

UNIVERSIDADE FEDERAL DO PARANÁ

BRUNO BENATO RUTYNA

**COMPARAÇÃO ENTRE TÉCNICAS DE EXTRAPOLAÇÃO ASSOCIADAS AO  
MÉTODO MULTIGRID APLICADAS NAS EQUAÇÕES DE NAVIER-STOKES**

CURITIBA

2018

BRUNO BENATO RUTYNA

**COMPARAÇÃO ENTRE TÉCNICAS DE EXTRAPOLAÇÃO ASSOCIADAS AO  
MÉTODO *MULTIGRID* APLICADAS NAS EQUAÇÕES DE NAVIER-STOKES**

Dissertação apresentada ao Curso de Pós-Graduação em  
Métodos Numéricos em Engenharia, Área de Concentração  
em Programação Matemática, do Setor de Ciências Exatas e de  
Tecnologia, Universidade Federal do Paraná, como requisito  
parcial para a obtenção do título de Mestre em Métodos  
Numéricos em Engenharia.

Orientador: Prof. Dr. Marcio Augusto Villela Pinto

CURITIBA  
2018

FICHA CATALOGRÁFICA ELABORADA PELO SISTEMA DE BIBLIOTECAS/UFPR  
BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

---

R982c      Rutyna, Bruno Benato  
                 Comparação entre técnicas de interpolação associadas ao método Multigrid aplicadas nas equações de Navier-Stokes / Bruno Benato Rutyna. – Curitiba, 2018.  
                 60 p. : il. color. ; 30 cm.

Dissertação - Universidade Federal do Paraná, Setor de Ciências Exatas e de Tecnologia, Programa de Pós-Graduação em Métodos Numéricos em Engenharia, 2018.

Orientador: Marcio Augusto Villela Pinto.

1. Aceleração de convergência. 2. Dinâmica dos fluidos computacional. 3. Método de volumes finitos. 4. Método Multigrid. 5. Equações de Navier-Stokes. I. Universidade Federal do Paraná. II. Pinto, Marcio Augusto Villela. III. Título.

CDD: 515.353

---

Bibliotecária: Romilda Santos - CRB-9/1214

## TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em MÉTODOS NUMÉRICOS EM ENGENHARIA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **BRUNO BENATO RUTYNA** intitulada: **COMPARAÇÃO ENTRE TÉCNICAS DE EXTRAPOLAÇÃO ASSOCIADAS AO MÉTODO MULTIGRID APLICADAS NAS EQUAÇÕES DE NAVIER-STOKES**, após terem inquirido o aluno e realizado a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 23 de Fevereiro de 2018.

  
MARCIO AUGUSTO VILLELA PINTO

Presidente da Banca Examinadora (UFPR)

  
ROBERTO DALLEDONE MACHADO

Avaliador Interno (UFPR)

  
LUCIANO KIYOSHI ARAKI

Avaliador Interno (UFPR)

  
SIMONE DE FATIMA TOMAZZONI GONÇALVES

Avaliador Externo (UFPR)

Dedico este trabalho a minha família e amigos.

## AGRADECIMENTOS

Agradeço o apoio financeiro da Capes (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pela bolsa de mestrado concedida, à banca avaliadora, constituída pelos professores: Simone de Fátima Tomazzoni Gonçalves, Roberto Dalledone Machado, Luciano Kiyoshi Araki, cujos apontamentos enriqueceram este trabalho, aos meus colegas do grupo de pesquisa *Método Multigrid* e ao meu orientador, professor Marcio Augusto Villela Pinto, que acompanharam com entusiasmo a realização desta dissertação.

## RESUMO

Os problemas de Dinâmica dos Fluidos Computacional aumentam constantemente em quantidade e complexidade, dada a importância de suas aplicações, tornando-se um desafio que exige aprimoramentos constantes nos métodos que buscam soluções. Neste sentido, esta dissertação aplica técnicas de extrapolação em associação ao método *multigrid* para obtenção de soluções, com o propósito de reduzir o erro de iteração, o tempo de processamento (tempo de CPU) e melhorar os fatores de convergência. Essa aplicação é feita em um caso de escoamento laminar bidimensional de um fluido incompressível em regime transiente ditado pelas equações de Navier-Stokes, resolvidas iterativamente via método de projeção e o Método de Volumes Finitos. Os métodos de extrapolação utilizados são: Aitken, Empírico, Mitin, Épsilon escalar, Rho escalar, Épsilon topológico e Rho topológico. Para descobrir qual é o mais apropriado para o problema proposto, foram realizadas duas etapas. Primeiramente, os extrapoladores foram aplicados individualmente após a aplicação do método *multigrid*, para averiguar qual resulta em melhores ganhos em alguns parâmetros, como por exemplo, na redução do erro de iteração. Os extrapoladores que obtiveram melhores resultados foram utilizados na segunda etapa, na qual foram aplicados entre os ciclos do método *multigrid*. Neste sentido, dentre as técnicas de extrapolação aplicadas, as que apresentaram melhores propriedades de convergência na primeira etapa frente ao *multigrid* sem extrapolação foram as de Épsilon topológico e Épsilon escalar. Já na segunda etapa, ambas as técnicas mantiveram seu êxito em relação ao *multigrid* sem extrapolação, entretanto, o extrapolador Épsilon topológico apresentou taxas de convergência mais significativas em relação ao Épsilon escalar. Os outros parâmetros analisados foram: pico de memória de armazenamento, norma adimensionalizada do resíduo com base na estimativa inicial e normas do erro de iteração. Assim é possível afirmar qual técnica de extrapolação é melhor e compará-la com a aplicação do método *multigrid* sem o uso de extrapoladores.

Palavras-chave: aceleração de convergência, Dinâmica dos Fluidos Computacional, Método de Volumes Finitos, método de projeção, Épsilon escalar, Épsilon topológico.

## ABSTRACT

Computational Fluid Dynamics problems are constantly increasing in quantity and complexity, given the importance of its applications, and this challenge always requires an improvement in methods for finding solutions. In this sense, this dissertation applies solutions of extrapolation techniques in association with the multigrid method, with the purpose of reducing the iteration error, the processing time (CPU time) and to improve the convergence factors. This application is made in a case of two-dimensional laminar flow of a time-dependent incompressible fluid dictated by the Navier-Stokes equations, solved iteratively by the projection method and the Finite Volume Method. The extrapolation methods used are: Aitken, Empiric, Mitin, Scalar epsilon, Scalar rho, Topological epsilon and Topological rho. To find out which one is most appropriate for the proposed problem, they were tested in two steps. First, they are applied individually after the use of the multigrid method, to find out which one presents greater efficiency in some parameters, for example, in the iterative error reduction. The best-performing extrapolation techniques will be used in the second step, where they will be applied between the cycles of the multigrid method. In this sense, among the applied extrapolation techniques, the ones that presented better convergence properties in the first stage against the multigrid without extrapolation were Topological epsilon and Scalar epsilon. In the second stage, both techniques maintained their success in relation to multigrid without extrapolation, however, the topological Epsilon extrapolator presented more significant convergence rates in relation to the scalar Epsilon. The other parameters analyzed were: storage memory peak, dimensionless norm of the residual based on the initial estimate and the error norms of iteration. Thus, it is possible to state which extrapolation technique is best and compare it with the multigrid method without extrapolation methods.

Key-words: convergence acceleration, Computational Fluid Dynamics, Finite Volume Method, projection method, scalar Epsilon, topological Epsilon.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Malha estruturada com ordenação do tipo lexicográfica . . . . .	17
Figura 2 – Exemplo de relação entre malhas . . . . .	18
Figura 3 – Ordenação <i>red-black</i> para o caso bidimensional . . . . .	26
Figura 4 – Modos de Fourier com $k = 1, 3, 6$ . . . . .	29
Figura 5 – Jacobi ponderado com $\omega = 2/3$ aplicado ao problema de condução de calor unidimensional, com $N=64$ , estimativa inicial igual a $(\mathbf{v}_1 + \mathbf{v}_6 + \mathbf{v}_{32})/3$ , 100 iterações e o erro mensurado em valor absoluto . . . . .	30
Figura 6 – Processo de geração de malhas . . . . .	31
Figura 7 – Volume de controle em uma malha desencontrada, no qual os símbolos $\bullet$ , $\circ$ e $\square$ representam variáveis distintas . . . . .	31
Figura 8 – Posições das variáveis $\bullet$ , $\circ$ e $\square$ na ordem lexicográfica . . . . .	32
Figura 9 – Diagrama de um ciclo V aplicado em 5 malhas. S = suavização, R = restrição, P = prolongação . . . . .	34
Figura 10 – Malha do tipo desencontrada . . . . .	42
Figura 11 – <i>Volume de controle centrado em P</i> . . . . .	42
Figura 12 – Fator de convergência média empírico para cada instante de tempo . . . . .	48
Figura 13 – $\ \mathbf{r}\ _2$ para cada instante de tempo . . . . .	48
Figura 14 – $\hat{q}^{(k)}$ e $\ \mathbf{r}\ _2$ a cada passo de tempo . . . . .	51
Figura 15 – Norma $L_2$ do resíduo da variável $u$ versus número de iterações no passo de tempo $t = 10$ . . . . .	52
Figura 16 – O valor de <i>speed-up</i> entre o <i>multigrid</i> puro e o <i>multigrid</i> com o extrapolador $\tilde{\epsilon}$ . . . . .	52
Figura 17 – Uso de memória RAM do <i>multigrid</i> com extrapolador em comparação ao <i>multigrid</i> puro . . . . .	53

## LISTA DE TABELAS

Tabela 2 – TIPOS DE ABORDAGENS EM ENGENHARIA . . . . .	16
Tabela 3 – Número de soluções utilizadas para extrapolações . . . . .	46
Tabela 4 – Classificação das aplicações pós- <i>multigrid</i> . . . . .	47
Tabela 5 – Resultados da aplicação A1 . . . . .	47
Tabela 6 – Resultados da aplicação A2 . . . . .	47
Tabela 7 – Resultados da aplicação B1 . . . . .	48
Tabela 8 – Resultados da aplicação B2 . . . . .	49
Tabela 9 – Resultados da aplicação C1 . . . . .	49
Tabela 10 – Classificação das aplicações entre ciclos <i>multigrid</i> . . . . .	50
Tabela 11 – Resultados da aplicação D1 . . . . .	50
Tabela 12 – Resultados da aplicação E1 . . . . .	50
Tabela 13 – Resultados da aplicação E2 . . . . .	50

## **LISTA DE SIGLAS**

- |       |                                                                            |
|-------|----------------------------------------------------------------------------|
| CFD   | - <i>Computational Fluid Dynamics</i> - Dinâmica dos Fluidos Computacional |
| CPU   | - <i>Central Processing Unit</i> - Unidade Central de Processamento        |
| CS    | - <i>Correction Scheme</i> - Esquema de Correção                           |
| EDPs  | - Equações Diferenciais Parciais                                           |
| GS-RB | - Gauss-Seidel <i>Red-Black</i>                                            |
| MDF   | - Método de Diferenças Finitas                                             |
| MEF   | - Método de Elementos Finitos                                              |
| MG    | - <i>Multigrid</i>                                                         |
| MVF   | - Método de Volumes Finitos                                                |
| RAM   | - <i>Random Access Memory</i> - Memória de Acesso Aleatório                |
| SBDF  | - <i>Semi Backward Difference Formula</i>                                  |
| SOR   | - <i>Successive Over Relaxation</i> - Sobre-Relaxação Sucessiva            |

## LISTA DE SÍMBOLOS

$E$	- espaço vetorial
$E^*$	- espaço dual algébrico de $E$
$E_\pi$	- erro de arredondamento
$E_h$	- erro de truncamento
$E_n$	- erro de iteração
$E_{Pr}$	- erro de programação
$h_x$	- comprimento do volume na direção coordenada $x$
$h_y$	- comprimento do volume na direção coordenada $y$
$I_h^{2h}$	- operador de restrição
$I_{2h}^h$	- operador de prolongação
$ITE_{max}$	- número máximo a realizar os ciclos V
$k$	- número de ondas
$l$	- diâmetro do tubo
$L_1$	- norma-1
$L_2$	- norma-2
$L_x$	- comprimento do domínio na direção coordenada $x$
$L_y$	- comprimento do domínio na direção coordenada $y$
$L_\infty$	- norma infinito
$L_{max}$	- número de níveis de engrossamento
$N_x$	- número de volumes na direção coordenada $x$
$N_y$	- número de volumes na direção coordenada $y$
$p$	- pressão do fluido [Pa]
$P^n$	- pressão no passo de tempo $n$
$p_v$	- expoente das componentes do erro de truncamento

$Re$	- número de Reynolds
$S_n$	- termo de sequência
$S_p$	- <i>speed-up</i>
$t$	- coordenada temporal
$T_n$	- sequência, ou extrapolador
$t_{cpu}$	- tempo de CPU
$Ti$	- tempo inicial
$v_k$	- componente do vetor referente aos modos de Fourier
$v_m$	- velocidade média do fluido [m/s]
$\hat{q}^{(k)}$	- fator de convergência média empírico

### Vetores

$\mathbf{b}^n$	- condições de contorno no passo de tempo $n$
$\mathbf{f}$	- vetor de termos independentes
$\mathbf{n}$	- vetor normal
$\mathbf{u}$	- vetor de velocidade
$\mathbf{u}^n$	- campo de velocidades no passo de tempo $n$
$\mathbf{u}^t$	- campo de velocidades auxiliar
$\mathbf{r}$	- resíduo do sistema algébrico

### Matrizes

$A$	- matriz de coeficientes
$D$	- matriz diagonal
$I$	- matriz identidade
$L$	- matriz triangular inferior
$M$	- matriz não-singular
$S$	- matriz de iteração
$U$	- matriz triangular superior

## Letras Gregas

$\mu$	- viscosidade dinâmica do fluido [Pa.s]
$\nu_1$	- número de pré-suavização
$\nu_2$	- número de pós-suavização
$\Omega$	- domínio de cálculo
$\Omega^h$	- malha fina
$\Phi$	- solução analítica exata
$\phi$	- solução numérica
$\phi^{(n)}$	- solução numérica para uma variável na iteração $n$
$\rho$	- massa específica do fluido [ $kg/m^3$ ]
$\tau$	- razão de engrossamento
$\varepsilon$	- tolerância

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>15</b>
1.1	GENERALIDADES EM CFD	15
1.2	REVISÃO BIBLIOGRÁFICA	17
1.2.1	Métodos de projeção para equações de Navier-Stokes	18
1.2.2	Método <i>multigrid</i>	20
1.2.3	Técnicas de extrapolação	22
1.3	MOTIVAÇÃO	23
1.4	DEFINIÇÃO DO PROBLEMA	23
1.5	OBJETIVOS	24
<b>2</b>	<b>FUNDAMENTAÇÃO</b>	<b>25</b>
2.1	MÉTODOS NUMÉRICOS PARA A RESOLUÇÃO DE SISTEMAS DE EQUAÇÕES LINEARES	25
2.2	ERROS NUMÉRICOS	27
2.3	MÉTODO <i>MULTIGRID</i>	28
2.3.1	Princípios fundamentais	29
2.3.2	Operadores de transferência	32
2.3.3	Esquemas, ciclos e algoritmos	34
2.4	MÉTODOS DE EXTRAPOLAÇÃO	36
<b>3</b>	<b>MODELOS MATEMÁTICO E NUMÉRICO</b>	<b>39</b>
3.1	MODELO MATEMÁTICO	39
3.2	MÉTODO DE PROJEÇÃO	40
3.3	DISCRETIZAÇÃO	41
<b>4</b>	<b>RESULTADOS</b>	<b>45</b>
4.1	DADOS DE IMPLEMENTAÇÃO	45
4.2	USO DE EXTRAPOLADORES PÓS- <i>MULTIGRID</i>	46
4.3	USO DE EXTRAPOLADORES ENTRE OS CICLOS	49
4.4	TESTES ADICIONAIS	51
<b>5</b>	<b>CONCLUSÃO</b>	<b>54</b>
5.1	CONSTATAÇÕES GERAIS	54
5.2	CONTRIBUIÇÕES	55
5.3	TRABALHOS FUTUROS	55
	<b>REFERÊNCIAS</b>	<b>56</b>

## 1 INTRODUÇÃO

Os problemas de Engenharia exigem métodos cada vez mais rápidos e eficientes para encontrar soluções, pois suas aplicações se relacionam diretamente com o avanço econômico e tecnológico. Uma das áreas desafiadoras é a Dinâmica dos Fluidos Computacional (CFD - *Computational Fluid Dynamics*), devido à natureza dos problemas, normalmente representados por modelos matemáticos complexos e que retratam uma versão simplificada do fenômeno real. Por exemplo, as equações de Navier-Stokes, muito aplicadas em CFD, tem soluções analíticas conhecidas somente em alguns casos. Isto ocorre pelo fato de se tratarem de Equações Diferenciais Parciais (EDPs) não lineares, área de constantes investigações científicas e com vasto potencial de exploração matemática (FORTUNA, 2000).

Esta dissertação traz uma abordagem de técnicas matemáticas computacionais para a obtenção de soluções e busca mensurar a eficiência de diferentes métodos numéricos, especialmente o método *multigrid* associado a algumas das técnicas de interpolação encontradas na literatura. Outros métodos que serão vistos neste trabalho e compõem o algoritmo implementado são: o Método de Volumes Finitos (MVF), método de projeção e métodos iterativos para resolução de sistemas lineares<sup>1</sup>. As comparações são quantificadas através de parâmetros referentes aos erros numéricos e desempenho de processamento da CPU.

A fim de contextualizar o trabalho, este capítulo introdutório irá primeiramente apresentar a abordagem de problemas gerais de Engenharia. Em seguida, será feita uma revisão bibliográfica dos métodos numéricos que formam a base desta dissertação. Por último, o problema relacionado ao escoamento laminar bidimensional de um fluido incompressível em regime transitório ditado pelas equações de Navier-Stokes será inserido no cenário construído e os objetivos, apresentados.

### 1.1 GENERALIDADES EM CFD

Segundo Marchi (Universidade Federal de Santa Catarina, Florianópolis, 2001), existem três metodologias para abordar um problema de engenharia, que são a experimental, a analítica e a numérica. Cada qual apresenta seus pontos fortes e fracos em relação às outras. A Tab. 2 sintetiza a questão.

Não raro, os três métodos mostrados na Tab. 2 são utilizados juntos para modelar ou resolver um dado problema, em especial, de maior complexidade. A aplicação conjunta do numérico com o experimental auxilia tanto na diminuição dos erros de ambas as metodologias quanto no aprimoramento do modelo matemático de determinado fenômeno. Assim sendo, é preciso adotar a estratégia mais adequada e viável a cada problema de Engenharia.

Em relação à abordagem numérica, o campo da Engenharia que aborda problemas de dinâmicas dos fluidos com esse viés é a CFD. Os problemas desta área são relacionados a

<sup>1</sup> Vale ressaltar que o problema abordado nesta dissertação é não linear, entretanto, devido à metodologia de resolução empregada, ocorre o aparecimento de sistemas lineares a serem resolvidos.

Tabela 2 – TIPOS DE ABORDAGENS EM ENGENHARIA

Métodos experimentais	Métodos analíticos	Métodos numéricos
Nem sempre é possível representar o modelo em escala real	Atuam sobre representações matemáticas dos fenômenos	Atuam sobre representações matemáticas dos fenômenos
Os resultados são obtidos através de equipamentos de medição	Normalmente são necessárias grandes simplificações do modelo estudado para que se consiga encontrar soluções	As soluções numéricas são discretas e se aplicam a casos mais gerais que as representações analíticas
Apresentam erros experimentais	Apresentam erros de modelagem	Apresentam erros de modelagem e numéricos

Fonte: Adaptado de Marchi (Universidade Federal de Santa Catarina, Florianópolis, 2001).

fenômenos físico-químicos de fluidos em movimento, com ou sem troca de calor (FORTUNA, 2000; MALISKA, 2004).

Para se resolver numericamente um problema é necessário transformar o domínio contínuo em um domínio discreto, ou seja, dividi-lo em pontos ou pequenas regiões nas quais deverá ser resolvido o problema. A esse processo se dá o nome discretização do domínio, e o conjunto das pequenas regiões, neste trabalho também chamadas de volumes (devido à utilização do MVF), ao qual o domínio contínuo foi discretizado chama-se malha. As malhas podem, também, ser classificadas em estruturadas ou não estruturadas, cujas definições podem ser vistas em Maliska (2004).

