**Estudo de métodos multigrid para
solução de equações do tipo Poisson em
malhas esféricas geodésicas icosaédricas**

Marline Ilha da Silva

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO TÍTULO
DE
MESTRE EM CIÊNCIAS

Programa: Matemática Aplicada

Orientador: Prof. Dr. Saulo Rabello Maciel de Barros

Durante o desenvolvimento deste trabalho o autor recebeu auxílio financeiro do CNPq

São Paulo, dezembro de 2014

Estudo de métodos multigrid para solução de equações do tipo Poisson em malhas esféricas geodésicas icosaédricas

Esta versão da dissertação contém as correções e alterações sugeridas pela Comissão Julgadora durante a defesa da versão original do trabalho, realizada em 15/12/2014. Uma cópia da versão original está disponível no Instituto de Matemática e Estatística da Universidade de São Paulo.

Comissão Julgadora:

- Prof. Dr. Saulo Rabello Maciel de Barros (orientador) - IME-USP
- Prof. Dr. Pedro da Silva Peixoto - IME-USP
- Prof. Dra. Alice de Jesus Kozakevicius - UFSM

Agradecimentos

Agradeço ao meu orientador, professor Saulo, por ter me aceitado no mestrado e pelos seus ensinamentos durante esta trajetória.

Agradeço ao professor Pedro pela coorientação e pelo forte incentivo durante todo o mestrado.

Faço um agradecimento muito especial aos meus queridos pais, Marlene e Milton. A eles dedico todo o meu esforço pois sempre acreditaram no meu sucesso. Estiveram presentes em todos os momentos da minha caminhada me incentivando e me apoiando, principalmente nas horas de dificuldades.

Agradeço ao meu irmão Marton pelo recebimento acolhedor com um gostoso chimarrão em cada retorno para a casa.

Não poderia deixar de agradecer a Sheldon pela sua amizade e por todo o seu amor.

Também agradeço a todos meus familiares e amigos distantes que contribuíram de forma direta ou indireta para a concretização deste mestrado.

Lembro de todos meus colegas e amigos que conheci aqui em São Paulo. Todos foram importantes para a minha adaptação e para os momentos de diversão. Em especial a Juliana e ao Eric que muitas vezes fizeram com que a ausência da minha família fosse um pouco suprida.

Agradeço a todos os professores que contribuíram para minha formação até hoje, em especial a minha orientadora de iniciação científica, Alice, sem a sua orientação eu não teria chegado até aqui.

Agradeço ao CNPq pelo apoio financeiro.

Resumo

SILVA, M. I. **Estudo de métodos multigrid para solução de equações do tipo Poisson em malhas esféricas geodésicas icosaédricas**. 2014. 86 f. Dissertação (Mestrado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2014.

O objetivo deste trabalho é o estudo de métodos multigrid para a solução de equações elípticas na esfera, discretizadas em malhas esféricas geodésicas icosaédricas. Malhas esféricas geradas a partir de sólidos platônicos receberam crescente atenção ao longo da última década, por serem razoavelmente uniformes e não apresentarem concentração de pontos em torno dos pólos como as tradicionais malhas latitude-longitude. Em especial, as malhas geodésicas icosaédricas (geradas a partir de um icosaedro inscrito na esfera com suas faces projetadas na superfície) têm sido adotadas no desenvolvimento de diversos modelos atmosféricos. Nestes é comum a necessidade de resolução de equações do tipo Poisson como parte do método de integração, motivando o nosso trabalho.

Adotamos uma discretização do operador de Laplace baseada em volumes finitos. Para tal escrevemos o laplaciano como o divergente do gradiente. O divergente é discretizado com base nos fluxos nos pontos médios das arestas das células computacionais (com o auxílio do teorema da divergência de Gauss) e no uso de diferenças centradas para aproximar as derivadas nesses pontos médios. Validamos a discretização para o operador de Laplace resolvendo uma equação de Poisson através dos métodos iterativos de Jacobi e Gauss-Seidel. Estes sabidamente não são eficientes computacionalmente, devido ao grande e crescente número de iterações necessárias para atingir a convergência ao refinar a malha. Uma alternativa muito eficiente para a resolução de equações elípticas é a metodologia multigrid. Investigamos alguns métodos multigrid propostos na literatura para a solução destas equações na malha esférica geodésica icosaédrica. A partir desse estudo, utilizando também como referência a Análise Local de Fourier para a equação de Poisson em malhas hexagonais uniformes, como uma aproximação para malhas geodésicas icosaédricas, escolhemos um algoritmo multigrid para implementação. Testamos algumas opções para as componentes do esquema multigrid. Obtivemos taxas de convergência muito boas com $V(1,1)$ ciclos com relaxação por Gauss-Seidel, restrição *full weighting* e interpolação linear.

Palavras-chave: Malha icosaédrica, equação de Poisson, discretização, multigrid.

Abstract

SILVA, M. I. **Study of multigrid methods for solving Poisson-type equations in geodesic icosahedral spherical grids**. 2014. 86 f. Dissertação (Mestrado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2014.

This work is dedicated to the numerical solution of elliptic equations on the sphere, discretized on geodesic icosahedral grids. Spherical meshes generated from projections of platonic solids received considerable attention in the last decade, once they are almost isotropic and do not present a concentration of grid points around the poles, as traditional latitude-longitude grids. In particular, the geodesic icosahedral spherical grids have been adopted in the development of several atmospheric models. In these models, the necessity to solve Poisson type equations is very common, providing a motivation for our present work.

We have employed a discretization of the Laplace operator based on finite volumes. We write the Laplacian as the divergent of the gradient operator and use Gauss theorem to derive the discretization of the operator. We integrate the fluxes along the cell borders and approximate them through finite-differences. We first validated the discretization solving Poisson's equation with a simple (and very inefficient) Jacobi-Relaxation and Gauss-Seidel. We then investigated the use of multigrid type schemes for the solution of this equation. We have analysed some schemes proposed in the literature, also using an idealized Local Fourier Analysis on hexagonal (planar) grids to estimate the behaviour of the schemes on the icosahedral grids. We have implemented and tested a multigrid method, comparing the performance with different relaxation schemes and transfer operators. We have obtained a very efficient method employing V(1,1) cycles with Gauss-Seidel relaxation, and *full-weighting* and linear interpolation as transfer-operators.

Keywords: Icosahedral grid, Poisson equation, discretization, multigrid.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
1 Introdução	1
2 Malha esférica geodésica icosaédrica e discretização do problema	5
2.1 Diagrama de Voronoi e Triangularização de Delaunay	5
2.2 Malha gerada a partir do icosaédro	9
2.3 Discretização do laplaciano	11
2.4 Teste Numérico	17
3 Multigrid	21
3.1 Matriz do problema	22
3.2 Método de Jacobi	23
3.3 Método de Gauss-Seidel	25
3.4 Esquema de relaxação: Weighted Jacobi e SOR	27
3.5 Processo de correção	27
3.5.1 Interpolação	29
3.5.2 Restrição	32
3.6 Métodos multigrid propostos na literatura	34
3.6.1 Método proposto por [Heikes e Randall, 1995a]	34
3.6.2 Método proposto por [Thuburn, 1997]	36
3.6.3 Método escolhido para implementação: proposto por [Heikes <i>et al.</i> , 2013]	39
3.7 Análise Local de Fourier	41
3.7.1 Grades hexagonais e operadores	41
3.7.2 Análise de suavização	45
3.7.3 Análise em duas grades	49
3.8 Resultados numéricos	55
3.8.1 Testes com malhas otimizadas	62
4 Conclusões	65
A Definições e resolução de exemplos	67
A.1 Definições	67
A.2 Resolução dos Exemplos	67

A.2.1	Resolução Exemplo 3.7.1	67
A.2.2	Resolução Exemplo 3.7.4	68
Referências Bibliográficas		71

Lista de Figuras

2.1	Região e diagrama de Voronoi	6
2.2	Triangularização de Delaunay e diagrama de Voronoi. Pontos pretos representam vértices dos triângulos ou geradores das regiões de Voronoi. Quadrados vermelhos representam circuncentros dos triângulos ou vértices das células de Voronoi.	8
2.3	Icosaedro e icosaedro projetado na esfera.	10
2.4	Malha primal (em azul) e diagrama de Voronoi para malhas icosaédricas com 12 e 42 geradores, respectivamente.	10
2.5	Representação do stencil para a obtenção do valor do laplaciano no centro de massa P_0	12
2.6	Erro de truncamento em duas normas distintas obtido ao aproximar o laplaciano da função dada em (2.23) utilizando o método apresentado em (2.18).	18
2.7	Erro do laplaciano em uma malha com 642 pontos (superior esquerda), 2562 (superior direita), 10242 (inferior esquerda) e 40962 (inferior direita).	19
3.1	E_∞ dado pelo método de Jacobi em diferentes níveis de recursão da malha icosaédrica.	24
3.2	Erro na solução em duas normas distintas (E_2 e E_∞) obtido ao aproximar a função dada em (3.5) pelo método de Jacobi.	25
3.3	Erro na solução em duas normas distintas (E_2 e E_∞) obtido ao aproximar a função dada em (3.5) pelo método de Gauss-Seidel.	26
3.4	Representação de três níveis consecutivos de malha. $G_2 = 162$ células, $G_1 = 42$ células. $G_0 = 12$ células.	28
3.5	Relação entre três níveis consecutivos da malha icosaédrica: a malha vermelha indica o nível $q - 1$ (nível mais grosso), a malha verde o nível q e a malha azul o nível $q + 1$ (nível mais fino).	29
3.6	Representação de células não totalmente encaixadas.	29
3.7	Representação da interpolação linear. A malha verde representa o nível mais grosso q e a malha azul o nível mais fino $q + 1$. Pontos verdes: valores da função permanecem os mesmos da malha verde. Quadrados vermelhos: valores da função interpolados nesses pontos.	30
3.8	Erro associado à interpolação linear considerando a função $u(\lambda, \theta) = \text{sen}(\theta)$	32
3.9	Processo de restrição da malha G_1 para G_0	33
3.10	Stencil de interpolação do método proposto por HR.	35
3.11	Representação dos índices das células utilizadas para obter Z_k^{N-1}	37
3.12	Representação dos índices das células utilizadas para obter Z_k^q	38

