901X. DOI: <https://doi.org/10.1115/1.1369598>. Citado nas p. 13, 107.

- 39 PESKIN, Charles S. The Immersed Boundary Method. **Acta Numerica**, v. 11, p. 479–517, 2002. DOI: <https://doi.org/10.1017/S0962492902000077>. Citado nas p. 14, 46, 47.
- 40 TORNBERG, Anna-Karin; ENGQUIST, Björn. Numerical Approximations of Singular Source Terms in Differential Equations. **Journal of Computational Physics**, v. 200, n. 2, p. 462–488, 2004. ISSN 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2004.04.011>. Citado na p. 14.
- 41 GRIFFITH, Boyce E.; PESKIN, Charles S. On the Order of Accuracy of the Immersed Boundary Method: Higher Order Convergence Rates for Sufficiently Smooth Problems. **Journal of Computational Physics**, v. 208, n. 1, p. 75–105, 2005. ISSN 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2005.02.011>. Citado na p. 14.
- 42 MARIANO, Felipe Pamplona. **Solução Numérica Das Equações de Navier-Stokes Usando Uma Hibridação Das Metodologias Fronteira Imersa e Pseudo-Espectral de Fourier**. 2011. Tese (Doutorado) – Universidade Federal de Uberlândia. Citado nas p. 14, 17, 18, 44, 47, 49, 60, 61, 77.
- 43 UHLMANN, Markus. An Immersed Boundary Method with Direct Forcing for the Simulation of Particulate Flows. **Journal of Computational Physics**, Elsevier BV, v. 209, n. 2, p. 448–476, nov. 2005. DOI: <https://doi.org/10.1016/j.jcp.2005.03.017>. Citado nas p. 14, 42, 49.
- 44 WANG, Zeli; FAN, Jianren; LUO, Kun. Combined Multi-Direct Forcing and Immersed Boundary Method for Simulating Flows with Moving Particles. **International Journal of Multiphase Flow**, Elsevier BV, v. 34, n. 3, p. 283–302, mar. 2008. Citado nas p. 14, 44, 49.
- 45 MITTAL, R.; DONG, H.; BOZKURTTAS, M.; NAJJAR, F.M.; VARGAS, A.; VON LOEBBECKE, A. A Versatile Sharp Interface Immersed Boundary Method for Incompressible Flows with Complex Boundaries. **Journal of Computational Physics**, v. 227, n. 10, p. 4825–4852, 2008. ISSN 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2008.01.028>. Citado na p. 15.
- 46 RIBEIRO NETO, Hélio. **Modelagem matemática e computacional de interações fluido-sólido utilizando o método de fronteira imersa tipo ghost em ambiente paralelo e com malha adaptativa**. Jul. 2021. f. 128. Tese (Doutorado em Engenharia Mecânica) – Universidade Federal de Uberlândia, Uberlândia, Minas Gerais, Brasil. DOI: <https://doi.org/10.14393/ufu.te.2021.412>. Citado na p. 15.
- 47 CHANG, Po-Hua; LIAO, Chuan-Chieh; HSU, Hsin-Wei; LIU, Shih-Huang; LIN, Chao-An. Simulations of Laminar and Turbulent Flows over Periodic Hills with Immersed Boundary Method. **Computers Fluids**, v. 92, p. 233–243, 2014. ISSN 0045-7930. DOI: <https://doi.org/10.1016/j.compfluid.2013.10.043>. Citado na p. 15.

- 48 BAO, Yuanxun; KAYE, Jason; PESKIN, Charles S. A Gaussian-like Immersed-Boundary Kernel with Three Continuous Derivatives and Improved Translational Invariance. **Journal of Computational Physics**, v. 316, p. 139–144, 2016. ISSN 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2016.04.024>. Citado nas p. 16, 46, 47.
- 49 BAO, Yuanxun; DONEV, Aleksandar; GRIFFITH, Boyce E.; MCQUEEN, David M.; PESKIN, Charles S. An Immersed Boundary Method with Divergence-Free Velocity Interpolation and Force Spreading. **Journal of Computational Physics**, v. 347, p. 183–206, 2017. DOI: <https://doi.org/10.1016/j.jcp.2017.06.041>. Citado na p. 16.