Portanto, para se ter uma malha bidimensional estruturada, deve-se particionar o domínio contínuo representado por  $\{(x, y) \in \mathbb{R}^2 : 0 \leq x \leq L_x, 0 \leq y \leq L_y\}$  em um número finito de volumes  $N = N_x N_y$ , em que  $N_x$  e  $N_y$  são o número de volumes nas direções coordenadas  $x$  e  $y$ . Neste caso, pelo fato da malha ser uniforme por direção, cada volume tem as dimensões dadas por  $h_x = 1/N_x$  e  $h_y = 1/N_y$ . Além disso, estes volumes podem ser ordenados de maneira lexicográfica. Assim, por exemplo, uma malha estruturada com  $N_x = N_y = 8$  fica conforme a Fig. 1.

Após a etapa da discretização do domínio, torna-se necessário encontrar uma maneira de representar as equações. Para isso, utilizam-se técnicas como o Método de Diferenças Finitas (MDF) (OZISIK, 1994; PEIRÓ; SHERWIN, 2005), Método de Elementos Finitos (MEF) (ZIENKIEWICZ et al., 1977; DHATT et al., 2012) ou, o que será utilizado neste trabalho, o Método de Volumes Finitos (MVF) (EYMARD; GALLOUËT; HERBIN, 2000; VERSTEEG; MALALASEKERA, 2007). Detalhes deste último método serão mostrados na seção 3.3.

Uma característica da discretização dos modelos matemáticos de dinâmica dos fluidos através do MVF é que se pode obter sistemas lineares do tipo

$$A\mathbf{u} = \mathbf{f}, \quad (1.1)$$

em que  $A$  é a matriz dos coeficientes,  $\mathbf{f}$  é vetor de termos independentes e  $\mathbf{u}$ , vetor de incógnitas. A Eq. (1.1) pode ser resolvida através de métodos diretos, como a eliminação de Gauss, ou iterativos, como o método de Jacobi ou Gauss-Seidel. Vale destacar que esta última classe possui um custo computacional menor do que os métodos diretos para sistemas de grande porte (BURDEN; FAIRES, 2008; YOUNG, 2014).

57	58	59	60	61	62	63	64
49	50	51	52	53	54	55	56
41	42	43	44	45	46	47	48
33	34	35	36	37	38	39	40
25	26	27	28	29	30	31	32
17	18	19	20	21	22	23	24
9	10	11	12	13	14	15	16
1	2	3	4	5	6	7	8

Figura 1 – Malha estruturada com ordenação do tipo lexicográfica

Também há uma maneira mais aprimorada para resolver problemas dados pela Eq. (1.1), proporcionando uma aceleração na convergência, que é utilizando o conceito do *multigrid*. Este método se utiliza de um conjunto de malhas em que as iterações são realizadas utilizando uma hierarquia de malhas, desde o nível de maior refinamento (que é a discretização original do problema) até níveis menos refinados, através de operadores que realizam a transferência de informações entre elas (BRIGGS; HENSON; MCCORMICK, 2000). Assim, a malha em que se pretende resolver o problema, denominada *a malha mais fina*, possui mais volumes que as auxiliares utilizadas no método *multigrid*, por isso estas são chamadas de malhas mais grossas. A Fig. 2 ilustra a relação entre duas malhas com diferentes graus de refinamento.

No intuito de melhorar a convergência do método *multigrid*, pode-se ainda utilizar técnicas de extrapolação ou de aceleração de convergência associadas aos métodos iterativos (BREZINSKI, 1974), obtendo assim soluções numéricas com menor esforço computacional.

## 1.2 REVISÃO BIBLIOGRÁFICA

Esta seção apresenta o desenvolvimento das principais ferramentas numéricas utilizadas neste trabalho, como o método de projeção, o método *multigrid* e o uso de extrapoladores para acelerar a convergência de processos iterativos. Desta forma é possível contextualizar os temas abordados nesta dissertação, evidenciando os desafios que emergem ao se propor alternativas para resolver o problema ditado pelas equações de Navier-Stokes, conforme proposto.

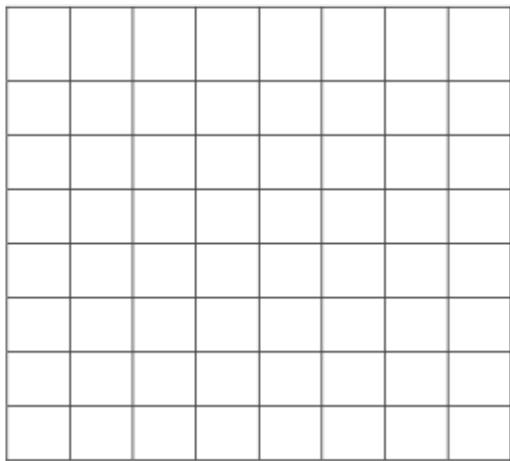
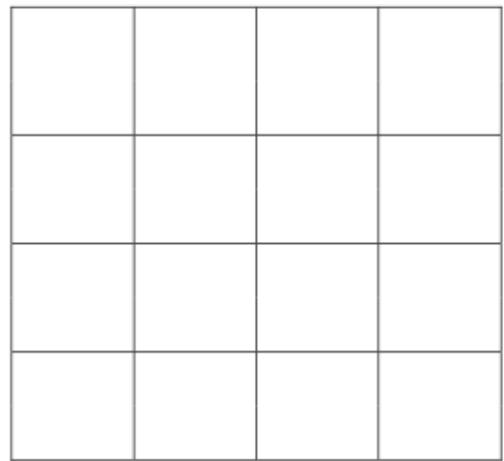
a) malha fina, com  $N_x = N_y = 8$ b) malha grossa, com  $N_x = N_y = 4$ 

Figura 2 – Exemplo de relação entre malhas

Fonte: Adaptado de Pinto (Universidade Federal do Paraná, Curitiba, 2006).

### 1.2.1 Métodos de projeção para equações de Navier-Stokes

A evolução temporal da pressão não aparece explicitada na formulação em variáveis primitivas das equações de Navier-Stokes para fluidos incompressíveis, diferente do que ocorre com a velocidade. Assim, os esquemas de solução que visam resolver as equações de maneira simultânea para a obtenção dos campos de pressão e velocidade se deparam com sistemas de equações algébricas mal condicionados, dificultando o uso de métodos iterativos. Neste sentido, alguns métodos, como o de projeção, desacoplam as soluções do campo de velocidade do campo de pressão, sendo essa uma abordagem alternativa e eficaz para a busca de soluções.

Os métodos de projeção, também conhecidos por métodos de passo fracionário, são frequentemente divididos em três classes: esquemas de correção na pressão (*pressure-correction schemes*), esquemas de correção na velocidade (*velocity-correction schemes*) e de separação consistente (*consistent splitting schemes*).

O esquema de correção na pressão é o mais simples e foi proposto por Chorin (1968) e Temam (1969). Nele, é utilizado o método de Euler na discretização temporal (BURDEN; FAIRES, 2008) e um campo de velocidades auxiliar, ou provisório, é criado. Com ele, estimam-se as velocidades do passo de tempo atual a partir das velocidades do passo anterior, sem levar em conta a pressão. Então, a pressão é calculada a partir da velocidade auxiliar, realizando assim a projeção, e são aplicadas as condições de incompressibilidade. Com o valor obtido da pressão são feitas correções nas velocidades, no tempo atual. Nessas duas etapas são resolvidas equações elípticas para a pressão e velocidades de forma desacoplada, sendo este o aspecto positivo do método. Por outro lado, inerente ao método, há uma imposição de condições de contorno de Neumann artificiais sobre a pressão, que reduzem a acurácia das soluções.

Com o objetivo de aprimorar a acurácia do algoritmo, Goda (1979) adicionou o valor do gradiente da pressão de uma iteração anterior à primeira etapa do esquema, e obteve um ganho nesse aspecto. Com esta percepção, Kan (1986) formulou um método de correção na

pressão de segunda ordem, conhecido como esquema de correção incremental da pressão na forma padrão.

Seguindo a ideia da correção incremental da pressão, Bell, Colella e Glaz (1989) desenvolveram um método de segunda ordem bastante conhecido e utilizado na literatura, que alguns autores o referenciam como método BCG (que são as iniciais dos respectivos autores). Ele tem sido utilizado para resolver problemas mais complexos, como escoamentos reativos (ALMGREN; BELL; SZYMCZAK, 1996) e multifásicos (VILLAR, Universidade Federal de Uberlândia, Uberlândia, 2007).

Embora os métodos de projeção tenham contado com diversas contribuições, a questão da acurácia continuou sendo um desafio. No trabalho de Kim e Moin (1985) apresenta-se um método que obteve melhores taxas de convergência para a pressão. Semelhante ao proposto em Chorin (1968), a pressão é ignorada na primeira etapa do algoritmo e não é feita a sua correção na segunda etapa, por isso este método também é conhecido como projeção independente da pressão (*pressure-free projection method*). Neste caso, é utilizado um esquema de avanço no tempo explícito que calcula a pressão no passo de tempo atual de modo que a equação de continuidade do próximo passo de tempo seja satisfeita.

Com taxas de convergência semelhantes às observadas em Kim e Moin (1985), há o método proposto por Timmermans, Minev e Vosse (1996), no qual foi adicionado o divergente das velocidades auxiliares na segunda etapa do algoritmo de Kan (1986). Isso fez com que as condições de contorno da pressão passassem a ser consistentes, o que se traduz em melhorias na taxa de convergência dessa variável. Neste novo algoritmo, as condições de contorno da pressão devem satisfazer o rotacional das velocidades nos contornos, e por isso o método é chamado de correção incremental na pressão na forma rotacional.

Existe uma outra família de métodos de projeção que, por sua vez, realizam correções nas velocidades e podem ser vistos em Orszag, Israeli e Deville (1986) e Karniadakis, Israeli e Orszag (1991). De maneira geral, esses métodos invertem a ordem das etapas vistas na correção da pressão. Também são encontrados com correção incremental na forma padrão e rotacional, esta última apresentada por Guermond e Shen (2003). Comparações entre os principais métodos de projeção podem ser vistos em Guermond, Minev e Shen (2006).

Através da extensão do caso de escoamentos incompressíveis para compressíveis (OSE-LEDETS, 1989), foi apresentado em Kadioglu, Klein e Minion (2008) um método de projeção numérico de quarta ordem para dinâmica de gases, com o número de Mach tendendo a zero. A discretização espacial é feita com o MVF e a temporal é baseada em correções espectrais adiadas.

Dois outros trabalhos apresentaram convergência de segunda ordem para a resolução das equações de Navier-Stokes para fluidos incompressíveis. O primeiro, que traz um novo método de separação consistente para casos com densidade variável<sup>2</sup>, foi proposto por Guer-

<sup>2</sup> Neste caso, é adicionado uma variável dependente para a densidade nas equações para fluidos incompressíveis, e mais detalhes podem ser vistos em Guermond e Salgado (2009).

mond e Salgado (2009). Seu tempo computacional é reduzido por resolver apenas uma equação por passo de tempo. Já o segundo, descrito em Griffith (2009), utiliza um método de projeção como precondicionador para o sistema linear resultante, que vem a ser resolvido pelo método dos gradientes conjugados precondicionado com o método *multigrid* geométrico. Dentre vários problemas, ele resolve também o da cavidade com tampa deslizante (GHIA; GHIA; SHIN, 1982) e apresenta resultados para Reynolds variando de 1 a  $3 \times 10^4$ .

Também existem trabalhos com discretização espacial pelo MEF, como por exemplo o de Astorino e Grandmont (2010). Neste, a discretização temporal é semi-implícita e utiliza uma estimativa analítica para o erro temporal, de modo a obter melhora na acurácia e convergência das soluções.

Em Almgren et al. (2013) encontra-se um método de quarta ordem para as discretizações espacial e temporal. Os autores sugerem uma resolução semelhante a de Kadioglu, Klein e Minion (2008), e avaliam a relação entre o método utilizado e a resolução de malha necessária para resolver problemas de escoamento turbulento. Concluíram que para determinados números de Reynolds, a resolução de malha necessária em problemas de escoamento turbulento é muito menor para o método de quarta ordem proposto do que para os de segunda ordem testados.

Como outro caso de método de projeção de quarta ordem nas discretizações espacial e temporal, há o trabalho de Zhang (2014), que também usa o MVF na discretização espacial. Para a discretização temporal é usado o método de Runge-Kutta semi-implícito e as malhas utilizadas são do tipo estruturadas com refinamento adaptativo. A condição de divergente nulo é satisfeita empregando um operador de projeção aproximado e o trabalho apresentou soluções estáveis e acuradas.

No grupo de pesquisa "Método *Multigrid*"<sup>3</sup> encontra-se o trabalho de Neundorf et al. (2016), que aplica um método de projeção com correção incremental na pressão e convergência temporal de segunda ordem para resolver um problema de escoamento laminar bidimensional de um fluido incompressível, governado pelas equações de Navier-Stokes. Vale ressaltar que este problema é o mesmo analisado nesta dissertação, porém o método de resolução deste trabalho envolve o uso de extrapoladores.

Estes métodos de projeção citados elucidam qual é o caminho viável para que se obtenha uma resolução com a eficiência esperada de métodos atuais.

### 1.2.2 Método *multigrid*

Os métodos iterativos que resolvem sistemas lineares são avaliados de acordo com sua eficiência. Esta pode ser expressa de maneira satisfatória através da taxa de convergência. A busca contínua pela melhoria desta taxa resulta na produção de uma série de trabalhos, dada a importância em resolver problemas com dimensões cada vez maiores. Como um exemplo bem sucedido, pode-se citar o método *multigrid* (BRIGGS; HENSON; MCCORMICK, 2000; WES-

<sup>3</sup> Grupo de pesquisa credenciado no Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), em atividade desde 2016.

SELING, 1995; HACKBUSCH, 1985; TROTENBERG; OOSTERLEE; SCHULLER, 2000), que desde seu desenvolvimento apresenta resultados satisfatórios na resolução de sistemas lineares e não lineares, e por isso é bastante utilizado.

A eficácia do método *multigrid* em relação aos métodos iterativos utilizados isoladamente foi discutida em Fedorenko (1964), cujos resultados obtidos evidenciam a melhora na velocidade de convergência do primeiro em relação aos demais. Em um momento seguinte, passou-se então a ser discutida a escolha dos seus parâmetros internos (como o tipo de ciclo utilizado, quantidade de pré e pós-suavizações) e como isto interfere na qualidade dos resultados obtidos. Neste sentido, para o caso de problemas lineares, Brandt (1977) evidencia as vantagens do emprego do Esquema de Correção (CS - *Correction Scheme* - Esquema de Correção) sobre o Esquema de Aproximação Completa (FAS - *Full Approximation Scheme*), no que diz respeito ao custo computacional envolvido. Isto se deve ao fato de que os CS resolve a Eq. (1.1) apenas na malha mais fina, sendo que nas demais, resolve a equação residual. Já o FAS resolve a Eq. (1.1) em todas as malhas e é indicado para problemas não lineares.

Também foram analisados os casos em que é preferível aplicar o método *multigrid*. Hackbusch (2013) afirma que os melhores resultados são obtidos em problemas totalmente elípticos, caracterizados pela representação da difusão. Já o caso oposto, em que o método tem sua eficácia reduzida, foi verificado em problemas de advecção (FERZIGER; PERIC, 2002). De maneira geral, o conjunto de escolhas de parâmetros relacionados ao método, como os operadores de transferência de informações entre malhas ou o tipo de correção utilizado, e também a caracterização correta do problema a ser resolvido, fazem com que o método tenha mais êxito sem suas aplicações (TROTENBERG; OOSTERLEE; SCHULLER, 2000). Assim, resoluções em malhas mais refinadas podem ser executadas e, por conseguinte, o erro de discretização será menor (ROACHE, 1998).

Variações do método *multigrid*, como as dos tipos geométrico e algébrico, foram comparadas em Briggs, Henson e McCormick (2000). Enquanto o primeiro leva em conta as posições geométricas dos nós na malha mais fina para realizar as transferências de informação entre elas, o segundo opera sobre os índices das incógnitas  $1, 2, \dots, N$ , ou seja, da matriz de coeficientes gerada. Isso faz com que o *multigrid* geométrico utilize apenas as informações da discretização das diversas malhas geradas para as operações, sendo adequado a malhas estruturadas e o algébrico, ao oposto, ou seja, malhas não estruturadas. A aplicação vista em Briggs, Henson e McCormick (2000) foi feita em uma equação de Poisson bidimensional com malha cartesiana e condições de contorno de Dirichlet. Neste caso, concluiu-se que o *multigrid* geométrico consome menos memória e tempo computacional que o *multigrid* algébrico, devendo-se restringir o uso deste segundo método apenas para os casos em que o primeiro apresente-se falho.

Por se tratar de uma ferramenta computacional, nota-se que a maior parte do esforço inovador relacionado ao método *multigrid* consiste em ampliar sua área de aplicação, estendendo a uma gama de problemas, e ao mesmo tempo ter o controle do custos de CPU en-

volvidos, reduzindo-os sempre que possível. Diretrizes com esse perfil permeiam muitos dos trabalhos em CFD e podem ser notadas neste trabalho.

### 1.2.3 Técnicas de extrapolação

Dentre as alternativas empregadas para acelerar a convergência dos métodos iterativos, além do *multigrid*, há metodologias que aplicam transformações de sequências e dão origem às técnicas de extrapolação. Sucintamente, uma sequência convergente é transformada em uma nova que, por sua vez, aproxima-se mais rapidamente de seu limite. Mais detalhes encontram-se na seção 2.4.

Em Evans (1963) encontra-se uma aceleração linear do método iterativo de Aitken baseadas em cinco pontos para equações diferenciais elípticas. Esta aplicação apresentou resultados compatíveis ao método de Sobre Relaxação Sucessiva (SOR - *Sucessive Over Relaxation*) acelerado. Outros três métodos de extrapolação, Epsilon do tipo escalar e vetorial, e por polinômio mínimo, são analisados em Skelboe (1980). Nesse trabalho é apresentado um teorema que garante a convergência quadrática para problemas lineares.

A extrapolação linear de sequências oscilatórias é explorada em Mitin (1985), em casos de resolução de sistemas de equações. Nesse trabalho, são propostos novos métodos para acelerar a convergência e os resultados obtidos são comparados com outras técnicas, validando assim o que foi produzido.

Um problema de valor de contorno modelado por uma equação de segunda ordem e com singularidades no termo fonte é visto em Lima e Graça (1995). A proposta de resolução é dada através aceleração de convergência pelo algoritmo Epsilon. Foram obtidos bons resultados quando comparados à extrapolação de Richardson. Em Jbilou e Sadok (2000) são analisadas as extrapolações do tipo vetorial, como as polinomiais e os algoritmos Epsilon. Apesar de se mostrarem mais caros computacionalmente, os algoritmos Epsilon mostraram maior capacidade de generalizar de forma simples e linear os problemas não lineares.

Como as extrapolações do tipo Epsilon se mostraram amplamente eficientes, elas continuam sendo alvo de diversas pesquisas. No trabalho de Mesgarani, Aghazadeh e Parmour (2010) foram utilizadas as extrapolações Aitken e Epsilon após realizar uma extrapolação de Richardson. Verificou-se que as duas primeiras, aplicadas somente uma vez, obtém melhores resultados que uma repetida aplicação da de Richardson. Já em Gao et al. (2010) são vistas técnicas de restauração de imagens, com uso dos extrapoladores Epsilon topológico e Rho na aceleração do algoritmo de Richardson-Lucy. Nesse caso, o algoritmo Rho se mostrou mais eficiente em acelerar a convergência.

Vale destacar também as pesquisas relacionadas às técnicas de extrapolação dos membros do grupo de pesquisa "Método *Multigrid*". Por exemplo, o uso de extrapoladores associados ao método *multigrid* para problemas como o da cavidade quadrada com tampa ditado pelas equações de Burgers, ou de condução de calor linear bidimensional, governado pela equação

de Poisson, com condições de contorno de Dirichlet, que pode ser visto em Anunciação et al. (2016).

Percebe-se com essa revisão que o estudo de técnicas de extrapolação se mostra amplamente discutido e apresenta potencial em contribuir nas aplicações em problemas de Engenharia, inclusive em CFD.