3.13	Representação do V-ciclo partindo de uma malha com 642 pontos.	40
3.14	Representação do W-ciclo partindo de uma malha com 642 pontos.	40
3.15	Grade hexagonal uniforme G_h . Vetores a_1 e a_2 definidos no sistema de coordenadas obliquas. Os pontos representam os centros das células da malha G_h	42
3.16	Baixas frequências (região em branco), altas frequências (região mais escura).	43
3.17	Representação da indexação do stencil utilizado para obter a discretização de $-\nabla^2$ apresentada em (3.20). Sendo a_1 e a_2 os vetores unitários definidos no sistema de coordenadas oblíquas.	44
3.18	Representação do stencil em coordenadas oblíquas utilizado para obter o símbolo dado em (3.19).	45
3.19	Representação dos fatores de suavização μ_1^J (Jacobi) e μ_1^G (Gauss-Seidel). Em verde está a representação para o fator de suavização de Jacobi e em vermelho, para Gauss- Seidel.	48
3.20	Posições dos $\Psi^{(\ell)}$ na malha de acordo com (3.34).	50
3.21	Função trigonométrica dada em (3.49).	55
3.22	Função trigonométrica dada em (3.50).	56
3.23	Fatores de convergência para Jacobi e restrição full weighting.	57
3.24	Fatores de convergência para Gauss-Seidel e restrição full weighting.	58
3.25	Erro associado à aproximação da solução através do método multigrid para o pro- blema dado em (3.47). Foram utilizados Gauss-Seidel para o esquema de relaxação, <i>full weighting</i> como operador de restrição e V(1, 1) ciclos.	62

Lista de Tabelas

2.1	Características da malha esférica geodésica icosaédrica.	11
2.2	Erros na estimativa do laplaciano utilizando o método apresentado em (2.18) e considerando a função dada em (2.23).	18
3.1	Erros na estimativa da solução utilizando o método de Jacobi apresentado em (3.3) considerando a função dada em (3.5).	24
3.2	Erros na estimativa da solução utilizando o método de Gauss-Seidel considerando a função dada em (3.5).	26
3.3	Erros associados à interpolação linear considerando a função dada em (3.9). Primeira coluna: nível em que a função está sendo interpolada. Segunda coluna: número de pontos da malha. Terceira e quarta colunas: erros nas normas dois e infinito.	31
3.4	Valores ótimos para μ_2^J e μ_2^G usando full weighting.	53
3.5	Valores ótimos para μ_2^J e μ_2^G usando injeção.	53
3.6	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi e <i>full weighting</i>	59
3.7	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel e <i>full weighting</i>	59
3.8	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi e <i>injeção</i>	60
3.9	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel e <i>injeção</i>	60
3.10	Erros na estimativa da solução para o problema dado em (3.47) utilizando o método multigrid, considerando Gauss-Seidel, <i>full-weighting</i> e V(1,1) ciclos para $w = 1.0$. . .	61
3.11	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi, <i>full weighting</i> e a otimização SCVT.	63
3.12	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi, <i>full weighting</i> e a otimização HR95.	63
3.13	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel, <i>full weighting</i> e a otimização SCVT.	64
3.14	Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel, <i>full weighting</i> e a otimização HR95.	64

Capítulo 1

Introdução

A obtenção de soluções numéricas para equações diferenciais parciais é uma das principais tarefas em inúmeras aplicações nas mais diferentes áreas da ciência. Entre essas aplicações, podemos citar aquelas que utilizam a esfera como base para a modelagem dos fenômenos físicos envolvidos, como por exemplo, a previsão do tempo. Para obter soluções numéricas das equações que determinam problemas físicos dessa natureza, faz-se uso de métodos numéricos e, para isso, é muitas vezes necessário transformar o domínio contínuo no qual a EDP está definida em um domínio discreto formado por *malhas* ou *grades*.

Ao longo das últimas décadas, foram desenvolvidos vários métodos para solução das equações que descrevem a dinâmica de um fluido sobre a esfera. Entre esses métodos podemos citar esquemas de diferenças finitas em malhas uniformes em coordenadas esféricas (latitude e longitude), onde as relações de diferenças aproximam as derivadas que aparecem nas equações diferenciais que regem o problema. No entanto, neste tipo de malha tem-se uma concentração de pontos na proximidade de ambos os polos. Isto acarreta forte anisotropia da malha (não uniformidade), condições de CFL para estabilidade de métodos explícitos muito restritivas, além de se constituir em potencial gargalo para paralelização dos modelos (veja [Staniforth e Thuburn, 2002]).

Uma maneira alternativa para evitar este tipo de problema é o uso de malhas mais uniformes, por exemplo, malhas geradas a partir de sólidos platônicos inscritos na esfera. Entre estas, está a malha geodésica icosaédrica, proposta por [Williamson, 1968] e [Sadourny *et al.*, 1968], e que será estudada neste trabalho. Esta malha é formada a partir de um icosaedro inscrito na esfera, com suas faces projetadas na superfície. Após este processo origina-se uma malha triangular composta por 12 triângulos geodésicos a qual pode ser sucessivamente refinada acrescentando-se novos pontos nos pontos médios das arestas geodésicas e os conectando. Utilizando esta malha triangularizada, conhecida também como *malha primal*, constrói-se sua malha dual. Esta malha dual é formada a partir da união dos circuncentros dos triângulos da malha primal, definindo assim, o *diagrama de Voronoi*. O diagrama de Voronoi constitui então a malha geodésica icosaédrica, utilizada neste trabalho. Esta malha é formada por um determinado número de hexágonos, que varia de acordo com o número de triângulos da malha primal e por 12 pentágonos associados aos vértices do icosaedro.

A malha geodésica icosaédrica tem sido utilizada no estudo de soluções de Equações do tipo Poisson. Equações do tipo Poisson resultam naturalmente do uso de métodos semi-implícitos em modelos de circulação atmosférica. A solução destas equações em malhas geodésicas icosaédricas necessita de métodos eficientes para a boa performance dos modelos. Para tal, consideraremos uma discretização para o operador de Laplace sobre a malha geodésica icosaédrica baseada em volumes

finitos [Heikes e Randall, 1995a]. Este esquema é derivado considerando o operador de Laplace como o divergente do gradiente. O divergente é discretizado com base nos fluxos nos pontos médios das arestas das células computacionais, combinado com aproximações do gradiente nesses pontos médios por diferenças centradas. Para a solução do sistema linear resultante da discretização, métodos diretos não são eficientes. O método dos gradientes conjugados poderia ser utilizado, uma vez que a matriz resultante é simétrica, porém sua eficiência computacional fica aquém de outros métodos [Burden e Faires, 2010].

Métodos *multigrid* [Trottenberg *et al.*, 2001] originados nos anos 60 [Fedorenko, 1964] são comumente eficientes para solução de equações elípticas. Nestes métodos é utilizada uma hierarquia de malhas, que auxilia na solução do problema na malha mais fina. As altas frequências do erro na malha fina são reduzidas por métodos de relaxação, enquanto as frequências mais baixas são recursivamente corrigidas nas malhas mais grossas. O método de relaxação deve ter a propriedade de reduzir rapidamente os erros oscilatórios (propriedades de suavização).

Neste trabalho, vamos analisar três versões para o método multigrid propostos na literatura para a resolução da equação de Poisson na esfera. O primeiro trabalho estudado foi o pioneiro a propor um método multigrid em malhas geodésicas icosaédricas para resolver uma equação elíptica resultante de um método para as equações de águas rasas apresentado em [Heikes e Randall, 1995a]. A seguir estudamos um segundo resultado que também propôs um método multigrid para equações deste tipo [Thuburn, 1997]. Veremos que, na verdade, o método de [Thuburn, 1997] não se constitui em um método multigrid pelo fato de não utilizar as malhas grossas para corrigir os erros nas malhas finas. Mais recentemente, em [Heikes *et al.*, 2013] foi apresentada uma nova versão para o método multigrid, esta escolhida para nossa implementação, com melhor operador de transferência entre malhas.

O método multigrid proposto por [Heikes *et al.*, 2013] é conhecido como *V-ciclo*. Ele recebe este nome em virtude da trajetória que o método percorre entre os diferentes níveis de malha. Resumindo, o V-ciclo parte da discretização do problema em uma malha fina, aplica-se um método de relaxação, utiliza as malhas mais grossas para correção das frequências baixas e retorna para finalmente obter a solução na malha mais fina, completando assim, o ciclo.

A escolha das componentes de um método multigrid pode ser guiada por meio de uma Análise Local de Fourier que em muito auxilia, por exemplo, na análise de suavização da relaxação utilizada. Em nosso trabalho apresentamos a Análise Local de Fourier proposta em [Zhou e Fulton, 2009] para a equação de Poisson em malhas hexagonais uniformes no plano, que constituem modelo idealizado para as malhas icosaédricas na esfera. Com base nesta análise vamos estimar o comportamento do método com determinadas componentes de relaxação e operadores de transferência de malha, guiando a escolha também para as malhas esféricas.

Destacamos a seguir os principais objetivos deste trabalho:

- (i) Estudar a formação, a estrutura da malha esférica geodésica icosaédrica e o algoritmo gerador desta malha. Empregamos software desenvolvido por Pedro Peixoto em sua tese de doutorado [Peixoto, 2013];
- (ii) Discretizar o laplaciano e verificar os resultados numéricos envolvendo esta discretização;
- (iii) Estudar alguns métodos multigrid propostos na literatura, escolher e implementar um algoritmo para resolver equações do tipo Poisson em malhas icosaédricas. Destacamos que o

algoritmo implementado foi incorporado como um módulo no código de Pedro Peixoto;

- (iv) Prever e comparar a eficiência do multigrid através da Análise Local de Fourier proposta por [Zhou e Fulton, 2009].

Para aprofundarmos a discussão de cada uma das etapas estudadas, este trabalho está estruturado como segue:

No capítulo 2 apresentamos o diagrama de Voronoi, definimos a triangularização de Delaunay e verificamos que a malha dual a este diagrama é a malha formada pela triangularização de Delaunay, sendo assim, possível formar uma a partir da outra. Veremos que a partir desta triangularização o diagrama de Voronoi é constituído por um determinado número de hexágonos que varia de acordo com o refinamento da malha e por 12 pentágonos, formando a malha esférica geodésica icosaédrica utilizada neste trabalho. A seguir, apresentamos a discretização do laplaciano por volumes finitos e discutimos a consistência desta discretização.