- 50 CHEN, Zhe; PESKIN, Charles S. A Fourier Spectral Immersed Boundary Method with Exact Translation Invariance, Improved Boundary Resolution, and a Divergence-Free Velocity Field. **Journal of Computational Physics**, Elsevier, v. 509, p. 113048, 2024. ISSN 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2024.113048>. Citado nas p. 16, 48, 49.
- 51 GRUNINGER, Cole; GRIFFITH, Boyce E. **Composite B-Spline Regularized Delta Functions for the Immersed Boundary Method: Divergence-free Interpolation and Gradient-Preserving Force Spreading**. [S. l.: s. n.], 2025. arXiv: 2408.08280 [math.NA]. Citado nas p. 16, 17.
- 52 MARIANO, Felipe Pamplona; MOREIRA, Leonardo de Queiroz; NASCIMENTO, Andreia Aoyagui; SILVEIRA, NETO, Aristeu. An Improved Immersed Boundary Method by Coupling of the Multi-Direct Forcing and Fourier Pseudo-Spectral Methods. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, Springer Science and Business Media LLC, v. 44, n. 9, ago. 2022. Citado nas p. 18, 62, 105.
- 53 KINOSHITA, Denise. **Desenvolvimento e Implementação Da Metodologia Combinada Fronteira Imersa Térmica e Pseudoespectral de Fourier**. 2015. Tese (Doutorado) – Universidade Federal de Uberlândia (UFU). DOI: <https://doi.org/10.14393/ufu.te.2015.58>. Citado nas p. 18, 60, 61, 139.
- 54 JACOB, Bruno Tadeu Pereira. **Modelagem Matemática de Esteiras Em Desenvolvimento Temporal Utilizando o Método Pseudoespectral de Fourier**. Ago. 2015. Diss. (Mestrado) – Universidade Federal de Uberlândia (UFU). Citado na p. 19.
- 55 VILLELA, Mariana. **Modelagem Matemática de Escoamentos Bifásicos Usando a Metodologia IMERSPEC Combinada Com Os Métodos VOF e Front-Tracking**. Ago. 2015. Tese (Doutorado) – Universidade Federal de Uberlândia (UFU). Citado nas p. 19, 61.
- 56 VILLELA, M.F.S.; VILLAR, M.M.; SERFATY, R.; MARIANO, F.P.; SILVEIRA-NETO, A. Mathematical Modeling and Numerical Simulation of Two-Phase Flows Using Fourier Pseudospectral and Front-Tracking Methods: The Proposition of a New Method. **Applied Mathematical Modelling**, v. 52, p. 241–254, 2017. ISSN 0307-904X. DOI: <https://doi.org/10.1016/j.apm.2017.06.041>. Citado na p. 19.

- 57 NASCIMENTO, Andreia Aoyagui; MARIANO, Felipe Pamplona; DA SILVEIRA NETO, Aristeu; PADILLA, Elie Luis Martínez. Coupling of the Immersed Boundary and Fourier Pseudo-Spectral Methods Applied to Solve Fluid–Structure Interaction Problems. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, Springer, v. 46, n. 4, p. 213, 2024. Citado na p. 19.
- 58 MONTEIRO, Lucas Marques; MARIANO, Felipe Pamplona. Flow Modeling over Airfoils and Vertical Axis Wind Turbines Using Fourier Pseudo-Spectral Method and Coupled Immersed Boundary Method. **Axioms**, MDPI AG, v. 12, n. 2, p. 212, fev. 2023. Citado nas p. 19, 105.
- 59 SILVA, Mylena Carvalho; NASCIMENTO, Andreia Aoyagui. Analysis of the Fluid-Structure Interaction of a 2D Model of the Savonius Rotor, Using Immersed Boundary and Fourier Pseudospectral Methods. **Ibero-Latin American Congress on Computational Methods in Engineering (CILAMCE)**, dez. 2024. ISSN 2675-6269, 2675-6269. DOI: <https://doi.org/10.55592/cilamce.v6i06.10179>. Citado na p. 19.