### 1.3 MOTIVAÇÃO

A necessidade constante de métodos aprimorados em CFD, seja para resolver um problema de grande complexidade (como os não lineares ou com sistemas de equações fortemente acopladas) ou com o maior números de pontos possíveis no domínio, serve como prumo para pesquisas da área. Perante isso, foi identificado um problema que apresenta tais características e é explorado neste trabalho. Trata-se do escoamento laminar bidimensional de um fluido incompressível em regime transitório representado pelas equações de Navier-Stokes. Além de desafiadora, também há um leque de aplicações práticas em Engenharia que dependem da obtenção de soluções deste tipo de problema. Como exemplos, pode-se citar a projeção de usinas hidrelétricas ou eólicas, modelagens aeroespaciais e outros tipos de aplicações tais como a computação gráfica, previsão do tempo, o estudo do *El Niño* e fluxo sanguíneo. Observa-se a importância desta formulação matemática e seu respectivo potencial.

Também é possível mesclar diferentes técnicas no desenvolvimento de um método numérico a fim de maximizar sua eficiência. Portanto, uma questão que se mostra aberta à investigação é a de como as diferentes técnicas de extrapolação influenciam na taxa de convergência e no consumo de recursos computacionais quando usadas em conjunto ao método *multigrid*, que por si só já é um poderoso acelerador de métodos iterativos. Mais especificamente, a dúvida que surge é: será possível aprimorar a resolução formada por métodos de projeção, discretização por volumes finitos e com uso de *multigrid*?

O uso de extrapoladores associado ao método *multigrid* já se mostrou promissor em outros casos, como em problemas de transferência de calor bidimensional descritos por equações de Poisson e de Burgers (ANUNCIAÇÃO et al., 2016). É fundamental utilizar essa estratégia em novas equações e, assim, verificar se os resultados obtidos manterão os níveis de êxito observados.

### 1.4 DEFINIÇÃO DO PROBLEMA

Esta dissertação se desenvolve a partir das estratégias em resolver as equações de Navier-Stokes que modelam um escoamento laminar bidimensional de um fluido incompressível em regime transitório. Uma vez definida a discretização a ser utilizada, neste caso o MVF, e o algoritmo base para a resolução, que é da família dos métodos de projeção, pode-se inserir ainda alguns recursos de aceleração de convergência. São eles: o método *multigrid* e as diferentes técnicas de extrapolação associadas ao processo iterativo. Essas metodologias, por sua

vez, oferecem vantagens computacionais que podem ser mensuradas através do tempo de CPU, norma do resíduo, normas do erro de iteração, fator de convergência empírico e memória de armazenamento.

A maneira escolhida para analisar o comportamento dos diferentes extrapoladores é aplicando-os na resolução dos sistemas lineares provenientes do método de projeção, referentes às variáveis de velocidade. Estes sistemas aparecem em cada passo de tempo do problema e representam o cálculo do incremento das variáveis. Assim, será possível descobrir se a abordagem surtirá efeitos positivos no resultado encontrado ao fim da simulação.

Os testes são realizados em duas metodologias. A primeira tem o propósito de filtrar dentre os extrapoladores aquele que se mostra mais promissor para o problema, e já será possível notar um ganho nos resultados então obtidos. Aquele que se destacar será utilizado também na segunda etapa. Nesta, o extrapolador é inserido de forma mais integrada ao algoritmo, possibilitando assim um aumento no desempenho por ele proporcionado. Essa análise levará a concluir qual técnica é a mais apropriada para ter vantagens em termos computacionais e de acurácia nas soluções.

Maiores detalhes são vistos no capítulo de fundamentação (capítulo 2) e também no de resultados (capítulo 4).

## 1.5 OBJETIVOS

Algumas metas foram definidas para este trabalho, cuja proposta é validar diferentes extrapoladores associados ao método *multigrid*.

Como objetivo geral, pretende-se:

- melhorar o tempo computacional de resolução e reduzir o erro de iteração através do uso de extrapoladores;

Já os objetivos específicos são dados por:

- resolver um problema de escoamento laminar bidimensional de um fluido incompressível em regime transiente modelado pelas equações de Navier-Stokes;
- mensurar o tempo computacional obtido com diferentes extrapoladores encontrados na literatura;
- mensurar os erros da soluções obtidas com tais extrapoladores;
- selecionar a técnica de extração mais eficaz quando associada ao método *multigrid* a fim de resolver o problema proposto.

## 2 FUNDAMENTAÇÃO

Este capítulo apresenta os conceitos teóricos a respeito dos métodos numéricos aplicados às equações diferenciais: métodos iterativos básicos, erro numérico, princípios do método *multigrid* e técnicas de interpolação. Desta forma, fica clara a base numérica em que se estrutura o objeto da dissertação.

### 2.1 MÉTODOS NUMÉRICOS PARA A RESOLUÇÃO DE SISTEMAS DE EQUAÇÕES LINEARES

Ao se resolver as equações diferenciais de forma numérica é necessário substituir os termos das derivadas por aproximações. Desta forma, o problema analítico de propriedade contínua se converte em um análogo, só que desta vez com característica discreta. Então, o domínio espacial passa a ser dividido em um número finito de pontos ou, no caso deste trabalho, de volumes. Esse novo domínio discretizado passa a ser chamado de malha computacional, ou apenas malha.

No Método de Volumes Finitos realiza-se a integração das equações diferenciais, na forma conservativa sobre um volume de controle elementar qualquer (MALISKA, 2004), e será tratado com mais detalhes na seção 3.3. Neste trabalho, ocorre o particionamento de um domínio bidimensional cartesiano  $\{(x, y) \in \mathbb{R}^2 : 0 \leq x \leq L_x \text{ e } 0 \leq y \leq L_y\}$  em malhas estruturadas do tipo  $N = N_x N_y$ , em que  $N_x$  e  $N_y$  são o número de volumes nas direções coordenadas  $x$  e  $y$ ,  $L_x$  e  $L_y$  representam o comprimento do domínio nas direções  $x$  e  $y$ , respectivamente. Com esta discretização, as EDPs passarão a ser resolvidas via sistemas lineares do tipo representado na Eq. (1.1). Caso o sistema de equações apresente características não lineares, usa-se a seguinte notação:

$$A(\mathbf{u}) = \mathbf{f}. \quad (2.1)$$

Em ambos os casos, a matriz  $A$  pode ser dividida em:

$$A = M - N', \quad (2.2)$$

com  $M$  não-singular.

Um método iterativo básico que resolve a Eq. (2.1) se origina da expressão

$$M\mathbf{u}^{(m+1)} = N'\mathbf{u}^{(m)} + \mathbf{f}, \quad (2.3)$$

em que o superescrito de  $u$  representa a iteração. Assim, a resolução da incógnita passa a ser dada por

$$\mathbf{u}^{(m+1)} = S\mathbf{u}^{(m)} + M^{-1}\mathbf{f}, \quad (2.4)$$

em que  $S = M^{-1}N'$ ,  $N' = M - A$ , e a matriz  $S$  é chamada de matriz de iteração do método dado pela Eq. (2.4). Esta formulação dá origem aos suavizadores Jacobi e Gauss-Seidel.

O suavizador Gauss-Seidel utiliza para o cálculo da componente  $\mathbf{u}^{(m+1)}$  os valores mais recentes de  $\mathbf{u}$ . Para isso,  $M = L + D$ , tal que  $L$  é a matriz que contém os elementos da parte inferior de  $A$  e  $D$ , a matriz que contém os elementos da diagonal de  $A$ . E também  $N' = -U$ , com  $U$  sendo a parte triangular superior de  $A$ . Assim, a matriz de iteração associada se torna

$$S_{GS} = I - M^{-1}A = I - (L + D)^{-1}A, \quad (2.5)$$

com  $I$  sendo a matriz identidade. O processo iterativo do método Gauss-Seidel passa a ser dado por:

$$\mathbf{u}^{(m+1)} = S_{GS}\mathbf{u}^{(m)} + (L + D)^{-1}\mathbf{f}. \quad (2.6)$$

Para os casos em que os elementos da diagonal de  $A$  são diferentes de zero, basta tomar uma estimativa inicial para  $\mathbf{u}^{(0)}$  e calcular  $\mathbf{u}^{m+1}$  por:

$$u_i^{(m+1)} = \frac{1}{a_{ii}} \left[ \mathbf{f}_i - \sum_{j=1}^{i-1} a_{ij}u_j^{(m+1)} - \sum_{j=i+1}^N a_{ij}u_j^{(m)} \right], \quad i = 1, \dots, N. \quad (2.7)$$

A Eq. (2.7) é utilizada em volumes ordenados de maneira lexicográfica, à exemplo da Fig. 1. Uma outra forma de ordenar os volumes conduz a um novo método, denominado Gauss-Seidel *Red-Black* (GS-RB). Este nome é devido à disposição dos dados aparentar um tabuleiro de xadrez. Um exemplo bidimensional, de tamanho  $N_x = N_y = 6$ , pode ser visto na Fig. 3.

34	16	35	17	36	18
13	31	14	32	15	33
28	10	29	11	30	12
7	25	8	26	9	27
22	4	23	5	24	6
1	19	2	20	3	21

Figura 3 – Ordenação *red-black* para o caso bidimensional

Assim, para atualizar o valor de cada volume são necessários os valores de seus vizinhos próximos, cujos valores são ser obtidos em etapa distinta. Isso se deve ao fato de que o método GS-RB atualiza as incógnitas alternadamente, ora as representadas dentro de círculos,

e ora as demais, ao exemplo da Fig. 3. Essa característica favorece o processamento paralelo do método, tornando-o mais vantajoso que o Gauss-Seidel com ordenação lexicográfica em alguns casos. Na computação serial também mostra-se vantajoso em relação ao lexicográfico, como por exemplo, ao apresentar menor tempo de CPU na resolução de problemas de difusão de calor modelados pelas equações de Laplace e Poisson (OLIVEIRA, Universidade Federal do Paraná, Curitiba, 2010). Esses fatores motivaram a escolha do GS-RB para ser utilizado neste trabalho. Mais detalhes sobre o método e estudo de seus parâmetros podem ser vistos em Trottenberg, Oosterlee e Schuller (2000) e Gonçalves (Universidade Federal do Paraná, Curitiba, 2013).

## 2.2 ERROS NUMÉRICOS

Ao lidar com métodos numéricos também é necessário conhecer os erros que estão envolvidos no processo, a fim de minimizá-los ou evitá-los. Ferziger e Peric (2002) definem o erro numérico como sendo a diferença entre a solução analítica exata ( $\Phi$ ) de uma variável e a sua solução numérica ( $\phi$ ):

$$E(\phi) = \Phi - \phi. \quad (2.8)$$

Este erro é gerado por uma combinação de outros erros (MARCHI, Universidade Federal de Santa Catarina, Florianópolis, 2001), dentre eles: os erros de truncamento ( $E_h$ ), erros de iteração ( $E_n$ ), erros de arredondamento ( $E_\pi$ ) e erros de programação ( $E_{Pr}$ ). De maneira geral, tem-se:

$$E(\phi) = E(E_h, E_n, E_\pi, E_{Pr}). \quad (2.9)$$

O erro de truncamento ocorre ao se truncar um processo infinito, ou seja, ao aproximar um modelo matemático contínuo por um modelo matemático discreto (ROACHE, 1998). Em outras palavras, ele é obtido entre a diferença da solução analítica exata de uma variável e a sua representação numérica dada por algum método de discretização (TANNEHILL; ANDERSON; PLETCHER, 1997).

Dividindo um domínio contínuo em pontos com espaçamento  $h$ , o erro de truncamento é representado genericamente por (FERZIGER; PERIC, 2002):

$$E_h = C_0 h^{p_0} + C_1 h^{p_1} + C_2 h^{p_2} + C_3 h^{p_3} + \dots, \quad (2.10)$$

em que os coeficientes reais  $C_0, C_1, C_2, C_3, \dots$ , podem ser funções da variável dependente e de suas derivadas, porém independem de  $h$ . Os superíndices  $p_v$ , com  $v = 0, 1, 2, 3, \dots$ , são expoentes inteiros positivos dos termos não nulos e seguem a relação:  $1 \leq p_0 < p_1 < p_2 < p_3 \dots$ . Não raro, os termos  $p_v$  formam uma progressão aritmética. O menor expoente  $p_0$  é chamado ordem assintótica.

Para  $h \rightarrow 0$ , o  $E_h$  é dominado pelo termo  $C_0 h^{p_0}$  da Eq. (2.10), que passa a ser o principal componente de  $E_h$ . Isso significa que para valores de  $h$  próximos a zero, a redução de  $E_h$  apresentará a ordem  $p_0$ . Ao minimizar ou eliminar outras fontes de erro tanto quanto

possível, o erro de truncamento recebe a denominação de erro de discretização ( $E_h$ ), conforme visto em Ferziger e Peric (2002). Pode-se afirmar que  $E_h$  está associado à solução numérica  $\phi$  na malha uniforme com um espaçamento  $h$  sobre o domínio  $\Omega$ , também chamada de malha  $\Omega^h$ .

O erro de iteração ( $E_n$ ) é originado do emprego de métodos iterativos para a resolução de sistemas algébricos. Então tem-se

$$E_n = \Phi - \phi^{(n)}, \quad (2.11)$$

com  $\phi^{(n)}$  representando uma solução numérica para uma variável na iteração  $n$ . Este tipo de erro ocorre, por exemplo, ao resolver um sistema linear através dos suavizadores Gauss-Seidel ou Jacobi. Para os casos em que  $n \rightarrow \infty$ , tem-se  $E_n \rightarrow 0$ , tornando-se possível controlar os efeitos de  $E_n$  em determinada simulação numérica.

A representação finita dos números reais em computadores trazem à tona os erros de arredondamento ( $E_\pi$ ). Também nota-se um aumento de  $E_\pi$  conforme aumenta o número de operações matemáticas realizadas, devido ao processo de propagação de erros. A forma de contorná-los é aumentando a precisão para representar as variáveis, porém isto acarreta no aumento do uso de memória computacional utilizada para a realização dos cálculos (MARCHI, Universidade Federal de Santa Catarina, Florianópolis, 2001).

O uso incorreto de um modelo numérico na aproximação de um modelo matemático, os erros gerados na implementação do modelo numérico em um programa computacional ou os cometidos no uso do programa computacional durante a obtenção da solução numérica e outras eventuais fontes de erro, geram os chamados erros de programação ( $E_{Pr}$ ) (ROACHE, 1998). Alguns testes podem ser feitos a fim de evitá-los, segundo Maliska (2004) e Roache (1998). Estes autores também fazem recomendações tais como: implementar um programa específico antes de generalizá-lo; organizar o algoritmo em módulos; verificar a convergência do método primeiramente em uma malha pequena (grosseira); e resolver um problema utilizando a técnica de soluções fabricadas, em que a solução analítica é conhecida, para comparar as ordens de convergência obtidas.

Com as resoluções propostas nesta dissertação será possível obter a redução de algumas fontes de erro, como por exemplo, o de iteração. Isso é feito com uso de aceleradores de convergência, tanto o método *multigrid* quanto o uso de extrapoladores.

### 2.3 MÉTODO *MULTIGRID*

Esta seção descreverá os conceitos relacionados ao método *multigrid*, a começar pelos mecanismos pelos quais esse tipo de abordagem atua, até uma breve descrição de seu funcionamento como um todo.

### 2.3.1 Princípios fundamentais

Para discutir o funcionamento do método *multigrid*, deve-se primeiro conhecer a atuação dos suavizadores relacionados à redução de erros em processos iterativos. Considerando um sistema de equações algébricas do tipo Eq. (1.1), será utilizada nesta seção a nomenclatura **u** para soluções exatas e **v** para as aproximações numéricas.

Ao estudar o comportamento dos erros do sistema Eq. (1.1), pode-se considerar o sistema linear homogêneo  $A\mathbf{u} = 0$  (BRIGGS; HENSON; MCCORMICK, 2000) e usar estimativas iniciais arbitrárias, dando início ao esquema iterativo, também chamado esquema de relaxação. Assim, a solução exata  $\mathbf{u} = 0$  é conhecida<sup>1</sup> e o erro da aproximação  $\mathbf{v}$  passa a ser  $-\mathbf{v}$ .

Utiliza-se uma estimativa inicial do tipo

$$v_k = \left[ \operatorname{sen} \left( \frac{jk\pi}{N} \right) \right]_j, \quad 0 \leq j \leq N, \quad 1 \leq k \leq N-1, \quad (2.12)$$

em que  $v_k$  são as componentes do vetor  $\mathbf{v}$ . Vetores com esse tipo de estimativa passam a se chamar modos de Fourier, dando origem à nomenclatura *análise de erros de Fourier*, e  $N$  é o número de pontos ou volumes da malha. Para cada  $k$  fixo, tem-se que o índice  $j$  denota a  $j$ -ésima componente do vetor  $\mathbf{v}_k$ . O inteiro  $k$  é denominado número de ondas ou frequência, e indica o número de "meios senos" que constituem o vetor  $\mathbf{v}_k$  no domínio do problema. A Fig. 4 exemplifica as estimativas iniciais  $\mathbf{v}_1$ ,  $\mathbf{v}_3$  e  $\mathbf{v}_6$ . Observa-se que valores pequenos de  $k$  reproduzem ondas longas e suaves, já os valores elevados, ondas curtas e oscilatórias.

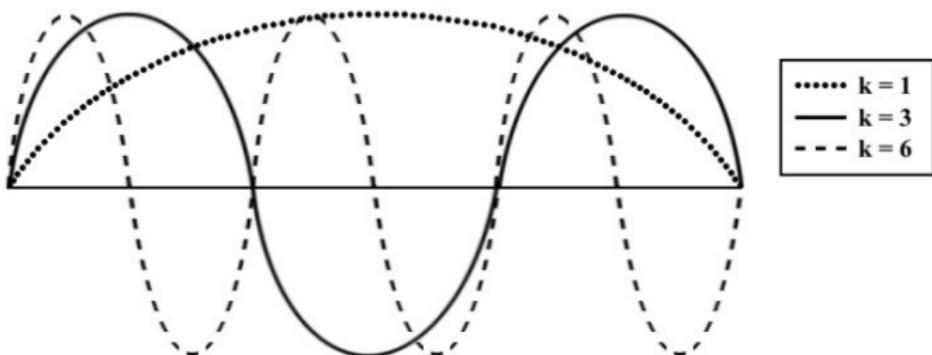


Figura 4 – Modos de Fourier com  $k = 1, 3, 6$

Fonte: Adaptado de (BRIGGS; HENSON; MCCORMICK, 2000).

Os valores de  $k$  no intervalo  $1 \leq k < (N/2)$  dão origem aos denominados modos de Fourier de baixa frequência, ou modos suaves. De maneira análoga, para  $(N/2) \leq k \leq (N-1)$ , têm-se os modos de Fourier de alta frequência, ou modos oscilatórios (BRIGGS; HENSON; MCCORMICK, 2000).

Conforme visto em Briggs, Henson e McCormick (2000) e em Trottenberg, Oosterlee e Schuller (2000), os métodos iterativos como Jacobi ponderado (com fator de ponderação

<sup>1</sup> A solução  $\mathbf{u} = 0$  refere-se ao sistema linear homogêneo, pois conhecendo o erro para este tipo de problema, é possível conhecer o comportamento do erro para equações do tipo Eq. (1.1).

representado por  $\omega$ ) e Gauss-Seidel possuem propriedades de suavização, o que significa que são capazes de reduzir rapidamente os erros de alta frequência, e essa característica é essencial para o funcionamento do método *multigrid*. Essa redução ocorre logo nas primeiras iterações, e na sequência essa propriedade vai se deteriorando. Este comportamento pode ser observado na Fig. 5.

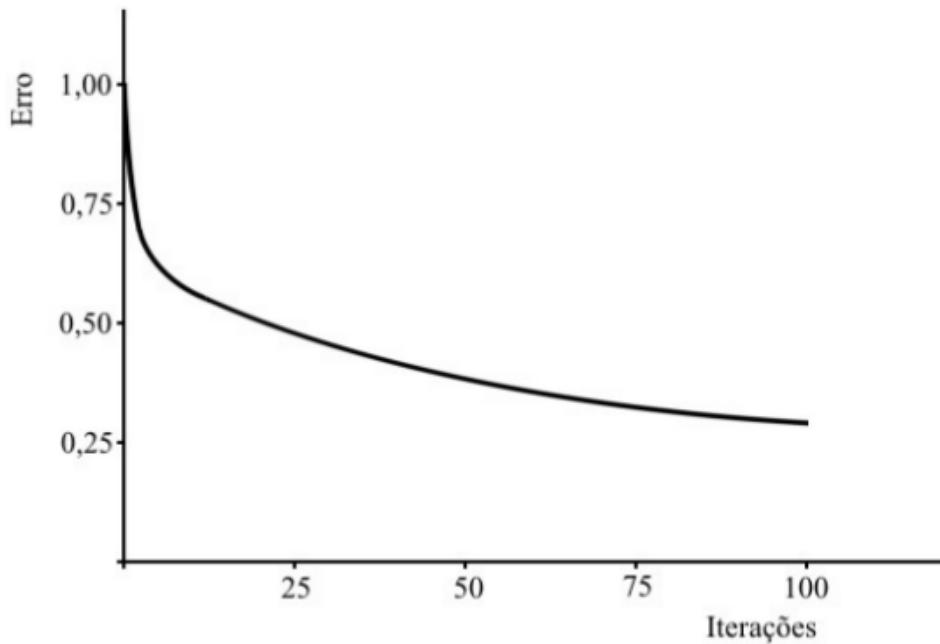


Figura 5 – Jacobi ponderado com  $\omega = 2/3$  aplicado ao problema de condução de calor unidimensional, com  $N=64$ , estimativa inicial igual a  $(v_1 + v_6 + v_{32})/3$ , 100 iterações e o erro mensurado em valor absoluto

Fonte: Briggs, Henson e McCormick (2000).