No capítulo 3, primeiramente comentamos sobre o sistema obtido ao discretizar uma equação de Poisson utilizando a aproximação discreta para o laplaciano, estudada no capítulo 2. Mostramos resultados numéricos ao utilizar os métodos de Jacobi e Gauss-Seidel para resolver o sistema obtido ao discretizar a equação de Poisson na malha geodésica icosaédrica. A seguir comentamos sobre mais alguns métodos de relaxação. Definimos os operadores de transferências de informações entre as malhas: restrição e interpolação.

Para o estudo do método de interpolação, mostramos a relação existente entre os níveis das malhas icosaédricas e em seguida apresentamos o esquema de interpolação linear. Finalmente, discutimos três métodos multigrid propostos na literatura e apresentamos o método multigrid mais recente.

Para a verificação da eficiência do método implementado realizamos um estudo, baseado na Análise Local de Fourier proposta em [Zhou e Fulton, 2009]. O método é avaliado utilizando uma equação do tipo Poisson, comparando os fatores de convergência analíticos obtidos teoricamente com o fator de convergência numérico, obtido ao utilizar o multigrid considerando os métodos de Jacobi e Gauss-Seidel como esquemas de relaxação. Apresentamos os fatores de convergência ao utilizar ambos os esquemas de relaxação. Além disso, vamos abordar dois tipos de otimizações existentes na literatura (SCVT e HR95) e mostremos os fatores de convergência para o método implementado, ao utilizar estas otimizações.

No capítulo 4 apresentamos as conclusões obtidas no estudo desta dissertação e no Apêndice acrescentamos algumas definições com relação a polígonos (Apêndice A.1) e detalhamos alguns exemplos relacionados à obtenção dos fatores de convergência (Apêndice A.2).

Capítulo 2

Malha esférica geodésica icosaédrica e discretização do problema

A partir de um icosaedro inscrito na esfera, com suas faces projetadas na superfície temos a geração de uma malha triangular, a qual podemos refinar acrescentando os pontos médios das arestas dos triângulos geodésicos e os conectando [Williamson, 1968]. Esta malha triangular, também conhecida como *malha primal*, recebe o nome de *malha triangular de Delaunay* e associado a ela, podemos construir o *diagrama de Voronoi*, como veremos neste capítulo.

Na primeira seção deste capítulo definimos o diagrama de Voronoi [Voronoi, 1908] e a triangularização de Delaunay [Delaunay, 1934] no plano, e mostramos que um é dual do outro. Em seguida, expomos como o diagrama de Voronoi e a triangularização de Delaunay estão relacionados de forma semelhante na esfera [Renka, 1997]. O diagrama de Voronoi, formado a partir da triangularização de Delaunay, constitui a malha esférica geodésica icosaédrica. Os vértices dos triângulos da malha primal são considerados como os pontos geradores (nós) das células da malha icosaédrica.

Além da construção da malha esférica geodésica icosaédrica, também apresentamos uma discretização do operador de Laplace, a fim de obter sua aproximação em cada nó das células da malha [Heikes e Randall, 1995a]. Novamente, o estudo é feito inicialmente para o plano e em seguida estendido para a esfera.

Na última seção do capítulo é mostrado um teste numérico para a análise do erro de truncamento na aproximação do laplaciano de uma dada função na malha icosaédrica.

2.1 Diagrama de Voronoi e Triangularização de Delaunay

Definição 2.1.1. Seja $S \subset \mathbb{R}^2$ um conjunto com n pontos distintos no plano. Consideremos a norma euclidiana $\|\cdot\|_2$ e sejam p e q dois pontos de S . O conjunto

$$B(p, q) = \{x \in \mathbb{R}^2 : \|x - p\|_2 = \|x - q\|_2\} \quad (2.1)$$

é chamado *bissetriz* de p e q . Em outras palavras, $B(p, q)$ é a reta perpendicular ao segmento $\overline{pq}$, passando por seu ponto médio, e que divide o plano euclidiano em dois semi-planos abertos, a saber, no semi-plano

$$D(p, q) = \{x \in \mathbb{R}^2 : \|p - x\|_2 < \|q - x\|_2\}, \quad (2.2)$$

que contém o ponto p e no semi-plano $D(q, p)$ que contém q , definido de forma análoga a (2.2). Dessa maneira, o semi-plano $D(p, q)$ é o conjunto de pontos em $\mathbb{R}^2$ mais próximos a p do que q . Sendo assim, o conjunto dos pontos que estão mais próximos de p do que qualquer outro ponto de S , é dado por:

$$VR(p, S) = \bigcap_{\substack{q \in S \\ p \neq q}} D(p, q), \quad (2.3)$$

que é chamado de *região de Voronoi de p* . Note que $VR(p, S)$ é a intersecção de $n - 1$ semi-planos abertos e portanto $VR(p, S)$ é um conjunto aberto e, além disso, é uma região poligonal convexa de no máximo $n - 1$ lados (veja a Figura 2.1a que mostra $VR(p, S)$ em cinza). Diferentes regiões de Voronoi são disjuntas e, portanto, definimos o *diagrama de Voronoi* de S por

$$VD(S) = \bigcup_{\substack{p, q \in S \\ p \neq q}} \overline{VR(p, S)} \cap \overline{VR(q, S)}, \quad (2.4)$$

sendo $\overline{VR(p, S)}$ o fecho da região de Voronoi $VR(p, S)$ e $\overline{VR(q, S)}$ o fecho da região de Voronoi $VR(q, S)$. Em outras palavras, o Diagrama de Voronoi é o conjunto formado pelos pontos da fronteira em comum de duas regiões de Voronoi (veja a Figura 2.1b que mostra $VD(S)$ em vermelho). Os segmentos de reta de um diagrama de Voronoi são chamados *arestas de Voronoi* e seus pontos de extremidade são chamados *vértices de Voronoi*. Os pontos de S são frequentemente chamados de *geradores* ou *nós*.

Observação 2.1.1. Observe pela Figura 2.1 que cada ponto do plano $\mathbb{R}^2$ ou pertence a uma região de Voronoi, ou pertence a uma das arestas de Voronoi ou é um vértice de Voronoi. Observe também que podem existir regiões de Voronoi ilimitadas.

Figura 2.1: Região e diagrama de Voronoi

Existe uma grande variedade de algoritmos para a construção de diagramas de Voronoi de um conjunto de n geradores no plano. Discussões sobre alguns destes algoritmos e mais resultados sobre o diagrama de Voronoi podem ser estudados com maiores detalhes em [Aurenhammer e Klein, 2000].

Uma *triangularização* T de um polígono em $\mathbb{R}^2$ é a subdivisão deste polígono em um conjunto de triângulos cujos interiores são dois a dois disjuntos. A intersecção entre cada dois triângulos é vazia ou constituída de uma aresta ou de um vértice.

Um subconjunto $H \subset \mathbb{R}^2$ é *convexo* em $\mathbb{R}^2$ se para todo par de pontos em H , o segmento de reta que liga estes pontos está contido em H . O *fecho convexo* de $H \subset \mathbb{R}^2$ é o menor conjunto convexo que contém todos os pontos de H . Seja $S = \{p_i, i = 1, \dots, n\}$ um conjunto de n pontos distintos não alinhados em $\mathbb{R}^2$ e H o fecho convexo de S . Além disso, consideremos uma triangularização de H com conjunto de vértices S .

Definição 2.1.2. Seja $S \subset \mathbb{R}^2$ um conjunto com n pontos distintos não alinhados no plano. A *triangularização de Delaunay* de S é uma triangularização de tal forma que não exista nenhum nó no interior do circuncírculo de qualquer triângulo da triangularização.

Uma observação interessante desta triangularização é que entre todas as triangularizações possíveis de S a triangularização de Delaunay é exatamente a que maximiza o menor ângulo entre todos os ângulos de todos os triângulos. Dessa maneira, os triângulos formados por esta triangularização são mais equiláteros possíveis [Øyvind Hjelle e Dæhlen, 2006].

O diagrama de Voronoi definido anteriormente nos fornece suporte para construir de uma forma geométrica uma triangularização de Delaunay. De fato, conectamos todos os geradores de Voronoi pertencentes a regiões vizinhas por segmentos de reta, como mostrado na Figura 2.2b. Pode-se mostrar que essa construção sempre resulta em uma subdivisão do plano em uma coleção de triângulos adjacentes não sobrepostos cobrindo o fecho convexo de S . Os segmentos de reta que conectam os geradores de Voronoi associados a regiões ilimitadas de Voronoi definem o fecho convexo de S e os outros segmentos de reta definem as arestas comuns de pares de triângulos adjacentes.

Os triângulos obtidos a partir da construção acima são chamados de *triângulos de Delaunay*. Da mesma forma, as arestas que ligam os geradores de Voronoi em S são chamadas de *arestas de Delaunay*. Enquanto o diagrama de Voronoi é definido unicamente pela sua própria construção, uma triangularização de Delaunay obtida a partir do diagrama de Voronoi não é necessariamente única. O caso em que quatro ou mais pontos de S são cocirculares é o caso em que há ambiguidade. Se este não for o caso, a triangularização de Delaunay de um conjunto finito de pontos, é única, [Øyvind Hjelle e Dæhlen, 2006]. Supomos o caso em que a triangularização de Delaunay é única então cada vértice de Voronoi é a intersecção comum de exatamente três arestas de Voronoi o que segundo [Preparata e Shamos, 1985] equivale a dizer que os vértices de Voronoi são os circuncentros dos triângulos da triangularização de Delaunay. Assim, cada vértice de Voronoi pode ser associado com um triângulo da triangularização Delaunay (vide Figura 2.2b). Além disso, seja e_{ij} uma aresta de Delaunay entre dois pontos p_i e p_j de um conjunto de pontos S , e_{ij} é sempre perpendicular a uma aresta de Voronoi e esta aresta de Voronoi é a aresta compartilhada entre as regiões de Voronoi $VD(p_i)$ e $VD(p_j)$ [Øyvind Hjelle e Dæhlen, 2006].