- 60 ALBUQUERQUE, Laura Augusta Vasconcelos de; VILLELA, Mariana Fernandes dos Santos; MARIANO, Felipe Pamplona. Numerical Simulation of Flows Using the Fourier Pseudospectral Method and the Immersed Boundary Method. **Axioms**, v. 13, n. 228, 2024. ISSN 2075-1680. DOI: <https://doi.org/10.3390/axioms13040228>. Citado nas p. 19, 105.
- 61 ALBUQUERQUE, Laura Augusta Vasconcelos de; VILLELA, Mariana Fernandes dos Santos; MARIANO, Felipe Pamplona. Numerical Evaluation of the IMERSPEC Methodology and Spalart–Allmaras Turbulence Model in Fully Developed Channel Flow Simulations. **Fluids**, v. 10, n. 45, 2025. ISSN 2311-5521. DOI: <https://doi.org/10.3390/fluids10020045>. Citado nas p. 19, 105.
- 62 DE FREITAS, Thiago Fernando Santiago; DA SILVEIRA NETO, Aristeu; MARIANO, Felipe Pamplona. Accuracy Assessment of the 2D Laminar Boundary Layer on a Flat Plate in an Immersed Boundary-Fourier Pseudospectral Simulations. **Alexandria Engineering Journal**, v. 122, p. 279–287, 2025. ISSN 1110-0168. DOI: <https://doi.org/10.1016/j.aej.2025.02.110>. Citado nas p. 19, 61, 80, 110.
- 63 BOYD, John P. **Chebyshev and Fourier Spectral Methods**. 2. ed. New York, USA: Courier Corporation, 2001. Citado nas p. 20, 30, 62–64, 105.
- 64 BLAISDELL, GA; SPYROPOULOS, ET; QIN, JH. The Effect of the Formulation of Nonlinear Terms on Aliasing Errors in Spectral Methods. **Applied Numerical Mathematics**, Elsevier, v. 21, n. 3, p. 207–219, 1996. Citado nas p. 20, 65, 66.
- 65 HOU, Thomas Y.; LI, Ruo. Computing Nearly Singular Solutions Using Pseudo-Spectral Methods. **Journal of Computational Physics**, v. 226, n. 1, p. 379–397, set. 2007. ISSN 00219991. DOI: <https://doi.org/10.1016/j.jcp.2007.04.014>. Citado nas p. 20, 64, 65.

- 66 BECK, Andrea D; BOLEMAN, Thomas; FLAD, David; FRANK, Hannes; GASSNER, Gregor J; HINDENLANG, Florian; MUNZ, Claus-Dieter. High-Order Discontinuous Galerkin Spectral Element Methods for Transitional and Turbulent Flow Simulations. **International Journal for Numerical Methods in Fluids**, Wiley Online Library, v. 76, n. 8, p. 522–548, 2014. Citado na p. 21.
- 67 LODATO, Guido; CASTONGUAY, Patrice; JAMESON, Antony. Structural Wall-Modeled LES Using a High-Order Spectral Difference Scheme for Unstructured Meshes. **Flow, turbulence and combustion**, Springer, v. 92, p. 579–606, 2014. Citado na p. 21.
- 68 KRAIS, Nico; BECK, Andrea; BOLEMAN, Thomas; FRANK, Hannes; FLAD, David; GASSNER, Gregor; HINDENLANG, Florian; HOFFMANN, Malte; KUHN, Thomas; SONNTAG, Matthias *et al.* FLEXI: A High Order Discontinuous Galerkin Framework for Hyperbolic–Parabolic Conservation Laws. **Computers & Mathematics with Applications**, Elsevier, v. 81, p. 186–219, 2021. Citado na p. 21.