Em poucas palavras, o êxito do método *multigrid* é devido à transferência da solução para malhas cada vez mais grossas, à medida que a força de suavização vai se perdendo na malha atual. Essa transição faz com que os modos suaves fiquem mais oscilatórios, permitindo que a aplicação dos suavizadores tenha maior impacto na suavização. Esse processo ocorre até que se atinja uma malha estabelecida ou a mais grossa possível. Ao fim, a informação passa pelo caminho inverso, sendo transferida até chegar à malha original do problema. As informações são transferidas entre as malhas por operadores de transferência.

Pode-se definir a razão de engrossamento no caso bidimensional e com malhas uniformes por  $\tau = H/h$ , em que  $h$  representa a distância entre dois pontos consecutivos na malha fina  $\Omega^h$  e  $H$ , o equivalente na malha imediatamente mais grossa  $\Omega^H$ . A razão  $\tau = 2$  é uma das mais utilizadas na literatura. Segundo Brandt (1977), essa razão torna o processo de interpolação mais econômico e facilita a implementação do algoritmo. Em Briggs, Henson e McCormick (2000) consta que em geral  $\tau \neq 2$  não apresenta vantagens significativas. A Fig. 6 mostra uma sequência de quatro malhas com razão de engrossamento  $\tau = 2$  e será a adotada nesta dissertação.

Neste trabalho, como foi dito anteriormente, será adotado o MVF. Esse método será usado com malhas desencontradas (MALISKA, 2004), ou seja, as variáveis de interesse se localizam nas faces e no interior de cada volume de controle, conforme mostrado na Fig. 7.

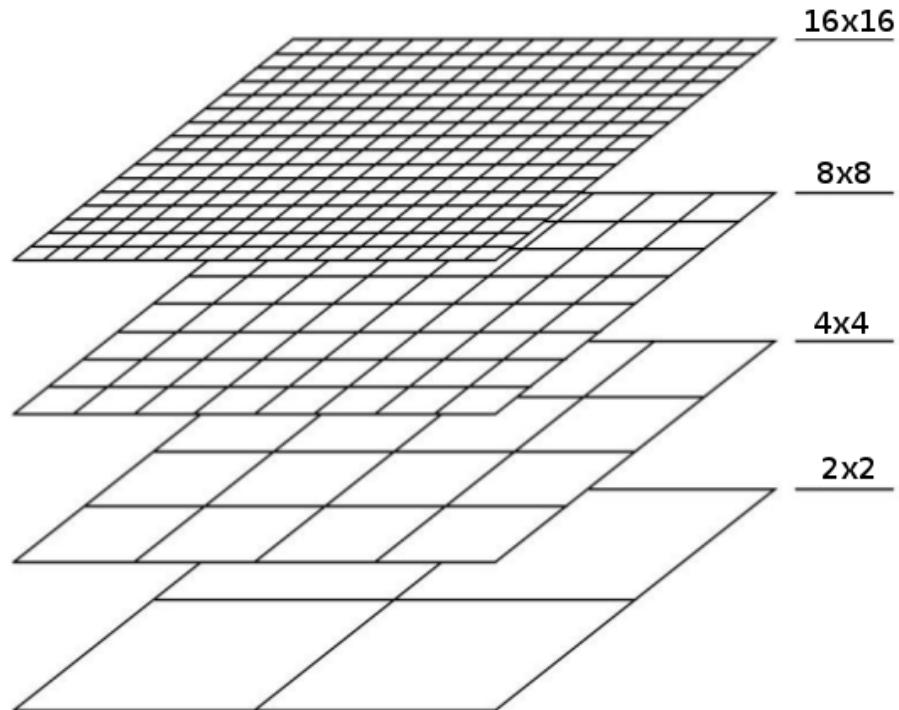


Figura 6 – Processo de geração de malhas

Fonte: Adaptado de <http://www.mgnet.org/mgnet/tutorials/xwb/mg.html>.

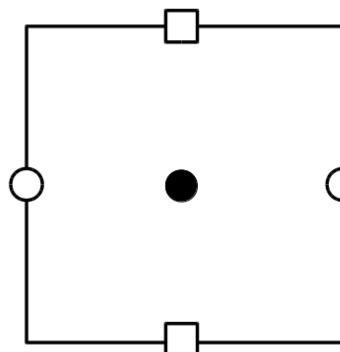


Figura 7 – Volume de controle em uma malha desencontrada, no qual os símbolos  $\bullet$ ,  $\circ$  e  $\square$  representam variáveis distintas

Vale ressaltar que a taxa de convergência ideal do método *multigrid* pode ser obtida independente do tamanho da malha, ou seja, da quantidade de volumes (FERZIGER; PERIC, 2002). Para que o método seja altamente eficiente, diversos autores recomendam usar tantos níveis de malha quanto possível (TANNEHILL; ANDERSON; PLETCHER, 1997; PINTO; MARCHI, 2007; SANTIAGO; MARCHI, 2007)

### 2.3.2 Operadores de transferência

O método *multigrid* utiliza dois tipos de operadores: de restrição e de prolongação. Os de restrição transferem informações da malha fina  $\Omega^h$  para a malha grossa  $\Omega^{2h}$  e são denotados por  $I_h^{2h}$ . Para o caso oposto, em que a transferência é da malha grossa  $\Omega^{2h}$  para a fina  $\Omega^h$ , são usados os operadores de prolongação, denotados por  $I_{2h}^h$ .

Existem vários tipos de operadores para ambos os casos (WESSELING, 1995; BRIGGS; HENSON; MCCORMICK, 2000; TROTTERBERG; OOSTERLEE; SCHULLER, 2000). Um operador de restrição comumente utilizado é o de ponderação completa (VILLAR, Universidade Federal de Uberlândia, Uberlândia, 2007), definido por  $[I]_h^{2h} \mathbf{v}^h = \mathbf{v}^{2h}$ . Segue listada a formulação para o caso bidimensional, no caso da variável de interesse  $\Phi$ , no centro do volume ( $\Phi_\bullet$ ), conforme a Fig. 7:

$$\Phi_{\bullet(i)}^{2h} = \frac{\Phi_{\bullet(i)}^h + \Phi_{\bullet(i+1)}^h + \Phi_{\bullet(i+Nxi)}^h + \Phi_{\bullet(i+Nxi+1)}^h}{4},$$

no qual  $\Phi_{\bullet(i)}$  é avaliada no ponto  $i$  e  $Nxi$  é a quantidade de elementos do tipo  $\bullet$  na direção  $x$ , conforme ilustrado na Fig. 8.

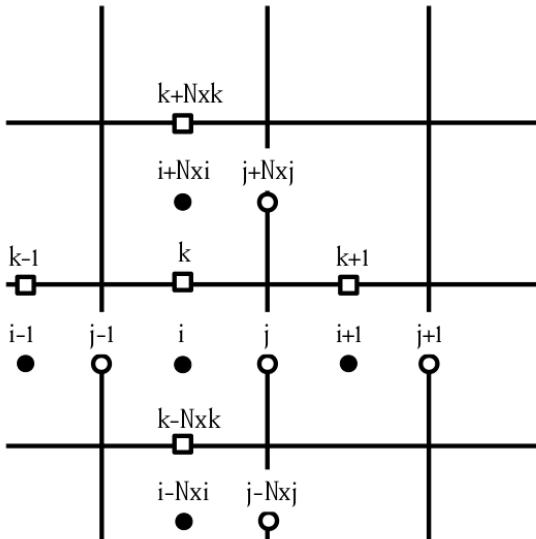


Figura 8 – Posições das variáveis  $\bullet$ ,  $\circ$  e  $\square$  na ordem lexicográfica

As formulações para o caso bidimensional de ponderação completa das variáveis de interesse  $\Phi$ , nas faces dos volumes ( $\Phi_\circ$  e  $\Phi_\square$ ), são:

$$\begin{aligned}\Phi_{\circ(j)}^{2h} &= \frac{\Phi_{\circ(j)}^h + \Phi_{\circ(j+Nxj)}^h}{2}, \\ \Phi_{\square(k)}^{2h} &= \frac{\Phi_{\square(k)}^h + \Phi_{\square(k+1)}^h}{2},\end{aligned}$$

respectivamente, no qual  $\Phi_{\circ(j)}$  é avaliada no ponto  $j$ ,  $\Phi_{\square(k)}$  é avaliada no ponto  $k$ ,  $Nxj$  é a quantidade de elementos do tipo  $\circ$  na direção  $x$  e  $Nxk$  é a quantidade de elementos do tipo  $\square$  na direção  $x$ .

Quando um operador de restrição é aplicado nos modos de Fourier suaves, o número de ondas permanece exatamente o mesmo na malha grossa. Entretanto, o número de volumes será menor, logo o modo de Fourier está relativamente mais oscilatório, viabilizando a aplicação do método *multigrid*.

Em relação aos operadores de prolongação, um dos mais utilizados e eficientes é o de interpolação bilinear (VILLAR, Universidade Federal de Uberlândia, Uberlândia, 2007), definido por  $[I]_{2h}^h \mathbf{v}^{2h} = \mathbf{v}^h$ . Para o caso bidimensional, no caso da variável de interesse  $\Phi$ , no centro do volume ( $\Phi_{\bullet}$ ):

$$\begin{aligned}\Phi_{\bullet(i)}^h &= \frac{9\Phi_{\bullet(i)}^{2h} + 3\Phi_{\bullet(i-1)}^{2h} + 3\Phi_{\bullet(i-Nxi)}^{2h} + \Phi_{\bullet(i-Nxi-1)}^{2h}}{16}, \\ \Phi_{\bullet(i+1)}^h &= \frac{9\Phi_{\bullet(i)}^{2h} + 3\Phi_{\bullet(i+1)}^{2h} + 3\Phi_{\bullet(i-Nxi)}^{2h} + \Phi_{\bullet(i-Nxi+1)}^{2h}}{16}, \\ \Phi_{\bullet(i)}^h &= \frac{9\Phi_{\bullet(i)}^{2h} + 3\Phi_{\bullet(i-1)}^{2h} + 3\Phi_{\bullet(i+Nxi)}^{2h} + \Phi_{\bullet(i+Nxi-1)}^{2h}}{16}, \\ \Phi_{\bullet(i)}^h &= \frac{9\Phi_{\bullet(i)}^{2h} + 3\Phi_{\bullet(i+1)}^{2h} + 3\Phi_{\bullet(i+Nxi)}^{2h} + \Phi_{\bullet(i+Nxi+1)}^{2h}}{16},\end{aligned}$$

no qual  $\Phi_{\bullet(i)}$  é avaliada no ponto  $i$  e  $Nxi$  é a quantidade de elementos do tipo  $\bullet$  na direção  $x$ , conforme ilustrado na Fig. 8.

As formulações para o caso de interpolação bilinear das variáveis de interesse  $\Phi$ , nas faces dos volumes ( $\Phi_{\circ}$  e  $\Phi_{\square}$ ), são:

$$\begin{aligned}\Phi_{\circ(j)}^h &= \frac{3\Phi_{\circ(j)}^{2h} + \Phi_{\circ(j-Nxj)}^{2h}}{4}, \\ \Phi_{\circ(j-1)}^h &= \frac{3\Phi_{\circ(j)}^{2h} + 3\Phi_{\circ(j-1)}^{2h} + \Phi_{\circ(j-Nxj)}^{2h} + \Phi_{\circ(j-Nxj-1)}^{2h}}{8}, \\ \Phi_{\circ(j+1)}^h &= \frac{3\Phi_{\circ(j)}^{2h} + 3\Phi_{\circ(j+1)}^{2h} + \Phi_{\circ(j-Nxj)}^{2h} + \Phi_{\circ(j-Nxj+1)}^{2h}}{8}, \\ \Phi_{\circ(j+Nxj)}^h &= \frac{3\Phi_{\circ(j)}^{2h} + \Phi_{\circ(j+Nxj)}^{2h}}{4}, \\ \Phi_{\circ(j+Nxj-1)}^h &= \frac{3\Phi_{\circ(j)}^{2h} + 3\Phi_{\circ(j-1)}^{2h} + \Phi_{\circ(j+Nxj)}^{2h} + \Phi_{\circ(j+Nxj-1)}^{2h}}{8}, \\ \Phi_{\circ(j+Nxj+1)}^h &= \frac{3\Phi_{\circ(j)}^{2h} + 3\Phi_{\circ(j+1)}^{2h} + \Phi_{\circ(j+Nxj)}^{2h} + \Phi_{\circ(j+Nxj+1)}^{2h}}{8},\end{aligned}$$

e

$$\begin{aligned}
\varPhi_{\square(k)}^h &= \frac{3\varPhi_{\square(k)}^{2h} + \varPhi_{\square(k-1)}^{2h}}{4}, \\
\varPhi_{\square(k+Nxk)}^h &= \frac{3\varPhi_{\square(k)}^{2h} + 3\varPhi_{\square(k+Nxk)}^{2h} + \varPhi_{\square(k-1)}^h + \varPhi_{\square(k+Nxk-1)}^{2h}}{8}, \\
\varPhi_{\square(k-Nxk)}^h &= \frac{3\varPhi_{\square(k)}^{2h} + 3\varPhi_{\square(k-Nxk)}^{2h} + \varPhi_{\square(k-1)}^h + \varPhi_{\square(k-Nxk-1)}^{2h}}{8}, \\
\varPhi_{\square(k+1)}^h &= \frac{3\varPhi_{\square(k)}^{2h} + \varPhi_{\square(k+1)}^{2h}}{4}, \\
\varPhi_{\square(k-Nxk+1)}^h &= \frac{3\varPhi_{\square(k)}^{2h} + 3\varPhi_{\square(k-Nxk)}^{2h} + \varPhi_{\square(k+1)}^h + \varPhi_{\square(k-Nxk+1)}^{2h}}{8}, \\
\varPhi_{\square(k+Nxk+1)}^h &= \frac{3\varPhi_{\square(k)}^{2h} + 3\varPhi_{\square(k+Nxk)}^{2h} + \varPhi_{\square(k+1)}^h + \varPhi_{\square(k+Nxk+1)}^{2h}}{8},
\end{aligned}$$

respectivamente.

Este operador não mantém o mesmo número de ondas após ser aplicado. No entanto, os modos suaves da malha grossa têm maior influência sobre o operador e, sendo assim, a malha fina passará a contar com propriedades de baixa frequência também (BRIGGS; HENSON; MCCORMICK, 2000).

### 2.3.3 Esquemas, ciclos e algoritmos

Uma vez definidos os operadores de restrição e prolongação, resta mostrar como o método *multigrid* faz o uso dessas e outras componentes algorítmicas. A ordem em que as malhas são percorridas através desses operadores chama-se ciclo *multigrid*. Os principais tipos encontrados na literatura são os ciclos V, W e F. De acordo com Hirsch (2007), o ciclo W realiza aproximadamente 50% mais operações, tornado-se mais caro computacionalmente. Já o ciclo F possui um custo computacional entre os ciclos V e W, com maior robustez, entretanto, é um algoritmo de implementação mais complexa. Por isto, neste trabalho optou-se por utilizar o ciclo V, com baixo custo computacional e fácil implementação, cuja representação pode ser vista na Fig. 9.

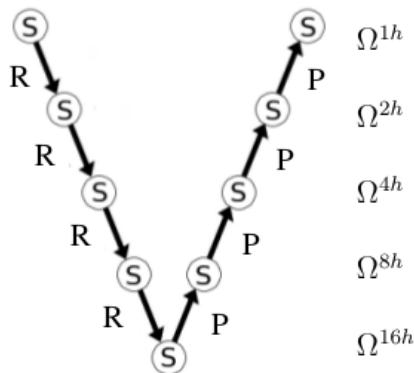


Figura 9 – Diagrama de um ciclo V aplicado em 5 malhas. S = suavização, R = restrição, P = prolongação

Existem dois esquemas que podem ser utilizados para a construção do algoritmo do método *multigrid*, o esquema de correção (*Correction Scheme*, CS) e o esquema de aproximação completa (*Full Approximation Scheme*, FAS). O primeiro transfere apenas o resíduo para a malha mais grossa, e retorna realizando as correções nas malhas finas. Este esquema é mais indicado para problemas do tipo lineares. Já o esquema FAS transfere o resíduo e a solução, resolvendo a equação  $A\mathbf{u} = \mathbf{f}$  em todas as malhas. Utiliza-se mais este esquema em problemas não lineares (BRIGGS; HENSON; MCCORMICK, 2000).

Um algoritmo do método *multigrid* é melhor representado em duas partes. A primeira é referente ao tipo de ciclo e esquema utilizados, neste caso o ciclo V e esquema CS, e pode ser vista no algoritmo 1. A segunda parte diz respeito às várias chamadas do algoritmo 1 necessárias para a execução do método, e pode ser conferida no algoritmo 2. No algoritmo 1, os valores  $\nu_1$  e  $\nu_2$  são as quantidades de suavizações a serem realizadas, respectivamente, nos processos de restrição (pré-suavização) e prolongação (pós-suavização). Os superescritos  $h$ ,  $2h$ ,  $4h$ ,..., indicam as malhas nas quais as operações estão sendo realizadas. No algoritmo 2,  $ITE_{max}$  representa o número máximo de ciclos V a realizar.

---

**Algoritmo 1: CICLO V E ESQUEMA CS PARA VÁRIOS NÍVEIS DE MALHAS**

---

CSMG( $\mathbf{v}^h, \mathbf{f}^h, \nu_1, \nu_2$ )

**início**

- 1. Suavizar  $A^h\mathbf{u}^h = \mathbf{f}^h$   $\nu_1$  vezes em  $\Omega^h$  com estimativa inicial  $\mathbf{v}_0^h$ ;
- 2. Calcular o resíduo usando  $\mathbf{r}^h = \mathbf{f}^h - A^h\mathbf{v}^h$ ;
- 3. Restringir o resíduo da malha  $\Omega^h$  para a malha  $\Omega^{2h}$ :  $\mathbf{f}^{2h} = I_{h}^{2h}\mathbf{r}^h$ ;
- a) Suavizar  $A^{2h}\mathbf{u}^{2h} = \mathbf{f}^{2h}$   $\nu_1$  vezes em  $\Omega^{2h}$  com estimativa inicial  $\mathbf{v}_0^{2h} = 0$ ;
- b) Calcular o resíduo usando  $\mathbf{r}^{2h} = \mathbf{f}^{2h} - A^{2h}\mathbf{v}^{2h}$ ;
- c) Restringir o resíduo da malha  $\Omega^{2h}$  para a malha  $\Omega^{4h}$ :  $\mathbf{f}^{4h} = I_{2h}^{4h}\mathbf{r}^{2h}$ ;
- i) Suavizar  $A^{4h}\mathbf{u}^{4h} = \mathbf{f}^{4h}$   $\nu_1$  vezes em  $\Omega^{4h}$  com estimativa inicial  $\mathbf{v}_0^{4h} = 0$ ;
- ii) Calcular o resíduo usando  $\mathbf{r}^{4h} = \mathbf{f}^{4h} - A^{4h}\mathbf{v}^{4h}$ ;
- iii) Restringir o resíduo da malha  $\Omega^{4h}$  para a malha  $\Omega^{8h}$ :  $\mathbf{f}^{8h} = I_{4h}^{8h}\mathbf{r}^{4h}$ ;
- ⋮
- Resolver  $A^{Kh}\mathbf{u}^{Kh} = \mathbf{f}^{Kh}$ ;
- ⋮
- iv) Corrigir  $\mathbf{v}^{4h} \leftarrow \mathbf{v}^{4h} + I_{8h}^{4h}\mathbf{v}^{4h}$ ;
- v) Suavizar  $A^{4h}\mathbf{u}^{4h} = \mathbf{f}^{4h}$   $\nu_2$  vezes em  $\Omega^{4h}$  com estimativa inicial  $\mathbf{v}^{4h}$ ;
- d) Corrigir  $\mathbf{v}^{2h} \leftarrow \mathbf{v}^{2h} + I_{4h}^{2h}\mathbf{v}^{4h}$ ;
- e) Suavizar  $A^{2h}\mathbf{u}^{2h} = \mathbf{f}^{2h}$   $\nu_2$  vezes em  $\Omega^{2h}$  com estimativa inicial  $\mathbf{v}^{2h}$ ;
- 4. Corrigir  $\mathbf{v}^h \leftarrow \mathbf{v}^h + I_{2h}^h\mathbf{v}^{2h}$ ;
- 5. Suavizar  $A^h\mathbf{u}^h = \mathbf{f}^h$   $\nu_2$  vezes em  $\Omega^h$  com estimativa inicial  $\mathbf{v}^h$ .