(a) Triangularização de Delaunay, circuncírculos e circuncentros dos triângulos da triangularização.

(b) Triangularização de Delaunay e diagrama de Voronoi.

Figura 2.2: Triangularização de Delaunay e diagrama de Voronoi. Pontos pretos representam vértices dos triângulos ou geradores das regiões de Voronoi. Quadrados vermelhos representam circuncentros dos triângulos ou vértices das células de Voronoi.

Portanto podemos construir o diagrama de Voronoi a partir da triangularização de Delaunay unindo ciclicamente os circuncentros dos triângulos da triangularização (vide Figura 2.2b). O diagrama de Voronoi pode ser construído a partir da triangularização de Delaunay e vice-versa. Denominamos este processo como *dualidade* [Preparata e Shamos, 1985], em outras palavras, o diagrama de Voronoi é dual da triangularização de Delaunay e a triangularização de Delaunay é dual do diagrama de Voronoi. Dessa forma, a dualidade entre o diagrama de Voronoi e a triangularização de Delaunay pode ser apresentada como segue:

- um gerador de Voronoi corresponde a um triângulo de Delaunay,
- uma aresta de Voronoi corresponde a uma aresta de Delaunay,
- uma região de Voronoi corresponde a um vértice em uma triangularização de Delaunay.

Em [Øyvind Hjelle e Dæhlen, 2006] é apresentada uma teoria para triangularizações e são desenvolvidos algoritmos para a triangularização de Delaunay. Além disso, mais propriedades e fatos relevantes sobre o diagrama de Voronoi podem ser encontrados em [Preparata e Shamos, 1985].

A triangularização de Delaunay e o diagrama de Voronoi são usados em muitas aplicações também em $\mathbb{R}^3$ e na superfície da esfera como, por exemplo, na geração de malhas de elementos finitos. Por isso, é necessário ter suas definições e propriedades definidas neste contexto. É o que faremos a seguir.

A *esfera unitária* centrada na origem é definida por

$$S^2 = \{p \in \mathbb{R}^3 : \|p\| = 1\},$$

sendo $\| \cdot \|$ a norma euclidiana do $\mathbb{R}^3$.

Um subconjunto $H \subset S^2$ é *convexo em S^2* se para todo par de pontos p e q em H , uma geodésica que liga p e q está contida em H . A *envoltória convexa* de $H \subset S^2$ é o menor conjunto convexo que contém todos os pontos de H [Renka, 1997].

Seja $S = \{p_i, i = 1, \dots, N\}$ um conjunto de N pontos distintos em S^2 de tal forma que seus elementos não sejam cocirculares para $N \geq 3$. Os elementos de S são chamados de *nós*. Sendo H a envoltória convexa de S , definimos a *triangularização* de H em S^2 como um conjunto de triângulos geodésicos que satisfazem as seguintes propriedades:

- i) os vértices dos triângulos são nós, em S ,
- ii) nenhum triângulo contém um outro nó, exceto os seus próprios,
- iii) os interiores dos triângulos são dois a dois disjuntos, e
- iv) a união dos triângulos cobre H .

Note que a partir da definição os vértices dos triângulos coincidem com S e particionam H .

Uma *triangularização de Delaunay* para S é uma triangularização de forma que não exista nenhum nó no interior do circuncírculo de qualquer triângulo esférico.

Agora, para cada nó $p_i \in S$ definimos a *região de Voronoi* $VR(p_i, S)$ como sendo o fecho do conjunto de pontos mais próximos de p_i do que qualquer outro ponto de S^2 , ou seja,

$$VR(p_i, S) = \{p \in S^2 : d(p, p_i) \leq d(p, p_j), \forall p_j \in S, i \neq j\},$$

sendo d a distância geodésica entre dois pontos na esfera.

O conjunto das regiões de Voronoi $VR(p_i, S)$ particiona a esfera S^2 em um *diagrama de Voronoi*. Uma *aresta de Voronoi* é a intersecção de duas regiões de Voronoi e um *vértice de Voronoi* é o ponto de intersecção de três ou mais regiões de Voronoi. Portanto as arestas de Voronoi são segmentos perpendiculares às geodésicas que unem cada par de nós e os vértices de Voronoi coincidem com os circuncentros dos triângulos com vértices em S [Renka, 1997].

Tal como fizemos acima para o plano, é possível mostrar que a triangularização de Delaunay tem como dual o diagrama de Voronoi e isso fornece a base para um algoritmo que constrói um a partir do outro. Para detalhes da justificativa deste fato veja a seção 3 de [Renka, 1997].

2.2 Malha gerada a partir do icosaédro

Consideremos uma malha formada a partir do icosaedro inscrito na esfera e suas faces projetadas na superfície como mostra a Figura 2.3b. Esta malha é composta por 12 triângulos esféricos e podemos refiná-la adicionando pontos nas posições médias das arestas dos triângulos e conectando-os. Obtemos assim quatro triângulos no interior de cada um dos 12 triângulos do icosaedro como podemos observar em vermelho na Figura 2.4b. Dessa maneira, se realizamos este processo recursivamente podemos observar que dado um nível q de resolução da malha, obtemos 2^{2q} triângulos no interior de cada um dos 12 triângulos do icosaedro.

Agora seja uma triangularização de Delaunay gerada a partir de partições do icosaedro inscrito e projetado na esfera. A malha formada por esta triangularização será chamada de *malha primal*. Na Figura 2.3a é mostrado o icosaedro e na Figura 2.3b o icosaedro projetado na esfera que dá origem à malha primal com os 12 vértices.

(a) *Sólido Platônico: icosaedro.* (b) *Icosaedro inscrito na esfera e projetado na superfície.***Figura 2.3:** *Icosaedro e icosaedro projetado na esfera.*

Já que estamos trabalhando com o icosaedro, temos que o diagrama de Voronoi gerado será constituído de polígonos esféricos pentagonais ou hexagonais. Os pentágonos estão associados aos 12 pontos originais do icosaedro e os demais aos hexágonos. Por exemplo, a Figura 2.4 ilustra em azul a malha primal e em preto seu diagrama de Voronoi. Na Figura 2.4a temos a representação da malha primal com 12 pontos geradores e seu respectivo diagrama de Voronoi. Já na Figura 2.4b foi realizado um refinamento da malha triangular com 12 vértices, formando assim uma malha triangular com 42 vértices que são os geradores do diagrama de Voronoi associado a esta triangularização. Chamamos cada região de Voronoi de *célula da malha*.

(a) *Malha primal: 12 geradores.* (b) *Malha primal: 42 geradores.***Figura 2.4:** *Malha primal (em azul) e diagrama de Voronoi para malhas icosaédricas com 12 e 42 geradores, respectivamente.*

Pensando na construção do diagrama de Voronoi a partir da triangularização de Delaunay gerada por partições do icosaedro e tendo em vista as informações da seção anterior, temos que os vértices dos pentágonos e hexágonos são os circuncentros dos triângulos obtidos na triangularização de Delaunay. Dessa maneira, podemos construir as células da malha conectando cinco ou seis cir-

cuncentros dos triângulos que compartilham o mesmo nó. Ao conectar cinco circuncentros, teremos que a região de Voronoi formada será um pentágono. Por outro lado, ao conectar seis circuncentros, teremos um hexágono. Além disso, os geradores das regiões de Voronoi são os vértices dos triângulos, e assim para definir o número de nós da malha, ou número de células, utilizamos a recorrência referente ao número de pontos N_q em cada nível de resolução q . Esta recorrência é dada por

$$N_q = 10(2^{2q}) + 2, \quad q = 0, \dots, n$$

sendo n o nível de resolução mais fino.

Observemos que a malha icosaédrica será formada por 12 pentágonos que estão associados aos vértices do icosaédro e que as demais células serão hexágonos. O número de hexágonos depende do refinamento da malha triangular inicial.

Na Tabela 2.1, apresentamos algumas características desta malha para diferentes níveis de resolução. Na primeira coluna apresentamos o nível q de recursão da malha. Na segunda, o número de nós N_q (ou número de células). Na terceira e quarta colunas temos, respectivamente, o número de triângulos e arestas para cada nível de recursão de malha.

Nível q	N_q	Triângulos	Arestas
0	12	20	30
1	42	80	120
2	162	320	480
3	642	1280	1920
4	2562	5120	7680
5	10242	20480	30720
6	40962	81920	122880
7	163842	327680	491520
8	655362	1310720	1966080
9	2621442	5242880	7864320
10	10485762	20971520	31457280

Tabela 2.1: Características da malha esférica geodésica icosaédrica.

Para a geração destas malhas, bem como para manipulação e visualização gráfica empregamos software desenvolvido em FORTRAN 90 por Pedro Peixoto em sua tese de doutorado [Peixoto, 2013].

2.3 Discretização do laplaciano

A discretização do laplaciano será feita considerando o laplaciano como o divergente do gradiente e discretizando o divergente com base nos fluxos nos pontos médios das arestas das células computacionais. Para tanto, faremos uso do Teorema da Divergência e de diferenças centradas para aproximar o gradiente nesses pontos médios (vide [Heikes e Randall, 1995a]). Primeiramente, faremos a discretização do laplaciano considerando polígonos no plano (neste caso, pentágonos e hexágonos) e, em seguida, estenderemos esta análise para a esfera (neste caso, trabalharemos em malhas icosaédricas).

Seja $\vec{v} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ um campo vetorial no plano, sendo $\vec{v} = (u, v)$. O *divergente* $\nabla \cdot \vec{v}$ de $\vec{v}$ no ponto (x_0, y_0) é dado por

$$\operatorname{div}(\vec{v})(x_0, y_0) = \nabla \cdot \vec{v}(x_0, y_0) = \frac{\partial u}{\partial x}(x_0, y_0) + \frac{\partial v}{\partial y}(x_0, y_0).$$

Para calcular o divergente de um campo vetorial $\vec{v}$ a partir da fórmula descrita, são necessárias as componentes de $\vec{v}$. Essas componentes, por sua vez, dependem da escolha das coordenadas em $\mathbb{R}^2$. Estamos interessados em obter uma forma para calcular o divergente que não dependa das coordenadas de $\vec{v}$. E assim, dada uma função escalar u , obter uma forma discreta para o laplaciano

$$\nabla^2 u = \nabla \cdot (\nabla u) = \operatorname{div}(\nabla u).$$

Queremos estimar o valor do laplaciano de uma função u em P_0 , sendo P_0 o baricentro (ou centro de massa) de uma célula de Voronoi C formada a partir da triangularização de Delaunay com triângulos equiláteros no plano. Assim, C é formada pelos pontos P_i , $i = 1, \dots, n$, sendo $n = 5$ (pentágono) ou $n = 6$ (hexágono). Seja V_i o vértice de Voronoi comum das três células com centros P_0, P_i, P_{i+1} , l_i o comprimento da aresta que une os vértices V_i e V_{i+1} e L_i o comprimento do segmento que une os centros das células P_0 e P_i . A Figura 2.5 ilustra a célula de Voronoi, C , descrita anteriormente.