- 69 STONE, Christopher; DUQUE, Earl; ZHANG, Yao; CAR, David; DAVIS, Roger; OWENS, John. GPGPU Parallel Algorithms for Structured-Grid CFD Codes. *In*: 20TH AIAA Computational Fluid Dynamics Conference. Honolulu, Hawaii: [s. n.], jun. 2011. DOI: <https://doi.org/10.2514/6.2011-3221>. Citado nas p. 22, 24.
- 70 XIANG, Yue; YU, Bo; YUAN, Qing; SUN, Dongliang. GPU Acceleration of CFD Algorithm: HSMAC and SIMPLE. **Procedia Computer Science**, Elsevier, v. 108, p. 1982–1989, 2017. Citado nas p. 22, 24, 57.
- 71 OYARZUN, Guillermo; BORRELL, Ricard; GOROBETS, Andrey; OLIVA, Assensi. Portable Implementation Model for CFD Simulations. Application to Hybrid CPU/GPU Supercomputers. **International Journal of Computational Fluid Dynamics**, IAHR Website, v. 31, n. 9, p. 396–411, 2017. DOI: <https://doi.org/10.1080/10618562.2017.1390084>. eprint: <https://doi.org/10.1080/10618562.2017.1390084>. Citado nas p. 22, 24.
- 72 KORNEEV, Boris; LEVCHENKO, Vadim. Runge-Kutta Discontinuous Galerkin Method and DiamondTorre GPGPU Algorithm for Effective Simulation of Large 3D Multiphase Fluid Flows with Shocks. *In*: 2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON). [S. l.: s. n.], 2019. p. 0817–0822. DOI: <https://doi.org/10.1109/SIBIRCON48586.2019.8958102>. Citado nas p. 23, 24.
- 73 LEI, Jiang; LI, Da-li; ZHOU, Yun-long; LIU, Wei. Optimization and Acceleration of Flow Simulations for CFD on CPU/GPU Architecture. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, v. 41, jul. 2019. DOI: <https://doi.org/10.1007/s40430-019-1793-9>. Citado nas p. 23, 24.
- 74 LENZ, Stephan; SCHÖNHERR, Martin; GEIER, Martin; KRAFCZYK, Manfred; PASQUALI, Andrea; CHRISTEN, Andreas; GIOMETTO, Marco. Towards Real-Time Simulation of Turbulent Air Flow over a Resolved Urban Canopy Using the Cumulant

- Lattice Boltzmann Method on a GPGPU. **Journal of Wind Engineering and Industrial Aerodynamics**, v. 189, p. 151–162, 2019. ISSN 0167-6105. DOI: <https://doi.org/10.1016/j.jweia.2019.03.012>. Citado nas p. 23, 24.
- 75 XUE, Weicheng; JACKSON, Charles W.; ROY, Christopher J. An Improved Framework of GPU Computing for CFD Applications on Structured Grids Using OpenACC. **CoRR**, abs/2012.02925, 2021. arXiv: 2012.02925. Citado nas p. 23, 24.
- 76 PISCAGLIA, Federico; GHIOLDI, Federico. GPU Acceleration of CFD Simulations in OpenFOAM. **Aerospace**, v. 10, n. 792, 2023. ISSN 2226-4310. DOI: <https://doi.org/10.3390/aerospace10090792>. Citado nas p. 23, 24.
- 77 THE OPENFOAM FOUNDATION. **OpenFOAM: The Open Source CFD Toolbox**. London, UK: [s. n.], 2024. OpenFOAM Foundation. Citado na p. 23.
- 78 DA SILVA REIS, César Augusto Borges; BOTEZELLI, Daniel; DE AZEVEDO, Arthur Mendonça; DOS SANTOS MAGALHÃES, Elisan; DA SILVEIRA NETO, Aristeu. Accelerating Conjugate Heat Transfer Simulations in Squared Heated Cavities through Graphics Processing Unit (GPU) Computing. **Computation**, MDPI, v. 12, n. 5, p. 106, 2024. Citado na p. 24.
- 79 SOUZA, Francisco; ABID, Hussain Ali; MARKESTEIJN, Annabel; LAWRENCE, Jack; KARABASOV, Sergey A. Gpu-Accelerated Large-Eddy Simulations. In: ANAIS Do 32º Simpósio Da Pós-graduação Em Engenharia Mecânica (POSMEC). [S. l.: s. n.], fev. 2025. DOI: <https://doi.org/10.29327/5441125.32-60>. Citado na p. 24.