**fim**

---

Fonte: Adaptado de Briggs, Henson e McCormick (2000).

---

**Algoritmo 2: DIVERSAS CHAMADAS DO ESQUEMA CSMG**


---

**Entrada:**  $\mathbf{v}_0^h, ITE_{max}$

**início**

$i = 1$

**repita**

2. CSMG( $\mathbf{v}^h, \mathbf{f}^h, \nu_1, \nu_2$ )

3.  $\mathbf{v}_0 = \mathbf{v}$

4.  $i = i + 1$

**até** *Enquanto não atingir a convergência ou  $ITE_{max}$ , faça;*

**fim**

---

Fonte: Adaptado de Briggs, Henson e McCormick (2000).

## 2.4 MÉTODOS DE EXTRAPOLAÇÃO

Os processos iterativos utilizados na resolução de sistemas de equações podem apresentar uma convergência lenta e não desejada nas aplicações. Uma das maneiras de contornar isso é através de técnicas de extração, que se baseiam em transformações de sequências numéricas. Isto é, se uma sequência converge lentamente, é possível transformá-la em outra, desde que se mantenha o limite da original, e esta nova sequência pode convergir mais rapidamente que a primeira.

Um exemplo de transformação de uma sequência ( $S_n$ ) de números reais ou complexos que converge para  $s$ , de acordo com Brezinski e Zaglia (2013), é:

$$T_n = \frac{S_n + S_{n+1}}{2}, n = 0, 1, 2, \dots, \quad (2.13)$$

ou

$$T_n = \frac{S_n S_{n+2} + S_{n+1}^2}{S_{n+2} - 2S_{n+1} + S_n}, n = 0, 1, 2, \dots, \quad (2.14)$$

onde  $T_n$  é a nova sequência gerada, também chamada de extração.

Este trabalho apresenta sete extrações, sendo cinco do tipo escalar e dois vetoriais. Considerando que  $C_1, C_2, \dots, C_k, \dots, C_\infty$  são vetores obtidos através de um processo iterativo e  $\phi_1 = C_1(k), \phi_2 = C_2(k), \phi_i = C_i(k)$  são os escalares para todos  $k$ -ésimos componentes dos vetores  $C_i$ , então define-se o seguinte extração:

$$\phi_\infty^{Aitken} = \frac{\phi_1 \phi_3 - \phi_2^2}{\phi_3 - 2\phi_2 + \phi_1}, \quad (2.15)$$

extração de Aitken (BURDEN; FAIRES, 2008).

Existem também o Empírico, proposto em Martins e Marchi (2008),

$$\phi_\infty^{Empírico} = \phi_3 + \frac{(\phi_3 - \phi_2)^2}{2\phi_2 - \phi_3 - \phi_1}, \quad (2.16)$$

e

$$\phi_\infty^{Mitin} = \frac{\phi_1 \phi_5 - \phi_3^2}{\phi_5 - 2\phi_3 + \phi_1}, \quad (2.17)$$

que é a fórmula utilizada na extrapolação de Mitin (MITIN, 1985).

Os outros dois extrapoladores utilizados neste trabalho, Épsilon e Rho escalares, possuem uma semelhança formal entre si, porém diferem quanto as capacidades de convergência. O extrapolador Épsilon escalar é um poderoso acelerador de métodos iterativos de resolução de sistemas lineares. Porém, para sequências que convergem logaritmicamente ele falha (DELAHAYE; GERMAIN-BONNE, 1982; DELAHAYE, 2012). Por outro lado, o extrapolador Rho escalar não acelera sequências com convergência linear, mas é muito eficiente para aquelas logaritmicamente convergentes (GAO et al., 2010; GRAVES-MORRIS; ROBERTS; SALAM, 2000).

Para uma sequência  $(S_n)$ , o extrapolador Épsilon escalar é dado por (BREZINSKI, 1974):

$$\epsilon_{-1}^{(n)} = 0, \quad \epsilon_0^{(n)} = S_n, \quad n = 0, 1, \dots, \quad (2.18)$$

$$\epsilon_{k+1}^{(n)} = \epsilon_{k-1}^{(n+1)} + \frac{1}{\epsilon_k^{(n+1)} - \epsilon_k^{(n)}}, \quad n, k = 0, 1, \dots, \quad (2.19)$$

para denominadores diferentes de zero. O superescrito de  $\epsilon$  representa o elemento e, o subscrito, a iteração. De maneira análoga, o extrapolador Rho escalar é dado por (WYNN, 1956):

$$\rho_{-1}^{(n)} = 0, \quad \rho_0^{(n)} = S_n, \quad n = 0, 1, \dots, \quad (2.20)$$

$$\rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \frac{k+1}{\rho_k^{(n+1)} - \rho_k^{(n)}}, \quad n, k = 0, 1, \dots, \quad (2.21)$$

Ambos os algoritmos, Épsilon e Rho, possuem versões escalar e topológica. Enquanto a primeira preserva as características de cada componente do vetor individualmente, a segunda trata de levar em consideração as propriedades do vetor como um todo, e as duas variações são analisadas neste trabalho. O algoritmo do Épsilon topológico (TAN, 1987) pode ser visto abaixo:

$$\tilde{\epsilon}_{-1}^{(n)} = 0, \quad n = 0, 1, \dots, \quad (2.22)$$

$$\tilde{\epsilon}_0^{(n)} = S_n, \quad n = 0, 1, \dots, \quad (2.23)$$

$$\tilde{\epsilon}_{2k+1}^{(n)} = \tilde{\epsilon}_{2k-1}^{(n+1)} + \frac{y}{\langle y, \tilde{\epsilon}_{2k+1}^{(n+1)} - \tilde{\epsilon}_{2k}^{(n)} \rangle}, \quad k, n = 0, 1, \dots, \quad (2.24)$$

$$\tilde{\epsilon}_{2k+2}^{(n)} = \tilde{\epsilon}_{2k}^{(n+1)} + \frac{\tilde{\epsilon}_{2k}^{(n+1)} - \tilde{\epsilon}_{2k}^{(n)}}{\langle \tilde{\epsilon}_{2k+2}^{(n+1)} - \tilde{\epsilon}_{2k+1}^{(n)}, \tilde{\epsilon}_{2k}^{(n+1)} - \tilde{\epsilon}_{2k}^{(n)} \rangle}, \quad k, n = 0, 1, \dots, \quad (2.25)$$

no qual, como  $S_n$  são elementos de um espaço vetorial  $E$ , então  $y$  é um vetor arbitrário tal que  $y \in E^*$ , o espaço dual algébrico de  $E$  (BREZINSKI; REDIVO-ZAGLIA, 2014) e os operadores  $\langle \cdot, \cdot \rangle$  representam um produto interno. O espaço dual, de maneira sintetizada, é o

espaço de todas as transformações lineares de  $E$  em um determinado corpo sobre sua atuação (OLIVEIRA, 2001). O extrapolador Rho topológico pode ser dado por (OSADA, 1991):

$$\tilde{\rho}_{-1}^{(n)} = 0, \quad n = 0, 1, \dots, \quad (2.26)$$

$$\tilde{\rho}_0^{(n)} = S_n, \quad n = 0, 1, \dots, \quad (2.27)$$

$$\tilde{\rho}_{2k+1}^{(n)} = \tilde{\rho}_{2k-1}^{(n+1)} + \frac{(2k+1)y}{\langle y, \tilde{\rho}_{2k}^{(n+1)} - \tilde{\rho}_{2k}^{(n)} \rangle}, \quad k, n = 0, 1, \dots, \quad (2.28)$$

$$\tilde{\rho}_{2k+2}^{(n)} = \tilde{\rho}_{2k}^{(n+1)} + \frac{(2k+2)\tilde{\rho}_{2k}^{(n+2)} - \tilde{\rho}_{2k}^{(n+1)}}{\langle \tilde{\rho}_{2k}^{(n+2)} - \tilde{\rho}_{2k}^{(n+1)}, \tilde{\rho}_{2k+1}^{(n+1)} - \tilde{\rho}_{2k+1}^{(n)} \rangle}, \quad k, n = 0, 1, \dots, \quad (2.29)$$

Detalhes do extrapolador Épsilon topológico podem ser vistos em Wynn (1956) e Brezinski (1974), e do extrapolador Rho topológico em Brezinski e Zaglia (2013). Aplicações de extrapoladores podem ser vistas em Anunciação (2013), Anunciação et al. (2016).

### 3 MODELOS MATEMÁTICO E NUMÉRICO

Os modelos matemático e numérico necessários para a compreensão deste trabalho estão expostos neste capítulo.

#### 3.1 MODELO MATEMÁTICO

As equações de Navier-Stokes podem ser escritas de muitas formas, dependendo das propriedades do fluido e do escoamento (FORTUNA, 2000). Foi escolhida uma representação análoga a de Kim e Moin (1985), visando facilitar a adaptação e entendimento dos algoritmos utilizados. Portanto, será abordado um problema de escoamento laminar bidimensional de um fluido incompressível ditado por

$$\nabla \cdot \mathbf{u} = 0, \quad (3.1)$$

$$\frac{\partial u}{\partial t} + \mathbf{u} \cdot \nabla u = -\frac{\partial p}{\partial x} + \frac{1}{Re} \nabla^2 u, \quad (3.2)$$

$$\frac{\partial v}{\partial t} + \mathbf{u} \cdot \nabla v = -\frac{\partial p}{\partial y} + \frac{1}{Re} \nabla^2 v, \quad (3.3)$$

sendo  $x$  e  $y$  as coordenadas espaciais;  $t$  a coordenada temporal;  $u$  e  $v$  as componentes das coordenadas do vetor velocidade  $\mathbf{u}$  nas direções  $x$  e  $y$ ;  $p$  a pressão do fluido;  $Re$ , o número de Reynolds;  $\nabla u$ , o gradiente de  $u$ ;  $\nabla \cdot \mathbf{u}$ , o divergente de  $\mathbf{u}$  e  $\nabla^2$ , o operador laplaciano. Para complementar,

$$Re = \frac{\rho v_m l}{\mu}, \quad (3.4)$$

com  $\rho$  sendo a massa específica do fluido;  $\mu$ , sua viscosidade dinâmica;  $v_m$ , sua velocidade média e  $l$ , o diâmetro do tubo. A Eq. (3.1) é conhecida como a equação da conservação da massa (ou da continuidade), e as Eqs. (3.1) e (3.3) como as equações da conservação da quantidade de movimento linear nas direções  $x$  e  $y$  ( $QML - x$  e  $QML - y$ ), respectivamente.

Este trabalho resolve o problema de vórtice de Taylor-Green (GREEN; TAYLOR, 1937), cujo domínio é dado por

$$\{(x, y) \in \mathbb{R}^2 : -\pi \leq x \leq \pi \text{ e } -\pi \leq y \leq \pi\}. \quad (3.5)$$

As soluções analíticas são:

$$u = -(\cos x \cdot \operatorname{sen} y) e^{-2(t+Ti)}, \quad (3.6)$$

$$v = (\operatorname{sen} x \cdot \cos y) e^{-2(t+Ti)},$$

$$p = -\frac{1}{4}(\cos 2x + \operatorname{sen} 2y) e^{-4(t+Ti)},$$

em que  $Ti$  é o tempo inicial e  $t = k\Delta t$ , onde  $k$  é o passo de tempo. Neste trabalho,  $Ti = 0$ . As soluções analíticas geram as condições iniciais e de contorno do problema proposto. Por fim, as soluções da Eq. (3.6) serão utilizadas para comparar com as soluções obtidas através do método desenvolvido e o erro numérico será calculado.

### 3.2 MÉTODO DE PROJEÇÃO

Os métodos de projeção possuem as propriedades de resolver a cada passo de tempo uma sequência de equações elípticas e desacopladas para as variáveis de velocidade e pressão. No entanto, o método proposto por Chorin (1968) não apresenta convergência de primeira ordem para a pressão em relação ao tempo. Isto ocorre por alguns fatores, como não levar em consideração o gradiente da pressão em todas as etapas do método e impor condições de contorno de Neumann a essa variável (GUERMOND; MINEV; SHEN, 2006).

Alguns autores buscaram aperfeiçoar o método de projeção original a fim de melhorar a ordem de convergência. Assim, com os trabalhos de Timmermans, Minev e Vosse (1996) e Guermond e Shen (2004) foi obtido o método de correção incremental na pressão na forma rotacional, cujas velocidades convergem em segunda ordem. Já a pressão apresenta ordem igual a 3/2, via norma  $L_\infty$ . Estes fatores foram decisivos para a escolha da metodologia, que é a utilizada nesta dissertação.

O método de projeção utilizado neste trabalho está descrito em (TIMMERMANS; MINEV; VOSSE, 1996), e sua formulação encontra-se abaixo:

Primeiro passo:

$$\frac{(3\mathbf{u}^t - 4\mathbf{u}^n + \mathbf{u}^{n-1})}{2h_t} = \beta_1 g_1(\mathbf{u}^n) + \beta_0 g_0(\mathbf{u}^{n-1}) + \omega \nabla^2 \mathbf{u}^t + \nabla P^n,$$

$$\mathbf{u}^t|_{\partial\Omega} = \mathbf{b}^n, \quad (3.7)$$

em que  $\mathbf{u}^t$  é o campo de velocidades auxiliar,  $\mathbf{u}^n$  o campo de velocidades no passo de tempo  $n$ ,  $h_t$  é o refinamento temporal,  $P^n$  é a pressão no passo de tempo  $n$ ,  $\mathbf{b}^n$  são as condições de contorno no passo de tempo  $n$ ,  $\beta_0$  e  $\beta_1$  são constantes e,  $\beta_0 g_0$  e  $\beta_1 g_1$ , os termos convectivos  $\mathbf{u} \cdot \nabla \mathbf{u}$ . As escolhas de  $\beta_1$  e  $\beta_0$  implicam se os termos convectivos serão tratados explicitamente ou implicitamente. Neste trabalho,  $\beta_1 = -3/2$  e  $\beta_0 = 1/2$  (BELL; COLELLA; GLAZ, 1989).

Segundo passo:

$$\nabla^2 \phi^{n+1} = \frac{3}{2h_t} \nabla \cdot \mathbf{u}^t \quad (3.8)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^t - \frac{2h_t}{3} \nabla \phi^{n+1} \quad (3.9)$$

$$P^{n+1} = P^n + \phi^{n+1} - \omega \nabla \cdot \mathbf{u}^t \quad (3.10)$$

em que  $\phi$  é o termo de correção da pressão e  $\omega = 1/Re$ . Esta versão do método de projeção tem condições de contorno para a pressão dadas por

$$\partial_n P^{n+1}|_{\partial\Omega} = -\omega (\nabla \times \nabla \times \mathbf{u}^{n+1}) \cdot \mathbf{n}|_{\partial\Omega}, \quad (3.11)$$

com  $\mathbf{n}$  sendo o vetor normal unitário externo ao contorno e o operador  $\times$  representa produto vetorial. Essas são condições de contorno consistentes para a pressão. Guermond e Shen (2004)

mostraram que a versão incremental na forma rotacional, dada pelos passos um e dois, é de segunda ordem para as velocidades nas normas  $L_2$ ,  $L_1$  e  $L_\infty$ . Para a pressão, o esquema é de segunda ordem nas normas  $L_2$  e  $L_1$ , e de ordem 3/2 na norma  $L_\infty$ . Existem outros métodos de projeção, como o proposto em Kan (1986), que também são versões melhoradas de Chorin (1968), mas que não atingem as mesmas ordens de convergência da usada neste trabalho.

O uso de métodos de projeção têm suas vantagens quando comparados a outros métodos, tais como o método SIMPLE. As equações resultantes evitam o acoplamento nas variáveis de velocidade e são tratadas separadamente, reduzindo a complexidade do algoritmo (GRIF-FITH, 2009). Para uma generalização dos métodos de projeção, ver Guermond, Minev e Shen (2006).

### 3.3 DISCRETIZAÇÃO

Neste trabalho utiliza-se o Método de Volumes Finitos (MVF) com malhas desencontradas (*staggered grid*), desenvolvidas por Harlow e Welch (1965), também encontrado em Maliska (2004). Note na Fig. 11, que as velocidades encontram-se nas faces (equivalente às variáveis  $\Phi_\circ$  e  $\Phi_\square$  da Fig. 7) e a pressão nos centros dos volumes (equivalente à variável  $\Phi_\bullet$  da Fig. 7), evitando assim instabilidades da pressão (SHIH; TAN; HWANG, 1989).

Como exemplo da Fig. 10, os volumes com linhas tracejadas representam os volumes fictícios e não pertencem ao domínio físico do problema, porém facilitam a implementação computacional uma vez que os volumes da fronteira do domínio passam a ser tratados como os que se situam em seu interior. As condições de contorno para a variável  $u$  são prescritas nos contornos oeste e leste, e são calculadas por extrapolação linear nos contornos norte e sul. De maneira análoga, a variável  $v$  encontra-se nos contornos norte e sul, com extrapolação linear nas outras duas direções.

Então, primeiramente as Eqs. (3.1) a (3.3) são integradas em cada volume de controle (representado na Fig. 11), através do teorema da divergência de Gauss (KREYSZIG, 2010), para que então seja possível discretizar as equações.

A equação da conservação da massa, Eq. (3.1), pode ser reescrita como:

$$\oint_S \mathbf{u} \cdot \mathbf{n} dS = 0, \quad (3.12)$$

em que  $\mathbf{n}$  representa o vetor normal exterior à região de contorno  $S$ .

A Eq. (3.2), da conservação da quantidade de movimento linear na direção  $x$  (QML- $x$ ), fica:

$$\frac{\partial}{\partial t} \int_V u dV = - \oint_S u \mathbf{u} \cdot \mathbf{n} dS - \oint_S p n_x dS + \frac{1}{Re} \oint_S \nabla u \cdot \mathbf{n} dS, \quad (3.13)$$

e a equação da conservação da quantidade de movimento linear na direção  $y$  (QML- $y$ ), dada pela Eq. (3.3):

$$\frac{\partial}{\partial t} \int_V v dV = - \oint_S v \mathbf{u} \cdot \mathbf{n} dS - \oint_S p n_y dS + \frac{1}{Re} \oint_S \nabla v \cdot \mathbf{n} dS. \quad (3.14)$$

em que  $n_x$  e  $n_y$  são vetores normais às direções  $x$  e  $y$ , respectivamente.

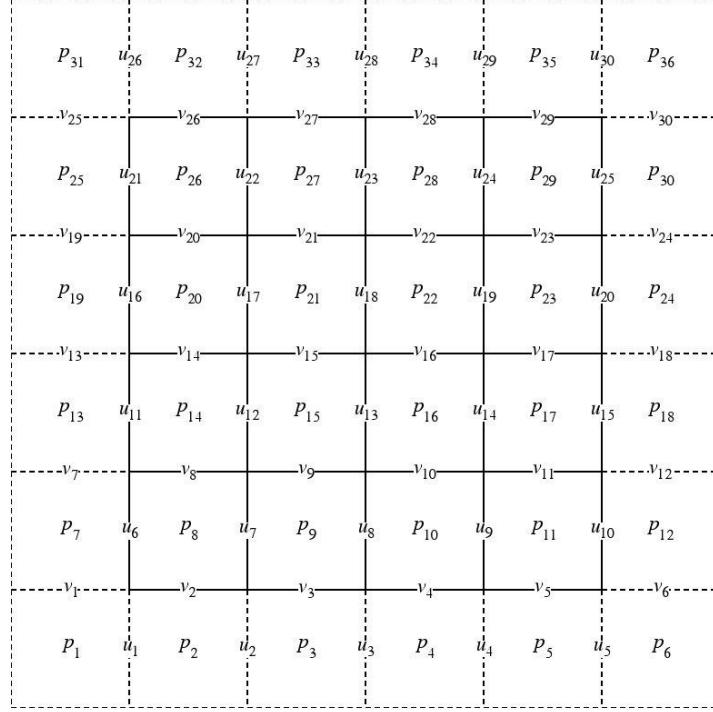


Figura 10 – Malha do tipo desencontrada

As Eqs. (3.12) a (3.14) podem ser discretizadas calculando as integrais sobre o volume de controle mostrado na Fig.11. Para simplificar, assume-se  $h_x = h_y = h$  (malhas quadrangulares). A discretização temporal depende do método de projeção utilizado e pode ser visto detalhadamente em Guermond, Minev e Shen (2006).