(a) Célula de Voronoi: pentágono.

(b) Célula de voronoi: hexágono.

Figura 2.5: Representação do stencil para a obtenção do valor do laplaciano no centro de massa P_0 .

Uma aproximação para o valor do divergente, $\operatorname{div}(\nabla u)$, no centro de massa da célula de Voronoi é dada por

$$\operatorname{div}(\nabla u)(P_0) \approx \frac{1}{A_c} \int_C \operatorname{div}(\nabla u) \, dC, \quad (2.5)$$

sendo A_c a área da célula C . Ao fazer esta aproximação do divergente no centro de massa P_0 da célula C pela média do divergente na célula, temos uma aproximação de ordem dois em relação ao diâmetro d da célula, sendo $d = \sup_{p,q \in C} \|p - q\|$, como mostrado, por exemplo, em [Peixoto e Barros, 2013].

Pelo teorema da divergência tem-se

$$\frac{1}{A_c} \int_C \operatorname{div}(\nabla u) \, dC = \frac{1}{A_c} \int_{\partial C} \nabla u \cdot \vec{n} \, d\partial C.$$

sendo $\vec{n}$ o vetor normal à fronteira, ∂C , da célula C e $d\partial C$ o elemento de linha para a fronteira da célula. Seja P_{mi} o ponto médio da aresta i da célula C . Utilizando a aproximação da integral de linha pela regra do ponto médio [Kaplan, 2002] temos a aproximação

$$\int_{\partial C} \nabla u \cdot \vec{n} \, d\partial C \approx \int_{\partial C_1} \nabla u(P_{m1}) \cdot \vec{n}_1 \, d\partial C_1 + \int_{\partial C_2} \nabla u(P_{m2}) \cdot \vec{n}_2 \, d\partial C_2 + \cdots + \int_{\partial C_n} \nabla u(P_{mn}) \cdot \vec{n}_n \, d\partial C_n,$$

sendo ∂C a fronteira da célula C , suave por partes e $\partial C_1, \partial C_2, \dots, \partial C_n$ cada parte da fronteira ∂C .

Portanto temos a seguinte aproximação

$$\operatorname{div}(\nabla u)(P_0) \approx \frac{1}{A_c} \sum_{i=1}^n \nabla u(P_{mi}) \cdot \vec{n}_i l_i, \quad (2.6)$$

sendo $\vec{n}_i$ e l_i respectivamente o vetor normal e o comprimento da aresta i de C . Novamente em [Peixoto e Barros, 2013] é mostrado que a aproximação do divergente médio em C pelo fluxo médio nos centros das células é uma aproximação pelo menos de ordem um em relação ao tamanho da célula. A partir deste fato e da ordem de aproximação do divergente em P_0 pela média do divergente em C pode-se concluir que o erro de aproximação do método de estimativa do divergente com base nos fluxos nos pontos médios tem erro de pelo menos ordem um em relação ao tamanho da célula, como mostra o seguinte teorema.

Teorema 2.3.1. [Peixoto, 2013]. Seja $\vec{v} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ um campo vetorial C^4 e C um polígono convexo em $\mathbb{R}^2$ com área satisfazendo $\alpha d^2 \leq A_c \leq d^2$, sendo d o diâmetro do polígono e $\alpha > 0$. Então existe uma constante K que independe do diâmetro d , tal que, para d pequeno, temos

$$\left| \operatorname{div}(\vec{v})(P_0) - \frac{1}{A_c} \sum_{i=1}^n \vec{v}(P_{mi}) \cdot \vec{n}_i l_i \right| \leq Kd, \quad (2.7)$$

sendo P_0 o centro de massa de C , $d = \sup_{p,q \in C} \|p - q\|$, P_{mi} o ponto médio da aresta i de C , $\vec{n}$ o vetor normal e l_i o comprimento da aresta i de C .

Definição 2.3.1. (Célula alinhada). Uma célula é alinhada se tiver lados opostos paralelos e o comprimento das arestas opostas for igual.

Observação 2.3.1. Se a célula C for alinhada temos que o erro de aproximação do método de estimativa do divergente com base nos fluxos nos pontos médios das arestas da célula tem erro de pelo menos ordem dois em relação ao tamanho da célula, como mostra o Teorema 2.3.2 a seguir.

Teorema 2.3.2. Seja $\vec{v} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ um campo vetorial C^4 e C uma célula alinhada em $\mathbb{R}^2$ com área satisfazendo $\alpha d^2 \leq A_c \leq d^2$, sendo d o diâmetro da célula. Então existe uma constante K que independe do diâmetro d , tal que,

$$\left| \operatorname{div}(\vec{v})(P_0) - \frac{1}{A_c} \sum_{i=1}^n \vec{v}(P_{mi}) \cdot \vec{n}_i l_i \right| \leq K d^2, \quad (2.8)$$

sendo P_0 o centro de massa de C , $d = \sup_{p,q \in C} \|p - q\|$, P_{mi} o ponto médio da aresta i de C , $\vec{n}_i$ o vetor normal e l_i o comprimento da aresta i de C .

Seja u_i a função u avaliada em cada ponto P_i da malha. Para obter a estimativa para o gradiente da função u na direção normal no ponto médio de cada aresta (vide equação (2.6)) diferenças centradas são utilizadas e obtemos

$$\nabla^2 u(P_0) = \operatorname{div}(\nabla u)(P_0) \approx \frac{1}{A_c} \left[\frac{l_1}{L_1}(u_1 - u_0) + \frac{l_2}{L_2}(u_2 - u_0) + \cdots + \frac{l_n}{L_n}(u_n - u_0) \right]. \quad (2.9)$$

Escrevendo de forma compacta a expressão (2.9), uma aproximação para o valor do laplaciano da função u em P_0 pode ser obtida pela seguinte fórmula:

$$\nabla^2 u(P_0) \approx \frac{1}{A_c} \sum_{i=1}^n \frac{l_i}{L_i}(u_i - u_0), \quad (2.10)$$

sendo $n = 5$ no caso de pentágonos ou $n = 6$ para hexágonos.

Agora vamos obter um método semelhante ao apresentado em (2.10) para aproximar o valor do laplaciano no centro de massa P_0 de uma célula de Voronoi da malha esférica geodésica icosaédrica, ou seja, no centro de massa de um polígono esférico pentagonal ou hexagonal. Temos que o centro de massa p^* de um polígono esférico C é um ponto de $\mathbb{R}^3$ tal que

$$p^* = \int_C p \, dC. \quad (2.11)$$

Temos que p^* como definido em (2.11) fica no interior da esfera, mas queremos definir o centro de massa sobre a célula C . Em [Du et al., 2003] é mostrado que o centro de massa esférico (ou centróide) P_0 pode ser obtido a partir de p^* por

$$P_0 = \frac{p^*}{\|p^*\|} \quad (2.12)$$

Observação 2.3.2. O centro de massa P_0 é a projeção radial (veja a Definição A.1.1 no Apêndice A.1) do centro de massa p^* sobre a esfera.

Como a célula C de Voronoi é um polígono esférico, precisamos considerar uma correspondência entre o polígono esférico e o polígono projetado radialmente no plano tangente à esfera. Esta correspondência pode ser encontrada com detalhes em [Peixoto e Barros, 2013] e [Peixoto, 2013]. Além disso, também encontra-se em [Peixoto e Barros, 2013] uma definição de campo vetorial da esfera para o plano tangente. Esta definição é obtida de tal forma que os fluxos através das arestas dos

polígonos sejam localmente preservados, e é utilizada para poder transferir a análise de discretização do divergente da esfera para o plano tangente.

Consideremos um campo vetorial $\vec{v} : S^2 \rightarrow TS^2(p)$. Assim como no plano, podemos utilizar o teorema da divergência para calcular a integral do divergente em uma região compacta C da esfera com fronteira ∂C suave por partes

$$\int_C \operatorname{div}(\vec{v}) dC = \int_{\partial C} \vec{v} \cdot \vec{n} d\partial C, \quad (2.13)$$

sendo $\vec{n}$ o vetor normal tangente à esfera. Utilizando as definições de integral de linha podemos escrever

$$\begin{aligned} \int_{\partial C} \vec{v} \cdot \vec{n} d\partial C &= \int_0^{l_1} \vec{v}(\gamma_1(t)) \cdot \vec{n}_1 \|\gamma'_1(t)\| dt + \int_0^{l_2} \vec{v}(\gamma_2(t)) \cdot \vec{n}_2 \|\gamma'_2(t)\| dt + \dots \\ &+ \int_0^{l_n} \vec{v}(\gamma_n(t)) \cdot \vec{n}_n \|\gamma'_n(t)\| dt = \sum_{i=1}^n \int_0^{l_i} \vec{v}(\gamma_i(t)) \cdot \vec{n}_i \|\gamma'_i(t)\| dt \end{aligned} \quad (2.14)$$

sendo $\gamma_i(t)$ o i -ésimo arco geodésico que pode ser parametrizado de forma que $\|\gamma'_i(t)\| = 1$, $\vec{n}_i$ o vetor normal de $\gamma_i(t)$ tangente à esfera e l_i o ângulo formado pelos pontos que definem $\gamma_i(t)$ e os pontos da esfera.