- 80 BRIGGS, William L; HENSON, Van Emden. **The DFT: An Owner's Manual for the Discrete Fourier Transform**. Philadelphia: Society for Industrial and Applied Mathematics, 1995. Citado nas p. 29, 30, 50, 58, 63.
- 81 POPE, Stephen B. **Turbulent Flows**. 1. ed. New York, USA: Cambridge University Press, ago. 2000. ISBN 978-0-521-59125-6 978-0-521-59886-6 978-0-511-84053-1. DOI: <https://doi.org/10.1017/CBO9780511840531>. Citado nas p. 31, 34, 118.
- 82 CANUTO, Claudio; HUSSAINI, M. Yousuff; QUARTERONI, Alfio; ZANG, Thomas A. **Spectral Methods Evolution to Complex Geometries and Applications to Fluid Dynamics. Evolution to Complex Geometries and Applications to Fluid Dynamics**. Berlin, Germany: Springer, 2007. p. 596. ISBN 978-3-642-43395-5. Citado na p. 32.
- 83 ALLAMPALLI, Vasanth; HIXON, Ray; NALLASAMY, M.; SAWYER, Scott D. High-Accuracy Large-Step Explicit Runge–Kutta (HALE-RK) Schemes for Computational Aeroacoustics. **Journal of Computational Physics**, Elsevier BV, v. 228, n. 10, p. 3837–3850, jun. 2009. Citado nas p. 44, 66, 67.
- 84 SHIH, Yu-hsuan; WRIGHT, Garrett; ANDÉN, Joakim; BLASCHKE, Johannes; BARNETT, Alex H. **cuFINUFFT: A Load-Balanced GPU Library for**

- General-Purpose Nonuniform FFTs.** [S. l.: s. n.], 2021. arXiv: 2102.08463 [cs.DC]. Citado nas p. 49, 58.
- 85 NASCIMENTO, AA; MARIANO, FP; SILVERIA-NETO, A; PADILLA, ELM. A Comparison of Fourier Pseudospectral Method and Finite Volume Method Used to Solve the Burgers Equation. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, Springer, v. 36, n. 4, p. 737–742, 2014. DOI: <https://doi.org/10.1007/s40430-013-0124-9>. Citado na p. 55.
- 86 SKJELLUM, Anthony. **Using MPI: Portable Parallel Programming with the Message-Passing Interface**. 3rd ed. Cambridge: MIT Press, 2015. (Scientific and Engineering Computation Ser). ISBN 978-0-262-52739-2 978-0-262-32660-5. Citado nas p. 55–57.
- 87 NVIDIA CORPORATION. **CUDA C++ Programming Guide, Release 12.8**. Manual. Santa Clara, CA, USA, abr. 2024. Citado na p. 57.
- 88 CHAPRA, Steven; CANALE, Raymond. **Numerical Methods for Engineers**. 7. ed. New York: McGraw Hill, 2014. p. 992. ISBN 0-07-339792-X. Citado na p. 58.
- 89 TAKAHASHI, Daisuke. **Fast Fourier Transform Algorithms for Parallel Computers**. Singapore: Springer Singapore Pte. Limited, 2019. (High-Performance Computing Ser, v.2). ISBN 978-981-13-9965-7. Citado na p. 59.
- 90 TAKAHASHI, Daisuke; FRANCHETTI, Franz. FFTE on SVE: SPIRAL-generated Kernels. In: PROCEEDINGS of the International Conference on High Performance Computing in Asia-Pacific Region. New York, NY, USA: Association for Computing Machinery, 2020. (HPCAsia '20), p. 114–122. ISBN 978-1-4503-7236-7. DOI: <https://doi.org/10.1145/3368474.3368488>. Citado nas p. 59, 72.