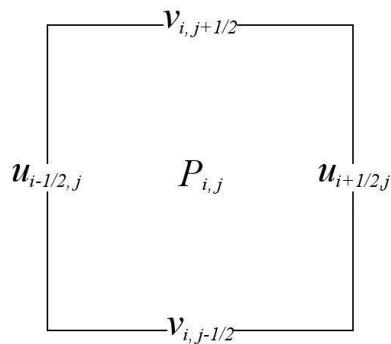


Figura 11 – Volume de controle centrado em  $P$

A Eqs. (3.12) é discretizada utilizando um volume centrado em  $P$ , conforme a Fig.11:

$$\oint_S \mathbf{u} \cdot \mathbf{n} dS \approx u_{i+1/2,j}^{n+1} - u_{i-1/2,j}^{n+1} + v_{i,j+1/2}^{n+1} - v_{i,j-1/2}^{n+1} = 0. \quad (3.15)$$

Os termos das pressões que aparecem nas Eqs (3.13) e (3.14) são discretizados utili-

zando o volume mostrado na Fig.11 da seguinte maneira:

$$\begin{aligned}\oint_S p n_x dS &\approx (p_{i+1,j} - p_{i,j})h, \\ \oint_S p n_y dS &\approx (p_{i,j+1} - p_{i,j})h.\end{aligned}\quad (3.16)$$

E os termos difusivos destas equações podem ser discretizados por

$$\oint_S \nabla u \cdot \mathbf{n} dS \approx u_{i+3/2,j}^n + u_{i-1/2,j}^n + u_{i+1/2,j+1}^n + u_{i+1/2,j-1}^n - 4u_{i+1/2,j}^n, \quad (3.17)$$

$$\oint_S \nabla v \cdot \mathbf{n} dS \approx v_{i,j+3/2}^n + v_{i,j-1/2}^n + v_{i+1,j+1/2}^n + v_{i-1,j+1/2}^n - 4v_{i,j+1/2}^n. \quad (3.18)$$

Os termos advectivos são discretizados com os volumes centrados nas componentes da velocidade:

$$\oint_S u \mathbf{u} \cdot \mathbf{n} dS \approx ((u^2)_{i+1,j}^n - (u^2)_{i,j}^n + (uv)_{i+1/2,j+1/2}^n - (uv)_{i+1/2,j-1/2}^n)h, \quad (3.19)$$

$$\oint_S v \mathbf{u} \cdot \mathbf{n} dS \approx ((v^2)_{i,j+1}^n - (v^2)_{i,j}^n + (uv)_{i+1/2,j+1/2}^n - (uv)_{i-1/2,j+1/2}^n)h, \quad (3.20)$$

Com as equações (3.19) e (3.20), exemplos de aproximações para as componentes da velocidade podem ser escritos como

$$(u^2)_{i,j}^n = \left( \frac{1}{2}(u_{i+1/2,j}^n + u_{i-1/2,j}^n) \right)^2, \quad (3.21)$$

$$(v^2)_{i,j}^n = \left( \frac{1}{2}(v_{i,j+1/2}^n + v_{i,j-1/2}^n) \right)^2, \quad (3.22)$$

$$(uv)_{i+1/2,j+1/2}^n = \left( \frac{1}{2}(u_{i+1/2,j}^n + u_{i+1/2,j+1}^n) \right) \left( \frac{1}{2}(v_{i,j+1/2}^n + v_{i+1,j+1/2}^n) \right). \quad (3.23)$$

A discretização temporal é feita de maneira semi-implícita pelo Método de Gear Extrapolado (SBDF - *Semi Backward Difference Formula*) (VILLAR, Universidade Federal de Uberlândia, Uberlândia, 2007). Através de análises feitas em Ascher, Ruuth e Wetton (1995), verificou-se que essa metodologia apresenta resultados satisfatórios quando utilizada em associação ao método *multigrid*, e isto pode ser verificado em Neundorf et al. (2016). O SBDF possui três etapas e exige um ajuste de parâmetros, dependendo do tipo de problema em que está sendo aplicado. Na primeira, as soluções são obtidas das condições iniciais e de contorno do problema. Na segunda, usa-se o Método de Euler para resolver as equações relacionadas ao primeiro avanço no tempo para, em seguida, utilizar o SBDF normalmente. Com isso, é possível obter convergência de segunda ordem na discretização temporal das variáveis  $u$  e  $v$  nas equações de Navier-Stokes. Devido à forma com que o método de projeção resolve a pressão nos contornos, ocorre uma poluição numérica que altera a ordem de convergência para esta variável, ficando reduzida a aproximadamente 1,5, conforme discutido em Guermond, Minev e Shen (2006) e Neundorf et al. (2016).

Para garantir a estabilidade das soluções, o passo de tempo  $h_t$  deve respeitar o critério (MALISKA, 2004):

$$h_t < \frac{1}{2Re} \frac{h_x^2 h_y^2}{h_x^2 + h_y^2}. \quad (3.24)$$

Em geral, os métodos de projeção fazem com que as condições de contorno sobre a pressão sejam do tipo Neumann. Tais condições são prescritas nos contornos leste e oeste para a variável  $u$  e utiliza-se extrapolação nos contornos norte e sul. De maneira análoga é feito o cálculo para  $v$ . Para garantir existência e unicidade da solução, deve-se ter (BRIGGS; HENSON; MCCORMICK, 2000):

$$\int_{\Omega} \nabla^2 \phi^{n+1} = \int_{\Omega} \frac{3}{2h_t} \nabla \cdot \mathbf{u}^t = 0. \quad (3.25)$$

Mais detalhes sobre métodos de projeção podem ser encontrados em Neundorf et al. (2016) e Guermond, Minev e Shen (2006).

## 4 RESULTADOS

Ao discretizar as equações do problema proposto através do método de volumes finitos, há uma etapa no método de projeção em que surgem sistemas lineares correspondentes ao cálculo das correções das variáveis de velocidade  $u$  e  $v$ , em cada instante de tempo  $t$ . Para a resolução destes sistemas foi utilizado o método *multigrid* associado a diferentes técnicas de extrapolação. O efeito que os extrapoladores causam nas soluções obtidas é discutido neste capítulo.

É possível dividir os experimentos em duas fases. Na primeira fase, os extrapoladores são executados após os ciclos do método *multigrid* e, na segunda, entre os ciclos. As simulações foram realizadas em diferentes tamanhos de malha e critérios de parada. É esperado que o uso dos extrapoladores aplicados nas equações de Navier-Stokes seja tão vantajoso quanto a aplicação feita nas equações de Burgers, como mostrado em Anunciação et al. (2016).

Vale ressaltar que os algoritmos implementados foram validados em outras aplicações, como em Neundorf et al. (2016) e Anunciação et al. (2016). No primeiro, problemas modelados por equações de Navier-Stokes foram resolvidos, porém sem o uso de extrapoladores. Já no segundo, foram utilizados os extrapoladores vistos neste trabalho, porém aplicados em outros tipos de problema, como as equações de Poisson e Burgers.

### 4.1 DADOS DE IMPLEMENTAÇÃO

O método *multigrid* deste trabalho está implementado com o esquema de correção e realiza ciclos V, com razão de engrossamento das malhas  $\tau = 2$  (engrossamento padrão), até que se atinja a malha mais grossa possível (BRIGGS; HENSON; MCCORMICK, 2000). O número de níveis de engrossamento será o número máximo, chamado aqui de  $L_{max}$ . Por exemplo, se  $N = 32 \times 32$ , o conjunto de malhas utilizadas será:  $N = 32 \times 32$ ,  $N = 16 \times 16$ ,  $N = 8 \times 8$ ,  $N = 4 \times 4$  e  $N = 2 \times 2$  e, portanto,  $L_{max} = 5$ . O suavizador associado é o Gauss-Seidel *Red-Black*, com números de pré e pós-suavização iguais a 3, ou seja,  $\nu = \nu_1 = \nu_2 = 3$  (OLIVEIRA, Universidade Federal do Paraná, Curitiba, 2010). O processo de restrição é realizado via ponderação completa e o de prolongação, através de interpolação bilinear. Neste trabalho foi adotado  $Re = 1$ .

A cada passo de tempo são resolvidas numericamente as velocidades e a pressão. O critério de parada utilizado no método iterativo é a norma euclidiana do resíduo adimensionada pela estimativa inicial (BRIGGS; HENSON; MCCORMICK, 2000; TROTENBERG; OOSTERLEE; SCHULLER, 2000):

$$\|\mathbf{r}\|_2 = \frac{\|\mathbf{r}^{it}\|_2}{\|\mathbf{r}^0\|_2}, \quad (4.1)$$

em que  $\mathbf{r}^{it}$  representa o resíduo da iteração atual. Ao final de cada ciclo V é calculada a equação (4.1), e ao atingir valores menores que uma tolerância  $\varepsilon$ , o processo é encerrado.

Os algoritmos foram implementados na linguagem Fortran 2003 e utilizam precisão dupla. Os testes foram realizados em um computador com processador Intel de 2.50 GHz x 4, 4GB de memória RAM e sistema operacional Windows 64-bits.

## 4.2 USO DE EXTRAPOLADORES PÓS-MULTIGRID

Nesta etapa, o método *multigrid* foi executado até que atingisse uma tolerância  $\varepsilon$  estabelecida. Em seguida, foi tomada uma dessas alternativas:

- foi realizado um novo ciclo de *multigrid*, denominado "MG + 1 ITE";
- foi aplicado um método de extrapolação, denominado MG + Extrapolador".

A Tab.3 mostra quantas aproximações são utilizadas para cada extrapolaror.

Tabela 3 – Número de soluções utilizadas para extrapolações

Extrapolaror	Número de soluções
MG + Aitken	3
MG + Empírico	3
MG + Mitin	5
MG + Épsilon escalar	5
MG + Rho escalar	5
MG + Épsilon topológico	5
MG + Rho topológico	5

As medições dos resultados foram feitas com os seguintes parâmetros: tempo de processamento ou tempo de CPU ( $t_{cpu}$ ), em segundos [s]; o pico de memória ( $M$ ) atingido durante a execução do algoritmo, em Megabytes [MB]; a norma do resíduo adimensionalizada pela estimativa inicial ( $\|\mathbf{r}\|_2$ ), dada pela Eq. (4.1); o fator de convergência média empírico ( $\hat{q}^{(k)}$ ), definido por (TROTTERBERG; OOSTERLEE; SCHULLER, 2000):

$$\hat{q}^{(k)} = \sqrt[k]{\frac{\|\mathbf{r}^{it}\|_2}{\|\mathbf{r}^0\|_2}}, \quad (4.2)$$

em que  $k$  representa a  $k$ -ésima iteração. Valores de  $\hat{q}^{(k)}$  considerados bons são próximos de zero (BRIGGS; HENSON; MCCORMICK, 2000; BURDEN; FAIRES, 2008). Por fim, também são medidas as normas infinito ( $\|E_n\|_\infty$ ) e euclidiana ( $\|E_n\|_2$ ) das variáveis  $u$ ,  $v$  e  $p$ , com  $E_n$  sendo o erro numérico no  $n$ -ésimo passo de tempo, ou seja, a diferença obtida entre a aproximação no  $n$ -ésimo passo de tempo e a solução analítica.

Para que as simulações sejam validadas em cenários distintos, foram feitas variações em relação aos tamanhos das malhas e aos critérios de parada escolhidos. Assim, os experimentos foram classificados conforme a Tab 4. Todos os casos utilizam o parâmetro passo de tempo  $t = 10$ , que é interpretado por  $10 \times ht$ , sendo que  $ht$  é calculado de maneira diferente para cada malha utilizada, respeitando o critério da Eq. (3.24).

Tabela 4 – Classificação das aplicações pós-*multigrid*

Nome da aplicação	Malha utilizada	Critério de parada ( $\epsilon$ )
A1	$256 \times 256$	$10^{-8}$
A2	$256 \times 256$	$10^{-10}$
B1	$512 \times 512$	$10^{-8}$
B2	$512 \times 512$	$10^{-10}$
C1	$1024 \times 1024$	$10^{-6}$

Através das Tabs. 5 e 6, referentes às aplicações A1 e A2, respectivamente, pode-se afirmar que apenas alguns dos extrapoladores apresentam resultados equivalentes ao do *multigrid* puro. Os extrapoladores que se destacam na Tab. 5 são: Empírico, Épsilon escalar e Épsilon topológico, devido aos valores obtidos nos quesitos tempo de CPU e erros nas variáveis  $u$ ,  $v$  e  $p$ . Vale ressaltar que ainda não é possível afirmar que os extrapoladores superam a aplicação do *multigrid* puro. O mesmo se confirma na Tab. 6, no entanto os resultados obtidos com esses mesmos extrapoladores de destaque dão indícios de que melhoram conforme é aplicado um maior rigor no critério de parada relacionado à tolerância. Até então, os demais extrapoladores não apresentaram ganhos significativos.

Tabela 5 – Resultados da aplicação A1

Metodologia	$t_{cpu}$ [s]	$M(\text{MB})$	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG + 1 ITE	12,4	27,984	9,091E-09	5,427E-07	9,094E-09	5,427E-07	5,008E-06	2,043E-04
MG + Aitken	11,1	33,508	3,099E-05	4,138E-05	1,655E-05	3,352E-05	4,064E-04	7,314E-04
MG + Empírico	11,7	33,512	1,331E-08	1,472E-06	1,328E-08	1,472E-06	5,074E-06	2,048E-04
MG + Mitin	11,5	33,516	8,524E-08	6,663E-07	1,385E-07	6,731E-07	4,798E-05	2,267E-04
MG + Épsilon escalar	11,9	33,516	9,100E-09	6,006E-07	9,101E-09	6,006E-07	5,004E-06	2,043E-04
MG + Rho escalar	12,0	33,512	1,589E-06	1,725E-04	1,589E-06	1,725E-04	2,087E-05	1,285E-03
MG + Épsilon topológico	13,2	38,132	9,101E-09	6,004E-07	9,101E-09	6,004E-07	5,004E-06	2,043E-04
MG + Rho topológico	12,2	38,132	1,590E-06	1,727E-04	1,590E-06	1,727E-04	2,081E-05	1,281E-03

Tabela 6 – Resultados da aplicação A2

Metodologia	$t_{cpu}$ [s]	$M(\text{MB})$	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG + 1 ITE	15,0	27,916	9,100E-09	6,001E-07	9,100E-09	6,001E-07	5,004E-06	2,043E-04
MG + Aitken	13,8	33,508	3,575E-03	7,457E-03	2,390E-03	6,307E-03	7,682E-01	1,303E+00
MG + Empírico	14,5	33,504	9,102E-09	6,047E-07	9,102E-09	6,047E-07	5,004E-06	2,043E-04
MG + Mitin	13,6	33,516	5,420E-05	1,155E-04	9,548E-05	1,360E-04	3,134E-02	4,646E-02
MG + Épsilon escalar	14,4	33,516	9,100E-09	6,005E-07	9,100E-09	6,005E-07	5,004E-06	2,043E-04
MG + Rho escalar	14,3	33,508	1,331E-08	1,472E-06	1,327E-08	1,472E-06	5,074E-06	2,048E-04
MG + Épsilon topológico	15,3	38,14	9,100E-09	6,004E-07	9,100E-09	6,004E-07	5,003E-06	2,043E-04
MG + Rho topológico	14,8	38,128	1,332E-08	1,472E-06	1,328E-08	1,472E-06	5,074E-06	2,048E-04

Na Tab. 7, referente à aplicação B1, o extrapolador que se destacou foi o Épsilon, segundo os critérios a seguir, tanto na versão escalar quanto na topológica. A versão escalar superou o *multigrid* puro nos seguintes parâmetros:  $t_{CPU}$ ,  $\|E_n\|_2 (u)$ ,  $\|E_n\|_2 (v)$  e  $\|E_n\|_\infty (p)$ , e a versão topológica superou em todos os quesitos, com exceção do uso de memória e do erro  $\|E_n\|_2 (p)$ . Vale ressaltar que o uso de memória nos métodos de extração sempre serão maiores que o *multigrid* puro devido ao armazenamento de soluções necessárias para o funcionamento dos métodos. O Empírico apresentou uma queda na eficiência se comparado às aplicações A1 e A2.

Tabela 7 – Resultados da aplicação B1

Metodologia	$t_{cpu}$ [s]	$M$ (MB)	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG + 1 ITE	60,2	101,464	5,982E-10	8,229E-08	5,967E-10	8,228E-08	4,493E-07	1,054E-04
MG + Aitken	52,5	116,204	7,225E-06	1,626E-05	7,434E-06	1,506E-05	8,552E-03	1,106E-02
MG + Empírico	55,8	116,200	9,733E-09	2,083E-06	9,732E-09	2,083E-06	6,924E-07	1,122E-04
MG + Mitin	51,1	116,232	2,592E-07	1,363E-06	2,642E-07	1,354E-06	5,617E-04	9,646E-04
MG + Épsilon escalar	60,2	116,240	1,529E-09	7,492E-08	1,369E-09	7,490E-08	4,459E-07	1,055E-04
MG + Rho escalar	58,3	116,220	1,746E-06	3,747E-04	1,746E-06	3,747E-04	4,086E-05	5,086E-03
MG + Épsilon topológico	59,5	134,684	5,961E-10	7,465E-08	5,961E-10	7,465E-08	4,460E-07	1,055E-04
MG + Rho topológico	58,6	134,676	1,746E-06	3,748E-04	1,746E-06	3,748E-04	4,069E-05	5,066E-03

Para o caso B1 também foram feitas as medições  $\hat{q}^{(k)}$  e  $\|\mathbf{r}\|_2$  (equações (4.2) e (4.1), respectivamente) para a variável  $v$ . Estas são medições obtidas em cada instante de tempo  $t$ , na etapa de resolução do sistema linear referente ao cálculo da correção da velocidade  $v$ , e podem ser vistas nas Figs.12 e 13.