Supomos P_0 o centro de massa de uma célula C de Voronoi da malha icosaédrica, com $n = 5$ ou $n = 6$ lados. Consideremos fluxos médios tangentes a esfera e utilizando a regra do ponto médio para a integral de linha temos que uma aproximação para o $\operatorname{div}(\vec{v})$ em P_0 pode ser dada por

$$\operatorname{div}(\vec{v})(P_0) \approx \frac{1}{A_c} \sum_{i=1}^n \vec{v}(P_{mi}) \cdot \vec{n}_i l_i. \quad (2.15)$$

sendo A_c a área da célula, P_{mi} o ponto médio da aresta γ_i e l_i o comprimento da aresta γ_i . Assim como no plano é possível obter resultados semelhantes na esfera para a ordem do erro de aproximação do método (2.15) de estimativa do divergente.

Teorema 2.3.3. Seja $\vec{v} : p \in S^2 \rightarrow TS^2(p)$ um campo vetorial C^4 e C um polígono esférico em S^2 com n arestas geodésicas. Suponha que C seja tal que $\alpha d^2 \leq A_c$, onde d é o diâmetro de C e $\alpha \in (0,1)$. Então existe uma constante K que independe do diâmetro de C , tal que,

$$\left| \operatorname{div}(\vec{v})(P_0) - \frac{1}{A_c} \sum_{i=1}^n \vec{v}(P_{mi}) \cdot \vec{n}_i l_i \right| \leq Kd, \quad (2.16)$$

sendo P_0 o centro de massa de C , P_{mi} o ponto médio da aresta i de C , $\vec{n}$ o vetor normal e l_i o comprimento da aresta i de C .

Em [Peixoto e Barros, 2013] são dadas condições suficientes para obter-se ordem 2 de convergência de discretização do divergente. É mostrado que o alinhamento das células é o causador de interferência no erro de discretização do divergente. Conclui-se que o erro de aproximação do método de estimativa do divergente com base nos fluxos nos pontos médios tem erro de pelo menos ordem

dois em relação ao tamanho da célula, se esta for um polígono esférico alinhado (veja a Definição A.1.3 de polígono esférico alinhado no Apêndice A.1).

Dado um campo escalar u e seu campo gradiente $\vec{v} = \nabla u$ podemos obter uma aproximação para $\nabla^2 u = \text{div}(\nabla u)$ em P_0 utilizando a equação (2.15). Esta aproximação é dada por

$$\text{div}(\nabla u)(P_0) \approx \frac{1}{A_c} \sum_{i=1}^n \nabla u(P_{mi}) \cdot \vec{n}_i l_i, \quad (2.17)$$

sendo A_c , $\vec{n}_i$, l_i e P_{mi} como definidos em (2.15). Para aproximar $\nabla u(P_{mi})$ diferenças centradas são utilizadas e obtemos

$$\nabla^2 u(P_0) \approx \frac{1}{A_c} \sum_{i=1}^n \frac{l_i}{L_i} (u_i - u_0), \quad (2.18)$$

sendo L_i o comprimento do segmento que une dois geradores de células vizinhas que compartilham a aresta i .

Temos que a aproximação para o laplaciano de uma função escalar u em cada ponto da malha pelo método (2.18) é feita utilizando uma aproximação do gradiente no ponto médio do segmento que conecta dois geradores de células vizinhas. Pela construção do diagrama de Voronoi este ponto está na aresta compartilhada por essas duas células vizinhas mas não necessariamente ele é o ponto médio da aresta da célula. Vamos fazer uma análise no que isso interfere na consistência do método. Primeiramente supomos que x_i é um ponto ao longo da aresta i de comprimento l_i de um polígono regular no plano (pentágono ou hexágono). Escrevemos $\nabla u \cdot \vec{n}$ como uma expansão em série de Taylor ao longo da aresta e avaliamos $\nabla u(P_{mi}) \cdot \vec{n}$, sendo P_{mi} o ponto médio da aresta i .

$$\nabla u(P_{mi}) \cdot \vec{n} = \nabla u(x_i) \cdot \vec{n} + (d_i)(\nabla u(x_i) \cdot \vec{n})' + \frac{(d_i)^2}{2!}(\nabla u(x_i) \cdot \vec{n})'' + \dots \quad (2.19)$$

sendo $d_i = P_{mi} - x_i$ com $d_i = r_i l_i$ e $-1 \leq r_i \leq 1$, o que significa que a distância entre os pontos P_{mi} e x_i é uma fração do comprimento l_i da aresta da célula. Multiplicando ambos os lados da equação em (2.19) por l_i obtemos

$$l_i \nabla u(P_{mi}) \cdot \vec{n} = l_i \nabla u(x_i) \cdot \vec{n} + l_i (d_i)(\nabla u(x_i) \cdot \vec{n})' + l_i \frac{(d_i)^2}{2!}(\nabla u(x_i) \cdot \vec{n})'' + \dots \quad (2.20)$$

Substituindo (2.20) em (2.6) e usando $d_i = r_i l_i$, escrevemos

$$\nabla^2(P_0) = \frac{1}{A_c} \sum_{i=1}^n \left(\frac{\nabla u(x_i) \cdot \vec{n}}{l_i} + r_i (\nabla u(x_i) \cdot \vec{n})' + \frac{l_i (r_i)^2}{2!} (\nabla u(x_i) \cdot \vec{n})'' + \dots \right). \quad (2.21)$$

Para que o método seja consistente precisamos que r_i convirja para zero quando a malha for refinada.

Agora, consideremos a malha icosaédrica e $r_i = \frac{d_i}{l_i}$, sendo d_i a distância entre o ponto médio da aresta i da célula e o ponto médio do segmento que conecta dois pontos geradores das células que compartilham a aresta i . Seja l_i o comprimento da aresta i da célula. Em [Heikes e Randall, 1995b] é mostrado que r_i não converge para zero quando a malha icosaédrica é refinada e portanto o método não é consistente em todo ponto. Os autores propõem um método de otimização da malha (HR95) para tornar esta propriedade válida. Esta otimização ajusta os nós fazendo com que os pontos de intersecção da malha primal e do diagrama de Voronoi sejam mais próximos possíveis

do ponto médio da aresta que liga os verticés das células da malha icosaédrica. Na Seção 3.8.1, do próximo capítulo apresentaremos os fatores de convergência para o método multigrid, utilizando esta otimização.

Seja $S = \{P_0, \dots, P_N\}$ um conjunto de $N + 1$ pontos distintos da malha esférica geodésica icosaédrica. Assumimos u_k como sendo o valor da função u avaliada no ponto P_k . Estamos interessados em estimar o valor do laplaciano da u nos $N + 1$ pontos da malha utilizando (2.18). Além disso, vamos analisar no teste numérico a seguir, o comportamento do erro de truncamento quando refinamos a malha. Lembramos que a malha utilizada aqui não é otimizada segundo o método de Heikes e Randall.

2.4 Teste Numérico

Sejam u e f funções contínuas dadas em coordenadas esféricas que dependem apenas da latitude $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ e da longitude $\lambda \in [-\pi, \pi]$. Neste caso, estamos considerando o raio $r = 1$, pois vamos trabalhar na esfera unitária S^2 . A mudança de coordenadas de um sistema cartesiano para um sistema de coordenadas esféricas pode ser encontrada com detalhes em [Kaplan, 2002].

O laplaciano em coordenadas esféricas é dado por

$$\nabla^2 u(\theta, \lambda) = \frac{1}{\cos \theta} \left(\frac{1}{\cos(\theta)} \frac{\partial^2 u}{\partial^2 \lambda} + \frac{\partial}{\partial \theta} \left(\cos(\theta) \frac{\partial u}{\partial \theta} \right) \right). \quad (2.22)$$

Consideremos a função escalar, em coordenadas esféricas, dada por

$$u(\theta, \lambda) = (\cos(\theta))^3 (\sin(\lambda))^2. \quad (2.23)$$

O laplaciano para a função dada na equação (2.23) em coordenadas esféricas é dado por

$$\nabla^2 u(\theta, \lambda) = \cos(\theta) (-5(\cos(\lambda))^2 - 12(\cos(\theta))^2 (\sin(\lambda))^2 + 7). \quad (2.24)$$

Utilizando a equação (2.24) é possível obter o valor analítico, u_j , para o laplaciano em cada ponto P_j da malha. Utilizando o método apresentado na seção anterior dado pela equação (2.18) obtemos o valor aproximado $\tilde{u}_j$ para o valor do laplaciano em cada ponto P_j da malha. Vamos calcular o erro $e_j = |u_j - \tilde{u}_j|$ de estimativa do laplaciano associado a cada ponto da malha esférica geodésica icosaédrica. Os resultados para os erros serão obtidos pelas equações

$$E_2 = \sqrt{\frac{\sum_{i=1}^{N_q} (e_j)^2}{N_q}}, \quad E_\infty = \max_{i=1, \dots, N_q} |e_j|, \quad (2.25)$$

sendo N_q o número de pontos da malha relativo ao nível q de recursão.

Mostramos na Tabela 2.2 os erros, E_2 e E_∞ , associados à estimativa do laplaciano utilizando o método apresentado em (2.18) para a função dada em (2.23) para diferentes níveis de refinamento da malha icosaédrica. Na primeira coluna da tabela, mostramos o número de recursão da malha, na segunda coluna, o número de pontos da malha (nós), na terceira e quarta colunas, os erros

obtidos pelas equações dadas em (2.25) na norma dois e norma do máximo, respectivamente. Além disso, apresentamos graficamente na Figura 2.6 os valores apresentados na Tabela 2.2. No eixo x representamos o número de pontos da malha referentes aos níveis zero até o nível 10 de sua resolução. No eixo y estão os erros E_∞ (em vermelho) e E_2 em verde.

ERRO DE TRUNCAMENTO			
Nível	N_q	Erro (E_2)	Erro (E_∞)
0	12	1.20272300	1.42641139
1	42	0.40894420	1.02870954
2	162	0.12443718	0.58663747
3	642	0.03814464	0.30305542
4	2562	0.01318858	0.15276780
5	10242	0.00534010	0.11809968
6	40962	0.00242115	0.11697483
7	163842	0.00116073	0.11669350
8	655362	0.00056982	0.11662316
9	2621442	0.00028254	0.11660557
10	10485762	0.00014072	0.11660118

Tabela 2.2: Erros na estimativa do laplaciano utilizando o método apresentado em (2.18) e considerando a função dada em (2.23).

Figura 2.6: Erro de truncamento em duas normas distintas obtido ao aproximar o laplaciano da função dada em (2.23) utilizando o método apresentado em (2.18).