- 91 BARNETT, Alexander H; MAGLAND, Jeremy; AF KLINTEBERG, Ludvig. A Parallel Nonuniform Fast Fourier Transform Library Based on an “exponential of Semicircle” Kernel. **SIAM Journal on Scientific Computing**, SIAM, v. 41, n. 5, p. c479–c504, 2019. Citado na p. 59.
- 92 AMERICAN SOCIETY OF MECHANICAL ENGINEERS (ASME). **Verification, Validation, and Uncertainty Quantification Terminology in Computational Modeling and Simulation**. 1. ed. New York, NY: ASME, 2022. ISBN 9780791875506. Citado nas p. 69, 70.
- 93 BLAIS, Bruno; BERTRAND, Francois. On the Use of the Method of Manufactured Solutions for the Verification of CFD Codes for the Volume-Averaged Navier–Stokes Equations. **Computers & Fluids**, Elsevier, v. 114, p. 121–129, 2015. Citado na p. 69.

- 94 OBERKAMPF, William L; ROY, Christopher J. **Verification and Validation in Scientific Computing**. New York, NY: Cambridge university press, 2010. ISBN 978-0-511-90800-2. Citado na p. 70.
- 95 HENSHAW, William D. A Fourth-Order Accurate Method for the Incompressible Navier-Stokes Equations on Overlapping Grids. **Journal of computational physics**, Elsevier, v. 113, n. 1, p. 13–25, 1994. Citado na p. 70.
- 96 UNIVERSIDADE FEDERAL DE GOIÁS (UFG). **LAMCAD – Laboratório de Modelagem e Computação Científica Avançada**. Goiânia, Brasil: UFG, 2024. Citado nas p. 72, 88.
- 97 MULLER, Jean-Michel; BRUNIE, Nicolas; DE DINECHIN, Florent; JEANNEROD, Claude-Pierre; JOLDES, Mioara; LEFÈVRE, Vincent; MELQUIOND, Guillaume; REVOL, Nathalie; TORRES, Serge. **Handbook of Floating-Point Arithmetic**. 2. ed. Cham, Switzerland: Springer, 2018. v. 1. DOI: <https://doi.org/10.1007/978-3-319-76526-6>. Citado na p. 74.
- 98 DE FREITAS, Thiago. **Modelagem Matemática e Computacional de Camada Limite Laminar Utilizando a Metodologia Mista Pseudoespectral de Fourier e Fronteira Imersa**. Jul. 2023. Dissertação, Mestrado – Universidade Federal de Uberlândia (UFU), Uberlândia. DOI: <https://doi.org/10.14393/ufu.di.2023.286>. Citado na p. 80.
- 99 LATINA, KaBuM! | Maior E-commerce de Tecnologia e Games da América. **Gpu Nvidia Geforce Gtx 1060 6gb GDDR5 192 Bits Vinik Gpuvk10606gb2f**. [S. l.: s. n.]. <https://www.kabum.com.br/produto/1030540/false>. Citado na p. 94.
- 100 NVIDIA A100 SXM4 80GB GDDR6 GPU COM HEATSINK - R\$ 150.200. [S. l.: s. n.]. <https://www.mercadolivre.com.br/nvidia-a100-sxm4-80gb-gddr6-gpu-com-heatsink/up/MLBU3131927354>. Citado na p. 94.
- 101 SCHLATTER, P.; ADAMS, N.A.; KLEISER, L. A Windowing Method for Periodic Inflow/Outflow Boundary Treatment of Non-Periodic Flows. **Journal of Computational Physics**, v. 206, n. 2, p. 505–535, jul. 2005. ISSN 00219991. DOI: <https://doi.org/10.1016/j.jcp.2004.12.015>. Citado na p. 106.
- 102 DE FREITAS, Thiago Fernando Santiago. **Vídeos: Desenvolvimento temporal do escoamento turbulento sobre degrau com modelos submalha Smagorinsky-Van Driest, WALE e Dinâmico de Germano-Lilly com Re=5000**. [S. l.]: Zenodo, ago. 2026. DOI: <https://doi.org/10.5281/ZENODO.21829610>. Citado nas p. 112, 119, 125.