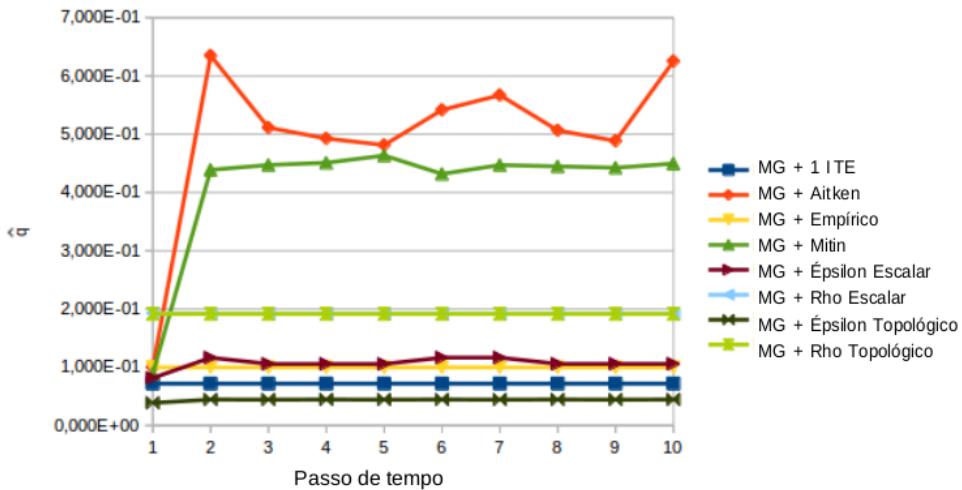
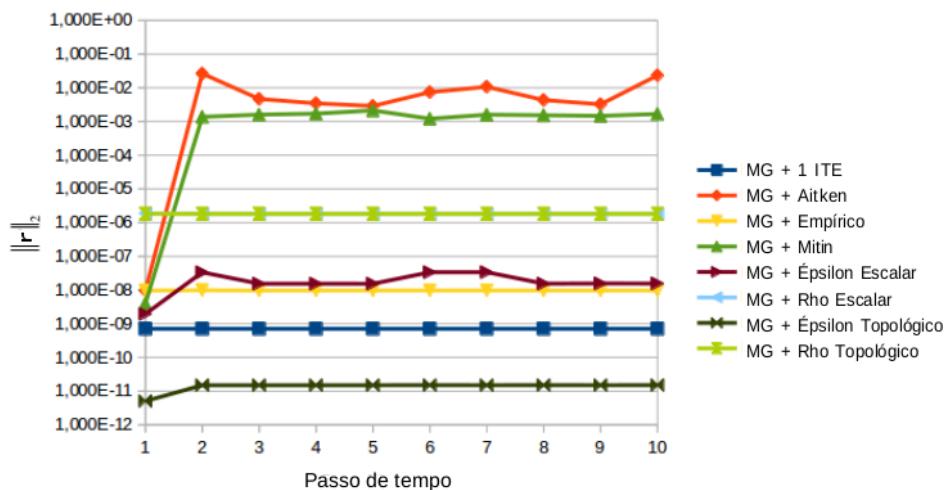


Figura 12 – Fator de convergência média empírico para cada instante de tempo

Figura 13 –  $\|\mathbf{r}\|_2$  para cada instante de tempo

Vale ressaltar que os parâmetros  $\hat{q}^{(k)}$  e  $\|\mathbf{r}\|_2$ , quando medidos para  $u$ , apresentaram resultados similares aos das Figs.12 e 13. Observa-se nessas figuras que o uso do extrapolador

Épsilon topológico reduz mais rapidamente a norma do resíduo, e consequentemente o fator de convergência média empírico, em relação ao *multigrid* puro. Pode-se afirmar também que, além do Épsilon topológico, os extrapoladores Empírico e Épsilon escalar estão entre os métodos que apresentam maior redução dos fatores avaliados.

A Tab. 8, que se refere à aplicação B2, confirma os extrapoladores que estão obtendo êxito (Épsilon escalar e Épsilon topológico). Nota-se que o tempo computacional destes é menor que do *multigrid* puro, e os demais parâmetros estão muito próximos, com exceção da memória.

Tabela 8 – Resultados da aplicação B2

Metodologia	$t_{cpu}$ [s]	$M$ (MB)	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG + 1 ITE	78,5	101,784	5,958E-10	7,385E-08	5,958E-10	7,385E-08	4,459E-07	1,055E-04
MG + Aitken	66,3	116,204	1,189E-02	2,339E-02	1,193E-02	2,100E-02	2,545E+00	7,368E+00
MG + Empírico	75,4	116,148	5,976E-10	8,476E-08	5,976E-10	8,476E-08	4,473E-07	1,055E-04
MG + Mitin	69,2	116,196	1,530E-04	4,625E-04	1,393E-04	4,567E-04	1,924E-01	3,148E-01
MG + Épsilon escalar	75,8	116,164	5,958E-10	7,463E-08	5,959E-10	7,463E-08	4,459E-07	1,055E-04
MG + Rho escalar	72,0	116,196	9,732E-09	2,082E-06	9,731E-09	2,082E-06	6,925E-07	1,122E-04
MG + Épsilon topológico	74,6	134,644	5,959E-10	7,459E-08	5,959E-10	7,459E-08	4,458E-07	1,055E-04
MG + Rho topológico	74,3	134,684	9,734E-09	2,083E-06	9,733E-09	2,083E-06	6,918E-07	1,121E-04

Como último caso analisado do tipo pós-*multigrid*, há a Tab. 9, que se refere à aplicação C1.

Tabela 9 – Resultados da aplicação C1

Metodologia	$t_{cpu}$ [s]	$M$ (MB)	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG + 1 ITE	197,5	378,868	9,933E-09	4,214E-06	9,933E-09	4,214E-06	6,171E-07	1,408E-04
MG + Aitken	181,6	424,944	1,762E-05	6,870E-05	1,244E-05	6,670E-05	5,403E-03	1,142E-02
MG + Empírico	194,6	430,34	1,622E-07	5,724E-05	1,620E-07	5,724E-05	6,567E-06	1,598E-03
MG + Mitin	181,5	430,364	5,796E-08	5,830E-07	5,796E-08	5,838E-07	6,619E-05	5,385E-04
MG + Épsilon escalar	188,7	424,952	2,671E-10	2,763E-08	2,426E-10	2,763E-08	1,848E-07	5,390E-05
MG + Rho escalar	190,8	430,384	2,496E-05	1,061E-02	2,496E-05	1,061E-02	1,018E-03	2,563E-01
MG + Épsilon topológico	196,0	504,052	7,998E-11	1,078E-08	7,997E-11	1,077E-08	1,248E-07	5,371E-05
MG + Rho topológico	192,9	504,028	2,496E-05	1,061E-02	2,496E-05	1,061E-02	1,012E-03	2,551E-01

Os resultados da Tab. 9 são os mais promissores dos casos analisados até o momento, para os extrapoladores Épsilon escalar e topológico. Eles superam o *multigrid* puro em todas as medições, exceto no uso de memória (que terá uma análise mais adequada na seção 4.4). Vale destacar a redução do erro  $\|E_n\|_2 (p)$ , que em ambos os métodos, obtiveram uma redução equivalente a uma ordem de grandeza de diferença sobre o *multigrid* puro. Estes dois métodos serão analisados na segunda etapa de resultados deste trabalho, uma vez que tenham sido submetidos a uma filtragem de seleção mostrada nesta seção.

#### 4.3 USO DE EXTRAPOLADORES ENTRE OS CICLOS

Nesta seção foram aplicados os extrapoladores Épsilon escalar e topológico, entre os ciclos *multigrid*. Isto ocorre da seguinte maneira: dada a estimativa inicial para o problema, são realizados cinco ciclos *multigrid*, o suficiente para a extração, que então é realizada. Na sequência, são realizados mais cinco ciclos *multigrid* e uma nova extração, e assim subsequentemente, até que o critério de parada seja atingido.

As simulações foram organizadas de acordo com as malhas e critérios de parada utilizados, conforme a Tab. 10. Assim como na seção 4.2, foram utilizados passos de tempo  $t = 10$  em todos os casos analisados.

Tabela 10 – Classificação das aplicações entre ciclos *multigrid*

Nome da aplicação	Malha utilizada	Critério de parada ( $\epsilon$ )
D1	$512 \times 512$	$10^{-10}$
E1	$1024 \times 1024$	$10^{-8}$
E2	$1024 \times 1024$	$10^{-10}$

A Tab. 11 representa a aplicação dos extrapoladores entre os ciclos *multigrid* ao caso D1.

Tabela 11 – Resultados da aplicação D1

Metodologia	$t_{cpu}$ [s]	$M(\text{MB})$	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG	65,1	97,568	5,976E-10	8,476E-08	5,976E-10	8,476E-08	4,473E-07	1,055E-04
MG + Épsilon topológico	57,5	130,72	5,967E-10	7,492E-08	5,966E-10	7,492E-08	4,465E-07	1,055E-04
MG + Épsilon escalar	53,3	112,264	5,974E-10	7,585E-08	5,978E-10	7,585E-08	4,471E-07	1,055E-04

Observa-se na Tab. 11 que o uso dos extrapoladores tiveram a maioria dos parâmetros iguais ou um pouco melhores que com o uso do *multigrid* puro. O Épsilon topológico apenas não superou o *multigrid* puro no quesito memória, como já era esperado. E o Épsilon escalar apresentou resultados inferiores ao *multigrid* puro nos quesitos: memória e  $\|E_n\|_\infty (v)$ . Entretanto este apresentou o menor tempo computacional.

Na Tab. 12 constam os dados referentes à aplicação E1. Neste caso, o extrapolador que teve o maior êxito foi o Épsilon escalar, superando o *multigrid* puro em todos os parâmetros, exceto em memória e normas dos erros de  $p$ , e superando também o Épsilon topológico em todos os parâmetros, exceto por uma pequena diferença em  $t_{CPU}$ . Por sua vez, o menor tempo computacional foi alcançado com o Épsilon topológico, também sendo superior ao *multigrid* puro em relação às normas  $\|E_n\|_2 (u)$  e  $\|E_n\|_2 (v)$ .

Tabela 12 – Resultados da aplicação E1

Metodologia	$t_{cpu}$ [s]	$M(\text{MB})$	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG	197,0	374,788	7,079E-10	2,999E-07	7,079E-10	2,999E-07	1,404E-07	5,384E-05
MG + Épsilon topológico	183,4	499,984	2,234E-09	2,305E-07	2,225E-09	2,305E-07	6,257E-07	8,969E-05
MG + Épsilon escalar	184,8	426,428	5,966E-10	6,380E-08	5,965E-10	6,380E-08	1,577E-07	5,529E-05

Para finalizar a análise do comportamento dos extrapoladores Épsilon escalar e topológico entre os ciclos *multigrid*, a Tab. 13 representa o caso E2.

Tabela 13 – Resultados da aplicação E2

Metodologia	$t_{cpu}$ [s]	$M(\text{MB})$	$\ E_n\ _\infty (u)$	$\ E_n\ _2 (u)$	$\ E_n\ _\infty (v)$	$\ E_n\ _2 (v)$	$\ E_n\ _\infty (p)$	$\ E_n\ _2 (p)$
MG	225,8	378,816	7,444E-11	3,171E-08	7,443E-11	3,171E-08	1,193E-07	5,376E-05
MG + Épsilon topológico	226,0	504,124	3,922E-11	9,674E-09	3,916E-11	9,675E-09	1,178E-07	5,370E-05
MG + Épsilon escalar	230,5	426,428	6,392E-11	1,146E-08	6,397E-11	1,146E-08	1,193E-07	5,371E-05

Nota-se na Tab. 13 os seguintes resultados relacionados aos extrapoladores Épsilon topológico e escalar, e o *multigrid* puro. Embora o tempo computacional dos dois métodos com extração tenha sido levemente superior ao *multigrid* puro, ambos superaram este em relação a todas as normas (com exceção de  $\|E_n\|_\infty(p)$ , obtida com o Épsilon escalar).

Considerando que o extrapolador Épsilon topológico apresentou melhores resultados na análise da Tab. 13, foram feitos testes adicionais em relação a este extrapolador mantendo os dados da aplicação E2. A Fig.14 mostra as medições  $\hat{q}^{(k)}$  e  $\|\mathbf{r}\|_2$  para a variável  $v$  a cada instante de tempo. A variável  $u$  possui resultados similares aos da Fig.14.

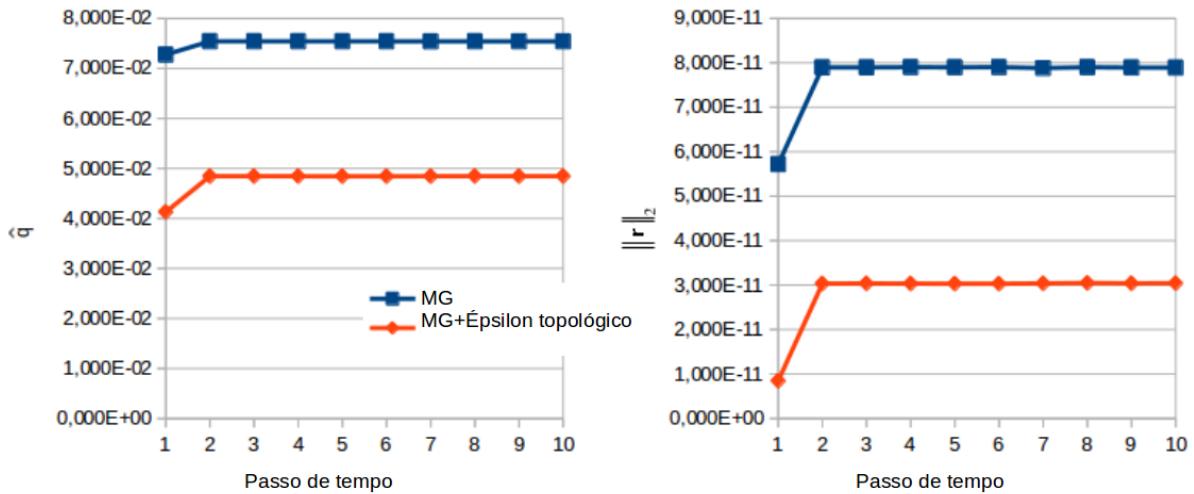


Figura 14 –  $\hat{q}^{(k)}$  e  $\|\mathbf{r}\|_2$  a cada passo de tempo

Ao aplicar o extrapolador entre os ciclos *multigrid*, a norma  $L_2$  do resíduo (em escala logarítmica) apresenta um decrescimento significativo (cerca de 100 vezes menor) comparada ao *multigrid* puro, como mostrado na Fig.15. A variável  $v$  apresenta um comportamento similar à Fig.15.

#### 4.4 TESTES ADICIONAIS

No intuito de obter resultados mais claros em relação ao tempo de CPU e uso de memória, testes adicionais foram feitos. Devido ao êxito obtido na seção 4.3, o único extrapolador analisado aqui será o Épsilon topológico.

O primeiro teste adicional diz respeito ao tempo de CPU. Para isso, foram feitas simulações com  $N$  variando de  $64 \times 64$  a  $1024 \times 1024$  e tolerância de  $10^{-8}$  para o problema proposto. Então, o parâmetro *speed-up*, dado por (GALANTE, 2006)

$$S_p = \frac{t_{CPU}(\text{algoritmo}A)}{t_{CPU}(\text{algoritmo}B)}, \quad (4.3)$$

foi calculado. Neste caso, o *algoritmo A* representa o método *multigrid* sem o extrapolador e, o *algoritmo B*, o método *multigrid* com o extrapolador. Valores de  $S_p$  maiores que a unidade

indicam que o método *multigrid* com o extrapolador é mais rápido do que o caso em que não o utiliza. Este resultado pode ser visto na Fig.16.

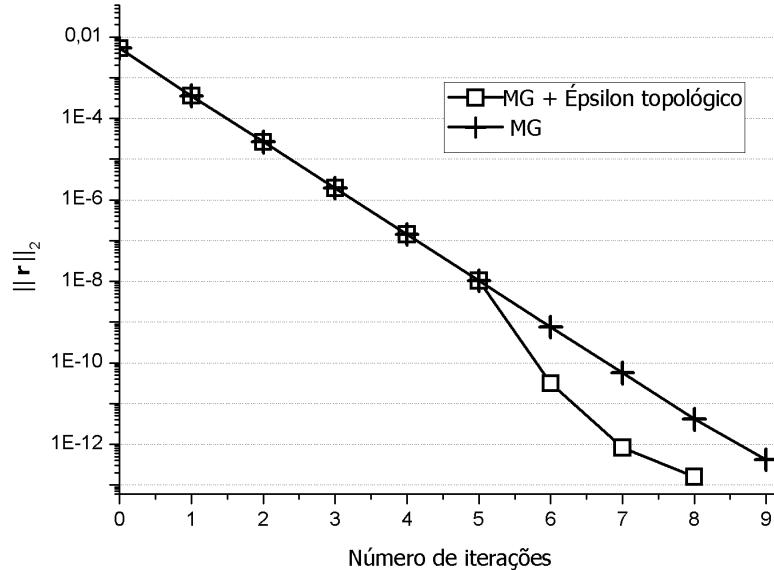


Figura 15 – Norma  $L_2$  do resíduo da variável  $u$  versus número de iterações no passo de tempo  $t = 10$

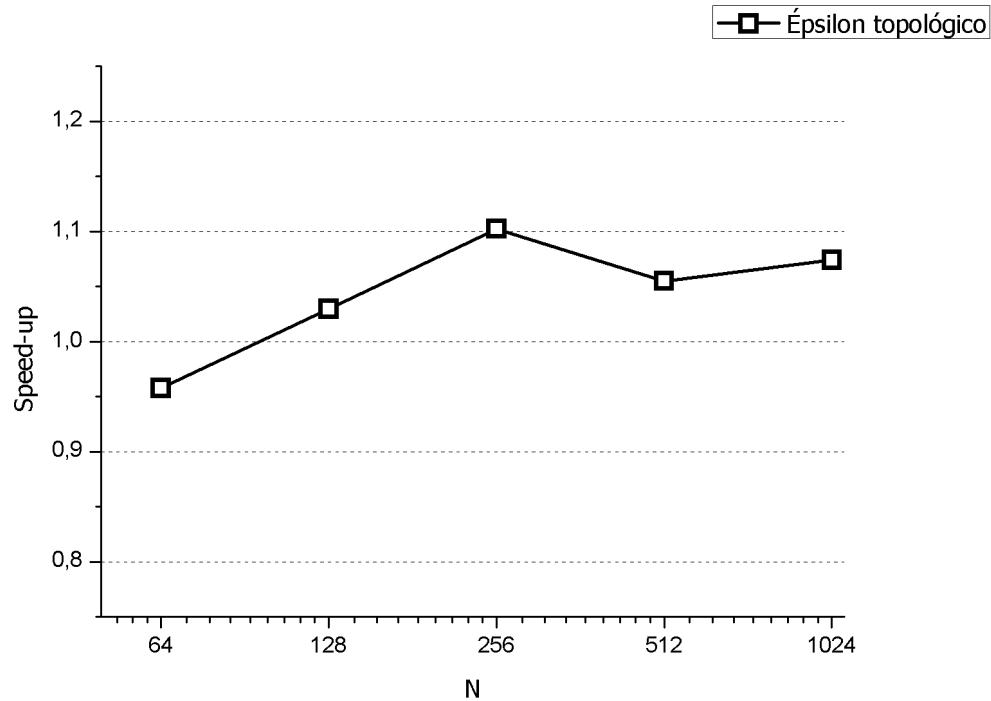


Figura 16 – O valor de *speed-up* entre o *multigrid* puro e o *multigrid* com o extrapolador Épsilon topológico

Observa-se na Fig. 16 que o uso do extrapolador Épsilon topológico faz com que o método seja mais veloz, para malhas maiores ou iguais a  $128 \times 128$ .

Como visto nas seções anteriores, o uso de extrapoladores leva a um aumento de memória RAM quando comparado ao *multigrid* puro. Para analisar este efeito, foi mensurado o aumento percentual de memória do método com extração em relação ao método sem extração, conforme aumenta-se o número de pontos na malha. Os resultados desta análise estão sintetizados na Fig. 17.

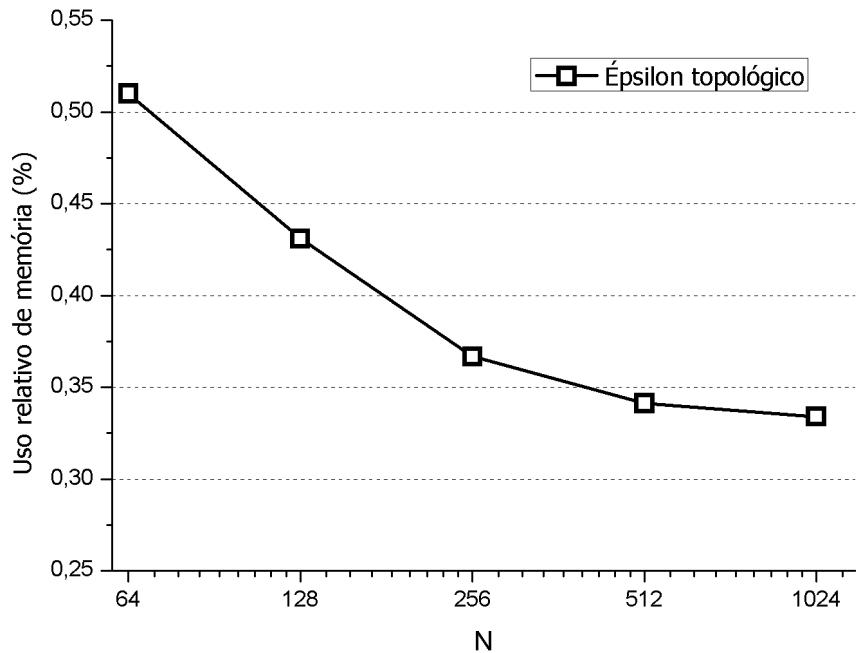


Figura 17 – Uso de memória RAM do *multigrid* com extrapolador em comparação ao *multigrid* puro

Pela Fig. 17, observa-se que o uso relativo da memória, além de diminuir, apresenta um comportamento de estabilidade assintótica conforme o refinamento das malhas. Isto significa que não há um crescimento exponencial em relação ao aumento do número de volumes no domínio, tornando as extrações viáveis.