Observemos que os erros E_2 e E_∞ têm comportamentos diferentes na medida que a malha vai sendo refinada. Há um padrão de convergência de ordem um na aproximação do laplaciano na norma 2, mas na norma uniforme o erro cessa de cair a partir da malha de nível 5. Estes resultados estão de acordo com as observações de [Heikes e Randall, 1995b] sobre a inconsistência da discretização do laplaciano, em malhas onde os pontos médios das arestas não são alinhadas com os centros das células.

Na Figura (2.7) apresentamos o comportamento dos erros associados à estimativa do laplaciano da função dada em (2.23) para diferentes níveis da malha. Nota-se que o erro cai em quase toda a malha, restando no entanto pontos onde o mesmo fica estagnado (isto explica a convergência na norma dois, apesar de não haver convergência uniforme).

Figura 2.7: Erro do laplaciano em uma malha com 642 pontos (superior esquerda), 2562 (superior direita), 10242 (inferior esquerda) e 40962 (inferior direita).

Felizmente, como já observado empiricamente por outros autores e como veremos em nossos resultados numéricos no próximo capítulo, a solução da equação de Poisson discretizada desta forma não só é convergente mas esta convergência se mostra de segunda ordem. Tal fato, chamado de supracovergência na literatura, já foi analisado em alguns contextos, por exemplo em [Barbeiro, 2009].

Uma análise para este fato em malhas icosaédricas ainda não foi desenvolvida, constituindo-se sem dúvida em interessante assunto de pesquisa, porém além do escopo desta dissertação.

Capítulo 3

Multigrid

Queremos resolver o sistema linear $Lu = f$ obtido ao transformar o problema contínuo $\nabla^2 u = f$ em um problema discreto utilizando a malha esférica geodésica icosaédrica. Para reduzir o erro de discretização ao discretizar o laplaciano como estudado no capítulo anterior (Seção 2.3), malhas muito refinadas são necessárias. Isso gera sistemas de equações muito grandes, por exemplo a formação de um sistema de dimensão $N_q \times N_q$ considerando $q > 5$ na Tabela 2.1. A resolução destes problemas através de métodos numéricos requer métodos computacionais eficientes. O método multigrid acelera consideravelmente a resolução dos sistemas lineares envolvidos no problema. Uma técnica eficiente, usada neste método, para reduzir as fortes oscilações do resíduo

$$r = f - Lu$$

do sistema $Lu = f$ em cada malha é suavizar as oscilações por um método de relaxação (método iterativo). As primeiras iterações deste processo, geralmente, têm rápida convergência, caracterizando a presença de frequências oscilatórias de erro. Porém, após algumas iterações o processo torna-se lento, sinalizando a predominância de frequências suaves. Este é exatamente o momento no qual é recomendável transferir o problema de relaxação para uma malha mais grossa, pois frequências de erros suaves na malha fina tornam-se erros oscilatórios na malha grossa [Trottenberg *et al.*, 2001].

No método multigrid apresentado neste capítulo o sistema $Lu = f$ é resolvido apenas na malha mais fina, denotada por G_N e as malhas mais grossas, $G_{N-1}, G_{N-2}, \dots, G_0$ são utilizadas como esquema de correção. Desse modo, o método multigrid é composto pelo esquema de relaxação que tem objetivo de suavizar as fortes oscilações do resíduo na malha mais fina e pelo processo de correção que utiliza as malhas mais grossas para corrigir os erros nas malhas finas. Como várias malhas são envolvidas no processo de correção, é necessário fazer transferência de informações entre as malhas e para isso são então definidos os operadores de interpolação e restrição. O primeiro transfere informações do nível grosso para o fino e o segundo transfere informações do nível fino para o grosso.

De forma geral, o algoritmo do método multigrid pode ser descrito como:

1. Considera-se a discretização do problema na malha fina e utiliza-se um método de relaxação para obter uma primeira aproximação para a solução.
2. Após, utilizam-se as demais malhas para correção das frequências baixas (aqui é necessário utilizar os operadores de transferência entre as malhas).

3. Por fim, retorna-se à malha fina inicial, com uma nova aproximação para a solução.

Esta nova aproximação, se necessário, é utilizada para começar novamente o processo, denominado de *ciclo*.

Na primeira seção deste capítulo apresentaremos os componentes básicos para o método multigrid: esquema de relaxação e o processo de correção o qual envolve operadores de transferências entre malhas (restrição e interpolação). Posteriormente, apresentaremos três formulações propostas na literatura para o método multigrid. Em seguida detalharemos o método multigrid escolhido para nossa implementação. Além disso, faremos um estudo para verificar sua convergência baseado na Análise Local de Fourier apresentada em [Zhou e Fulton, 2009]. Esta Análise Local de Fourier nos fornece fatores de convergência esperados pelo método de relaxação. Estes fatores serão utilizados para comparar com o fator numérico obtido ao resolver uma equação do tipo Poisson discretizada na malha icosaédrica, utilizando a formulação escolhida para o método multigrid.

3.1 Matriz do problema

Utilizando a discretização para o laplaciano apresentada no capítulo anterior podemos escrever o problema contínuo $\nabla^2 u = f$ da seguinte forma

$$\sum_{i=1}^n \frac{l_i}{L_i} (u_i - u_k) = f_k A_k, \quad (3.1)$$

para todas as células k , sendo f_k o valor da função f avaliada em um ponto P_k da malha geodésica icosaédrica, A_k a área da célula k , $n = 5$ no caso de pentágonos ou $n = 6$ no caso de hexágonos e $\frac{l_i}{L_i}$ como definido em (2.18). Observemos que ao variar k em (3.1) temos a formação de um sistema de equações lineares que vamos denotar por

$$L_h u_h = f_h, \quad (3.2)$$

sendo a dimensão da matriz L_h dada de acordo com o número de nós da malha, ou seja, dado um nível q de resolução da malha, a dimensão de L_h será $N_q \times N_q$, com $N_q = 10(2^{2q}) + 2$. Por exemplo, se a malha for de nível zero então o número de nós é 12 e, portanto, a dimensão de L_h é 12×12 . Vejamos as seguintes observações com relação à matriz deste problema.

Observemos que para $v = (1, 1, \dots, 1)^T := \mathbf{1}$ temos $L_h \mathbf{1} = \mathbf{0}$ o que implica que $\mathbf{1} \in \ker(L_h)$. Eliminando uma linha e uma coluna da matriz L_h temos uma submatriz $\hat{L}_h$ fracamente diagonal dominante e irredutível e portanto não singular. Assim L_h tem posto $N_q^2 - 1$ e $\dim \ker(L_h) = 1$.

Se u_h^1 e u_h^2 são duas soluções de (3.2), temos que $u_h^1 - u_h^2 = c\mathbf{1}$, $c \in \mathbb{R}$, ou seja, quaisquer duas soluções de (3.2) diferem por uma constante.

Supomos que o sistema (3.2) tem uma solução u_h^0 . Então $L_h u_h^0 = f_h$. Seja $v \in \ker(L_h^T)$. Como $\ker(L_h^T) = \text{Im}(L_h)^\perp$, então $\langle v, w \rangle = 0$ para qualquer $w \in \text{Im}(L_h)$, o que implica em $\langle v, f_h \rangle = 0$. Já que a matriz L_h é simétrica, então $\ker(L_h^T) = \ker(L_h) = \{c\mathbf{1} : c \in \mathbb{R}\}$. Logo $\langle \mathbf{1}, f_h \rangle = 0$. Pelo Teorema 4.7.3 de [Hackbusch, 2010] a recíproca também é verdadeira, ou seja, a condição $\langle \mathbf{1}, f_h \rangle = 0$, chamada de *condição de compatibilidade*, implica que o sistema em questão possui solução.

Garantindo que é possível encontrar uma solução para o sistema dado em (3.2), vamos resolvê-lo utilizando os métodos iterativos de Jacobi e Gauss-Seidel.

3.2 Método de Jacobi

O método de Jacobi consiste em resolver a k -ésima equação do sistema para a k -ésima incógnita usando as aproximações das demais incógnitas, obtidas no passo anterior. Matematicamente, para resolver o sistema $L_h u_h = f_h$ partindo de uma condição inicial $\tilde{u}_k^0$ o processo é definido por

$$\tilde{u}_k^{j+1} = \left(\sum_{i=1}^n \frac{l_i}{L_i} \tilde{u}_i^j - f_k A_k \right) \left(\frac{1}{\sum_{i=1}^n \frac{l_i}{L_i}} \right), \quad (3.3)$$

para todas as células k , sendo $j+1$ o índice de representação para a nova aproximação do valor da solução e j para a aproximação antiga da solução.

Consideremos o seguinte problema para o qual a condição de compatibilidade é válida

$$\nabla^2 u(\theta, \lambda) = f(\lambda, \theta), \quad (3.4)$$

sendo

$$\begin{aligned} f(\lambda, \theta) &= -\cos(\lambda) \cos^2(\theta) [\cos^2(\theta) - 4 \sin(\theta) \cos(\theta) \sin(\theta) \cos(\theta) \\ &\quad - 12 \cos^2(\theta) + 16 \cos^2(\theta) \cos^2(\theta)] / \cos^2(\theta). \end{aligned}$$

Temos que a solução analítica para (3.4) é dada por

$$u(\lambda, \theta) = \cos(\lambda) \cos^4(\theta). \quad (3.5)$$

Vamos resolver o problema (3.4) numericamente utilizando o método de Jacobi formulado em (3.3). Calculamos o erro, nas normas dois e infinito, utilizando as seguintes equações:

$$E_2 = \sqrt{\frac{\sum_{i=1}^{N_q} (e_i)^2}{N_q}}, \quad E_\infty = \max_{i=1, \dots, N_q} |e_i|, \quad (3.6)$$

sendo e_i a diferença entre o valor exato da função dada em (3.5) calculado em cada ponto da malha e o valor aproximado no mesmo ponto.

Primeiramente vamos analisar o comportamento de E_∞ quando aumentamos o número de iterações considerando diferentes níveis de resolução da malha geodésica icosaédrica.