É possível notar que o extrapolador Épsilon topológico, apesar de utilizar mais recursos de memória, possui grandes vantagens em relação ao método *multigrid* puro e aos demais extrapoladores, e isso foi mostrado com mais clareza nesta seção.

Para tratar o alto uso de memória do Épsilon Topológico, vale ressaltar que uma otimização no algoritmo pode contribuir para melhorar o desempenho, já que o esforço durante a implementação do extrapolador enfatizou principalmente a medição dos demais parâmetros envolvidos.

## 5 CONCLUSÃO

### 5.1 CONSTATAÇÕES GERAIS

Este trabalho se propôs a resolver um problema de escoamento laminar bidimensional de um fluido incompressível em regime transiente modelado pelas equações de Navier-Stokes. Elas foram resolvidas iterativamente via método de projeção, método de volumes finitos, e nos sistemas lineares que surgem da discretização foram usados o método *multigrid* e técnicas de interpolação. Os métodos de interpolação utilizados foram: Aitken, Empírico, Mitin, Épsilon escalar, Rho escalar, Épsilon topológico e Rho topológico. Os parâmetros relacionados ao *multigrid* são: uso do suavizador Gauss-Seidel *red-black*, número de pré e pós-suavização iguais a 3, restrição por ponderação completa, prolongação por interpolação bilinear, razão de engrossamento igual a 2 e ciclos do tipo V.

De acordo com os resultados da aplicação de extrapoladores pós-*multigrid*, mostrados na seção 4.2, pode-se afirmar:

1. Dentre os testados, dois foram os extrapoladores que mostraram potencial em aprimorar as soluções: Épsilon topológico e Épsilon escalar;
2. Com o refino da malha, alguns dos extrapoladores resultam em maiores ganhos. Isto é observado na Tab. 9, na qual os extrapoladores Épsilon topológico e Épsilon escalar reduziram as normas dos erros das variáveis  $u$ ,  $v$  e  $p$ , além do tempo computacional;
3. O uso de memória sempre será maior com o uso de extrapoladores em relação ao *multigrid* puro, conforme esperado;
4. O extrapolador Épsilon topológico obteve fator de convergência média empírico aproximadamente 38% menor que o obtido pelo *multigrid* puro e, em relação a norma do resíduo, aproximadamente 97% menor, ao se resolver iterativamente sistemas lineares, conforme as Figs. 12 e 13.

Já com os resultados obtidos na aplicação dos extrapoladores entre ciclos *multigrid*, da seção 4.3, e com os testes adicionais, seção 4.4 constatou-se:

1. Os extrapoladores Épsilon topológico e Épsilon escalar têm potencial para reduzir o tempo computacional, conforme visto nas Tabs. 11, 12 e na Fig. 16. Quando isto não ocorre, ou seja, o tempo computacional com o uso de extrapoladores passa a ser levemente superior ao *multigrid* puro, percebe-se que as normas do erros das variáveis de interesse são reduzidas, conforme mostra a Tab. 13. Nesta tabela, nota-se uma redução de aproximadamente 47% e 69% nas variáveis de velocidade medidas nas normas infinito e euclidiana, respectivamente, ao se usar o extrapolador Épsilon topológico em relação ao *multigrid* puro. Para o Épsilon escalar, estes valores são de 14% e 63%, respectivamente;

2. O extrapolador Épsilon topológico se sobressaiu em relação aos demais nesta aplicação para o caso da malha mais refinada ( $1024 \times 1024$ ) e o critério de parada mais rigoroso ( $\varepsilon = 10^{-10}$ ), conforme os percentuais de redução mostrados no Item 1. Sua influência na redução dos fatores de convergência média empírico e da norma do resíduo na resolução dos sistemas lineares são, respectivamente, aproximadamente 36% e 62% menores que os valores obtidos com o *multigrid* puro, conforme observado nas Fig. 14;
3. Um dos maiores desafios ao lidar com extrapolações é o uso de memória. Entretanto, ao passo que a malha do domínio é refinada, o uso de memória dos extrapoladores do tipo Épsilon em relação ao *multigrid* puro diminui e apresenta comportamento de estabilidade assintótica, conforme mostra a Fig. 17.

## 5.2 CONTRIBUIÇÕES

As contribuições que este trabalho oferece são:

1. Os efeitos de diferentes extrapoladores na resolução do problema proposto foram analisados e mensurados, possibilitando conhecer quais têm impacto positivo no método;
2. Evidenciou-se que uma das duas opções ocorrem ao utilizar o Épsilon topológico: o tempo computacional é reduzido ou as normas dos erros das variáveis de interesse ( $u$ ,  $v$  e  $p$ ) possuem maior redução, ambos os casos sendo aspectos positivos do extrapolador, considerando a complexidade do problema proposto;
3. Ao comparar com o método *multigrid* puro, constatou-se que o uso de memória do extrapolador Épsilon topológico apresenta comportamento assintótico conforme a malha é refinada, como mostrado na Fig. 17.

## 5.3 TRABALHOS FUTUROS

Os próximos trabalhos devem ser no sentido de explorar variantes das combinações das iteradas na aplicação do Épsilon Topológico, otimizando o uso de memória e abrangendo novas aplicações para o extrapolador. Também observa-se que alguns extrapoladores não foram inclusos nesta dissertação, como as recombinações de iteradas (TROTTERBERG; OOSTERLEE; SCHULLER, 2000), e que podem ser utilizados tanto em novas comparações quanto em aplicações conjuntas. Por fim, é possível explorar mais a fundo uma maneira de melhorar a resolução da variável pressão via método de projeção utilizando extrapoladores, o que traria ganhos consideráveis ao estado da arte desta metodologia.

## REFERÊNCIAS

- ALMGREN, A. S.; ASPDEN, A.; BELL, J. B.; MINION, M. L. On the use of higher-order projection methods for incompressible turbulent flow. **SIAM Journal on Scientific Computing**, SIAM, v. 35, n. 1, p. B25–B42, 2013.
- ALMGREN, A. S.; BELL, J. B.; SZYMCZAK, W. G. A numerical method for the incompressible Navier-Stokes equations based on an approximate projection. **SIAM Journal on Scientific Computing**, SIAM, v. 17, n. 2, p. 358–369, 1996.
- ANUNCIAÇÃO, M. A. M.; PINTO, M. A. V.; MARTINS, M. A.; GONÇALVES, S. d. F. T. Effect of parameters of multigrid method associated with extrapolators in CFD problems. XXXVII Iberian Latin American Congress on Computational Methods in Engineering, CILAMCE, 2016.
- ANUNCIAÇÃO, M. A. M. d. Redução do erro de iteração e aceleração do método multigrid com uso de extrapoladores. Dissertação (Mestrado) - Universidade Federal do Paraná, 2013.
- ASCHER, U. M.; RUUTH, S. J.; WETTON, B. T. Implicit-explicit methods for time-dependent partial differential equations. **SIAM Journal on Numerical Analysis**, SIAM, v. 32, n. 3, p. 797–823, 1995.
- ASTORINO, M.; GRANDMONT, C. Convergence analysis of a projection semi-implicit coupling scheme for fluid–structure interaction problems. **Numerische Mathematik**, Springer, v. 116, n. 4, p. 721–767, 2010.
- BELL, J. B.; COLELLA, P.; GLAZ, H. M. A second-order projection method for the incompressible Navier-Stokes equations. **Journal of Computational Physics**, Elsevier, v. 85, n. 2, p. 257–283, 1989.
- BRANDT, A. Multi-level adaptive solutions to boundary-value problems. **Mathematics of computation**, v. 31, n. 138, p. 333–390, 1977.
- BREZINSKI, C. **Forme confluente de l' $\varepsilon$ -algorithme topologique**. [S.l.]: Springer, 1974. v. 23. 363–370 p.
- BREZINSKI, C.; REDIVO-ZAGLIA, M. The simplified topological  $\varepsilon$ -algorithms for accelerating sequences in a vector space. **SIAM Journal on Scientific Computing**, SIAM, v. 36, n. 5, p. A2227–A2247, 2014.
- BREZINSKI, C.; ZAGLIA, M. R. **Extrapolation methods: theory and practice**. [S.l.]: Elsevier, 2013. v. 2.
- BRIGGS, W. L.; HENSON, V. E.; MCCORMICK, S. F. **A multigrid tutorial**. [S.l.]: SIAM, 2000.
- BURDEN, R.; FAIRES, J. **Análise numérica**. [S.l.: s.n.], 2008.
- CHORIN, A. J. Numerical solution of the Navier-Stokes equations. **Mathematics of computation**, v. 22, n. 104, p. 745–762, 1968.
- DELAHAYE, J.; GERMAIN-BONNE, B. The set of logarithmically convergent sequences cannot be accelerated. **SIAM Journal on Numerical Analysis**, SIAM, v. 19, n. 4, p. 840–844, 1982.

- DELAHAYE, J.-P. **Sequence transformations.** [S.l.]: Springer Science & Business Media, 2012. v. 11.
- DHATT, G.; LEFRANÃ, E.; TOUZOT, G. et al. **Finite element method.** [S.l.]: John Wiley & Sons, 2012.
- EVANS, D. The extrapolated modified Aitken iteration method for solving elliptic difference equations. **The Computer Journal**, The British Computer Society, v. 6, n. 2, p. 193–201, 1963.
- EYMARD, R.; GALLOUËT, T.; HERBIN, R. Finite volume methods. **Handbook of numerical analysis**, Elsevier, v. 7, p. 713–1018, 2000.
- FEDORENKO, R. P. The speed of convergence of one iterative process. **USSR Computational Mathematics and Mathematical Physics**, Elsevier, v. 4, n. 3, p. 227–235, 1964.
- FERZIGER, J. H.; PERIC, M. Solution of the Navier-Stokes equations. In: **Computational Methods for Fluid Dynamics.** [S.l.]: Springer, 2002. p. 157–216.
- FORTUNA, A. d. O. **Técnicas computacionais para dinâmica dos fluidos: conceitos básicos e aplicações.** [S.l.]: Edusp, 2000.
- GALANTE, G. Métodos multigrid paralelos em malhas não estruturadas aplicados à simulação de problemas de dinâmica de fluidos computacional e transferência de calor. Dissertação (Mestrado), Universidade Federal do Rio Grande do Sul, 2006.
- GAO, Q.; JIANG, Z.; LIAO, T.; SONG, K. Application of the vector  $\varepsilon$  and  $\rho$  extrapolation methods in the acceleration of the Richardson–Lucy algorithm. **Optics Communications**, Elsevier, v. 283, n. 21, p. 4224–4229, 2010.
- GHIA, U.; GHIA, K. N.; SHIN, C. High-re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. **Journal of computational physics**, Elsevier, v. 48, n. 3, p. 387–411, 1982.
- GODA, K. A multistep technique with implicit difference schemes for calculating two-or three-dimensional cavity flows. **Journal of Computational Physics**, Elsevier, v. 30, n. 1, p. 76–95, 1979.
- GONÇALVES, S. d. F. T. **Estudo de parâmetros do método multigrid geométrico para equações 2D em CFD e volumes finitos.** Tese (Doutorado), Universidade Federal do Paraná, Curitiba, 2013.
- GRAVES-MORRIS, P.; ROBERTS, D.; SALAM, A. The epsilon algorithm and related topics. **Journal of Computational and Applied Mathematics**, Elsevier, v. 122, n. 1, p. 51–80, 2000.
- GREEN, A.; TAYLOR, G. Mechanism of the production of small eddies from larger ones. v. 158, p. 499–521, 1937.
- GRIFFITH, B. E. An accurate and efficient method for the incompressible Navier-Stokes equations using the projection method as a preconditioner. **Journal of Computational Physics**, Elsevier, v. 228, n. 20, p. 7565–7595, 2009.
- GUERMOND, J.; SHEN, J. On the error estimates for the rotational pressure-correction projection methods. **Mathematics of Computation**, v. 73, n. 248, p. 1719–1737, 2004.

- GUERMOND, J.-L.; MINEV, P.; SHEN, J. An overview of projection methods for incompressible flows. **Computer methods in applied mechanics and engineering**, Elsevier, v. 195, n. 44, p. 6011–6045, 2006.
- GUERMOND, J.-L.; SALGADO, A. A splitting method for incompressible flows with variable density based on a pressure poisson equation. **Journal of Computational Physics**, Elsevier, v. 228, n. 8, p. 2834–2846, 2009.
- GUERMOND, J.-L.; SHEN, J. Velocity-correction projection methods for incompressible flows. **SIAM Journal on Numerical Analysis**, SIAM, v. 41, n. 1, p. 112–134, 2003.
- HACKBUSCH, W. **Multigrid methods and applications**, vol. 4 of **Springer series in computational mathematics**. [S.l.]: Springer-Verlag, Berlin, 1985.
- HACKBUSCH, W. **Multigrid methods and applications**. [S.l.]: Springer Science & Business Media, 2013. v. 4.
- HARLOW, F. H.; WELCH, J. E. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. **The physics of fluids**, AIP, v. 8, n. 12, p. 2182–2189, 1965.
- HIRSCH, C. **Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics**. [S.l.]: Butterworth-Heinemann, 2007.
- JBILOU, K.; SADOK, H. Vector extrapolation methods. Applications and numerical comparison. **Journal of Computational and Applied Mathematics**, Elsevier, v. 122, n. 1, p. 149–165, 2000.
- KADIOGLU, S. Y.; KLEIN, R.; MINION, M. L. A fourth-order auxiliary variable projection method for zero-mach number gas dynamics. **Journal of Computational Physics**, Elsevier, v. 227, n. 3, p. 2012–2043, 2008.
- KAN, J. V. A second-order accurate pressure-correction scheme for viscous incompressible flow. **SIAM Journal on Scientific and Statistical Computing**, SIAM, v. 7, n. 3, p. 870–891, 1986.
- KARNIADAKIS, G. E.; ISRAELI, M.; ORSZAG, S. A. High-order splitting methods for the incompressible Navier-Stokes equations. **Journal of computational physics**, Elsevier, v. 97, n. 2, p. 414–443, 1991.
- KIM, J.; MOIN, P. **Application of a fractional-step method to incompressible Navier-Stokes equations**. [S.l.]: Elsevier, 1985. v. 59. 308–323 p.
- KREYSZIG, E. **Advanced engineering mathematics**. [S.l.]: John Wiley & Sons, 2010.
- LIMA, P. M.; GRAÇA, M. M. Convergence acceleration for boundary value problems with singularities using the e-algorithm. **Journal of computational and applied mathematics**, Elsevier, v. 61, n. 2, p. 139–164, 1995.
- MALISKA, C. R. **Transferência de calor e mecânica dos fluidos computacional**. [S.l.: s.n.], 2004.
- MARCHI, C. H. **Verificação de soluções numéricas unidimensionais em dinâmica dos fluidos**. Tese (Doutorado), Universidade Federal de Santa Catarina, Florianópolis, 2001.

- MARTINS, M. A.; MARCHI, C. H. Estimate of iteration errors in computational fluid dynamics. **Numerical Heat Transfer, Part B: Fundamentals**, Taylor & Francis, v. 53, n. 3, p. 234–245, 2008.
- MESGARANI, H.; AGHAZADEH, N.; PARMOUR, P. Aitken extrapolation and epsilon algorithm for an accelerated solution of weakly singular nonlinear Volterra integral equations. **Physica Scripta**, IOP Publishing, v. 81, n. 2, p. 025006, 2010.
- MITIN, A. Linear extrapolation in an iterative method for solving systems of equations. **USSR Computational Mathematics and Mathematical Physics**, Elsevier, v. 25, n. 2, p. 1–6, 1985.
- NEUNDORF, R. L. A.; PINTO, M. A. V.; ARAKI, L. K.; CALVETTI, L. Desempenho de um algoritmo multigrid paralelo aplicado às equações de Navier-Stokes. **XXXVII Iberian Latin American Congress on Computational Methods in Engineering**, CILAMCE, 2016.
- OLIVEIRA, C. R. D. **Introdução à análise funcional**. [S.l.]: Impa, 2001.
- OLIVEIRA, F. d. **Efeito de malhas anisotrópicas bidimensionais sobre o desempenho do método multigrid geométrico**. Tese (Doutorado), Universidade Federal do Paraná, Curitiba, 2010.
- ORSZAG, S. A.; ISRAELI, M.; DEVILLE, M. O. Boundary conditions for incompressible flows. **Journal of Scientific Computing**, Springer, v. 1, n. 1, p. 75–111, 1986.
- OSADA, N. Acceleration methods for vector sequences. **Journal of Computational and Applied Mathematics**, Elsevier, v. 38, n. 1-3, p. 361–371, 1991.
- OSELEDETS, V. I. On a new way of writing the Navier-Stokes equation. the hamiltonian formalism. **Russian Mathematical Surveys**, Turpion Ltd, v. 44, n. 3, p. 210–211, 1989.
- OZISIK, N. **Finite difference methods in heat transfer**. [S.l.]: CRC press, 1994.
- PEIRÓ, J.; SHERWIN, S. Finite difference, finite element and finite volume methods for partial differential equations. In: **Handbook of materials modeling**. [S.l.]: Springer, 2005. p. 2415–2446.
- PINTO, M. A. V. **Comportamento do multigrid geométrico em problemas de transferência de calor**. Tese (Doutorado), Universidade Federal do Paraná, Curitiba, 2006.
- PINTO, M. A. V.; MARCHI, C. H. Optimum parameters of a geometric multigrid for the two-dimensional laplace's equation. In: **Proceedings of 19th International Congress of Mechanical Engineering, Brasília, Brazil**. [S.l.: s.n.], 2007.
- ROACHE, P. J. Fundamentals of computational fluid dynamics(book). **Albuquerque, NM: Hermosa Publishers, 1998.**, 1998.
- SANTIAGO, C. D.; MARCHI, C. H. Optimum parameters of a geometric multigrid for a two-dimensional problem of two-equations. In: **XIX International Congress of Mechanical Engineering, COBEM, Brasília, DF**. [S.l.: s.n.], 2007.
- SHIH, T.; TAN, C.; HWANG, B. Effects of grid staggering on numerical schemes. **International Journal for numerical methods in fluids**, Wiley Online Library, v. 9, n. 2, p. 193–212, 1989.

- SKELBOE, S. Computation of the periodic steady-state response of nonlinear networks by extrapolation methods. **IEEE Transactions on Circuits and Systems**, IEEE, v. 27, n. 3, p. 161–175, 1980.
- TAN, R. C. Computing derivatives of eigensystems by the topological  $\varepsilon$ -algorithm. **Applied numerical mathematics**, Elsevier, v. 3, n. 6, p. 539–550, 1987.
- TANNEHILL, J. C.; ANDERSON, D. A.; PLETCHER, R. H. Computational fluid dynamics and heat transfer. **Series in Computational and Physical Processes in Mechanics and Thermal Sciences**, 1997.
- TEMAM, R. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (ii). **Archive for Rational Mechanics and Analysis**, Springer, v. 33, n. 5, p. 377–385, 1969.
- TIMMERMANS, L.; MINEV, P.; VOSSE, F. V. D. An approximate projection scheme for incompressible flow using spectral elements. **International journal for numerical methods in fluids**, Wiley Online Library, v. 22, n. 7, p. 673–688, 1996.
- TROTTERBERG, U.; OOSTERLEE, C. W.; SCHULLER, A. **Multigrid**. [S.l.]: Academic press, 2000.
- VERSTEEG, H. K.; MALALASEKERA, W. **An introduction to computational fluid dynamics: the finite volume method**. [S.l.]: Pearson Education, 2007.
- VILLAR, M. M. **Análise numérica detalhada de escoamentos multifásicos bidimensionais**. Tese (Doutorado), Universidade Federal de Uberlândia, Uberlândia, 2007.
- WESSELING, P. **Introduction To Multigrid Methods**. [S.l.: s.n.], 1995.
- WYNN, P. On a procrustean technique for the numerical transformation of slowly convergent sequences and series. v. 52, n. 4, p. 663–671, 1956.
- YOUNG, D. M. **Iterative solution of large linear systems**. [S.l.]: Elsevier, 2014.
- ZHANG, Q. A fourth-order approximate projection method for the incompressible Navier-Stokes equations on locally-refined periodic domains. **Applied Numerical Mathematics**, Elsevier, v. 77, p. 16–30, 2014.
- ZIENKIEWICZ, O. C.; TAYLOR, R. L.; ZIENKIEWICZ, O. C.; TAYLOR, R. L. **The finite element method**. [S.l.]: McGraw-hill London, 1977. v. 3.