Vejamos na Figura 3.1 o comportamento de E_∞ na medida em que aumentamos o número de iterações do método considerando as malhas dos níveis zero com $N_0 = 12$ pontos até a malha do nível seis com $N_6 = 40962$ pontos. Observemos que, para todos estes níveis da malha, a partir de um certo número de iterações o erro se mantém constante e isso significa que a partir deste número de iterações a solução aproximada será essencialmente a mesma, já tendo ocorrido a convergência dentro deste nível de precisão. No eixo das ordenadas representamos o número de iterações e no eixo das abscissas, o erro E_∞ para os diferentes níveis de recursão da malha.

Figura 3.1: E_∞ dado pelo método de Jacobi em diferentes níveis de recursão da malha icosaédrica.

Como a partir de um certo número de iterações não faz diferença iterar mais, o número de iterações foi escolhido com base neste fato, ou seja, interrompemos as iterações quando a variação do erro é menor que 10^{-5} . Os erros E_2 e E_∞ serão apresentados na Tabela 3.1 para diferentes níveis q de recursão da malha. Mostramos na primeira coluna o número de recursão da malha e na segunda o número de nós. Já na terceira e quarta colunas, aparecem os erros E_2 e E_∞ respectivamente, obtidos através das equações dadas em (3.6). Na quinta coluna, mostramos o número de iterações realizadas até que o erro E_∞ entre duas iterações consecutivas seja menor ou igual a 10^{-5} .

Nível q	Nós(N_q)	Erro (E_2)	Erro (E_∞)	iteraões
0	12	0.2197710333754259	0.2880283870487813	15
1	42	0.0460648596784225	0.0934631190385380	60
2	162	0.0107284485081276	0.0208107410179481	250
3	642	0.0026108950494286	0.0054715219111409	789
4	2562	0.0006496213141786	0.0013500664136921	3685
5	10242	0.0001605563412257	0.0003300160682150	15311
6	40962	0.0000389384247175	0.0000860077633263	58000

Tabela 3.1: Erros na estimativa da solução utilizando o método de Jacobi apresentado em (3.3) considerando a função dada em (3.5).

Podemos observar nos resultados da Tabela 3.1 que, ao contrário do erro de truncamento, ambos os erros E_2 e E_∞ diminuem à medida que o nível de recursão da malha aumenta. No entanto, para um nível de resolução mais fino o número de iterações cresce para que se obtenha a solução discreta, evidenciando a deterioração das taxas de convergência dos métodos tipo Jacobi. Na Figura

3.2 temos a representação dos resultados da Tabela 3.1, onde E_2 está representado em verde e E_∞ em vermelho.

Figura 3.2: Erro na solução em duas normas distintas (E_2 e E_∞) obtido ao aproximar a função dada em (3.5) pelo método de Jacobi.

Analisando os resultados obtidos é possível observar uma convergência quadrática do erro ao refinar a malha.

3.3 Método de Gauss-Seidel

O método de Gauss-Seidel se diferencia do método de Jacobi pelo fato da atualização das variáveis ocorrerem sequencialmente, já fazendo uso imediato das variáveis recém atualizadas.

Novamente consideremos o problema de Poisson dado em (3.4). Mostramos na primeira coluna da Tabela 3.2 o número de recursão da malha, na segunda o número de nós e na terceira e quarta colunas, aparecem os erros E_2 e E_∞ respectivamente, obtidos através das equações dadas em (3.6). Na quinta coluna, mostramos o número de iterações realizadas até que o erro E_∞ entre duas iterações consecutivas seja menor ou igual a 10^{-5} . Já na Figura 3.3 temos a representação dos resultados da Tabela 3.2 onde E_2 esta representado em verde e E_∞ em vermelho.

Nível q	Nós(N_q)	Erro (E_2)	Erro (E_∞)	iterações
0	12	0.1167543259335401	0.1907215672231216	10
1	42	0.0439500695110995	0.1065605339404346	38
2	162	0.0101357102002586	0.0270590031330777	173
3	642	0.0024748457596242	0.0066720123196260	425
4	2562	0.0006227176113713	0.0017370043170786	2398
5	10242	0.0001538985040340	0.0004290013197032	9303
6	40962	0.0000383402732378	0.0001070000634638	41946

Tabela 3.2: Erros na estimativa da solução utilizando o método de Gauss-Seidel considerando a função dada em (3.5).

Figura 3.3: Erro na solução em duas normas distintas (E_2 e E_∞) obtido ao aproximar a função dada em (3.5) pelo método de Gauss-Seidel.

Podemos observar que para o método de Gauss-Seidel é necessário um número menor de iterações do que o método de Jacobi para obter a convergência. Porém, para atingir a convergência utilizando ambos os métodos (Jacobi e Gauss-Seidel) o consequente esforço computacional é muito elevado em malhas finas, devido a lenta convergência destes métodos. Estes métodos não são eficientes para resolver estes problemas. No entanto, veremos que os métodos de Jacobi com relaxação e SOR apresentados a seguir tem grande importância no contexto do multigrid. Estes são adequados quando usados como suavizadores do erro nos métodos multigrid.

3.4 Esquema de relaxação: Weighted Jacobi e SOR

Um método de relaxação que pode ser usado no multigrid é obtido a partir de uma simples modificação no método de Jacobi. Este esquema, chamado *método de Jacobi com parâmetro de relaxação*, é dado por

$$\tilde{u}_k^{j+1} = (1 - w)\tilde{u}_k^j + w\tilde{u}_k^*, \quad (3.7)$$

no nosso caso $\tilde{u}_k^* = \left(\sum_{i=1}^n \frac{l_i}{L_i} u_i^j - f_k A_k \right) \left(\frac{1}{\sum_{i=1}^n \frac{l_i}{L_i}} \right)$ e $w \in \mathbb{R}$ conhecido como *fator de relaxação*.

Se as variáveis recém atualizadas forem utilizadas para obter a nova aproximação então teremos o método de Gauss-Seidel com parâmetro de relaxação, conhecido também como SOR.

Nos métodos de Jacobi e Jacobi com relaxação, nenhum elemento do vetor aproximação é usado sem que todos os seus elementos tenham sido atualizados e a atualização das N_q variáveis que definem o sistema pode ser feita em ordem arbitrária em cada iteração, pois estas dependem apenas dos valores encontrados na solução anterior. Com isso, pode-se usar multiprocessamento na medida em que a atualização de variáveis distintas dentro da mesma iteração pode ser feita em paralelo. Já os métodos de Gauss-Seidel e SOR, na maioria das vezes, têm convergência mais rápida do que o método de Jacobi porém, estes exigem processamento serial.

Devido ao fato da malha icosaédrica ser quase uniforme, relaxações ponto a ponto serão adequadas para a suavização do erro. Este fato contrasta com as necessidades em malhas latitude-longitude na esfera, em que métodos multigrid requerem ao menos relaxações por linha [Barros, 1991].

3.5 Processo de correção

Os métodos multigrid baseiam-se na premissa de que cada faixa de frequência do erro deve ser suavizada no espaçamento mais adequado para tal. Para que as componentes do erro de baixa frequência possam ser eliminadas com eficiência, estes métodos procuram trabalhar com uma sequência de malhas $G_N, G_{N-1}, \dots, G_0$ cada vez mais grossas, onde o erro pode ser rapidamente suavizado. Em cada nível de malha, as componentes do erro correspondentes são eficientemente reduzidas, acelerando o processo de convergência. Um exemplo de uma sequência de três níveis consecutivos da malha icosaédrica utilizada neste trabalho é mostrado na Figura 3.4. Nesta, a malha G_2 representa o nível mais fino, com 162 células. A malha G_1 , com 42 células e G_0 , a malha do nível mais grosso com 12 células.

Figura 3.4: Representação de três níveis consecutivos de malha. $G_2 = 162$ células, $G_1 = 42$ células. $G_0 = 12$ células.

O objetivo do método multigrid é resolver o problema definido na malha mais fina G_N , empregando as demais malhas como esquemas de correção. Como várias malhas são envolvidas no processo de correção, este necessita de transferências de informações entre as malhas, sendo necessário para isso definir os operadores de *interpolação* ou *prolongamentos* e *restrição*.

Antes de apresentarmos estes operadores vejamos, primeiramente a relação existente entre dois níveis consecutivos de malhas. Por exemplo, se considerarmos uma malha em um nível grosso definido por q , precisamos saber qual a relação entre as células deste nível com as células do nível de resolução mais fino $q + 1$.

Definimos como G_q a malha esférica icosaédrica com $N_q = 10(2^{2q}) + 2$, assumimos q como sendo o nível mais grosso de resolução e consideremos que a malha G_{q+1} é um nível de resolução mais fino com relação a malha G_q . Pela construção da malha icosaédrica teremos que algumas células das malhas dos diferentes níveis $G_{q+1}, G_q, G_{q-1}, \dots, G_0$ são encaixadas, como mostra a Figura 3.5. No entanto, a discretização na esfera não necessariamente fornecerá células totalmente encaixadas, como mostra a Figura 3.6. Não será realizado nesta dissertação um estudo detalhado da defasagem causada por esta deformação da malha na esfera, porém pode ser um dos temas para trabalhos futuros.

É possível observar a relação existente entre três níveis da malha icosaédrica considerando apenas uma célula da malha do nível mais grosso $q - 1$ (célula em vermelho). Observemos que o centro desta célula é o centro de todas as outras células dos níveis mais finos com relação a ela. Além disso, o centro de uma determinada célula k da malha do nível q (células em verde) é o centro da célula também de índice k da malha do nível $q + 1$ (células em azul).

Figura 3.5: *Relação entre três níveis consecutivos da malha icosaédrica: a malha vermelha indica o nível $q - 1$ (nível mais grosso), a malha verde o nível q e a malha azul o nível $q + 1$ (nível mais fino).*

Figura 3.6: *Representação de células não totalmente encaixadas.*

Considerando a relação entre os diferentes níveis da malha esférica geodésica icosaédrica, vejamos a seguir os operadores utilizados para a transferência de informações entre dois níveis consecutivos.

3.5.1 Interpolação

O objetivo da interpolação é obter aproximações para uma determinada função nos pontos da célula do nível fino $q + 1$ a partir dos valores já determinados nos pontos da malha do nível grosso q . O operador de interpolação transfere informações da malha grossa G_q para a malha mais fina

G_{q+1} e é denotado por

$$I_q^{q+1} = I_{G_q}^{G_{q+1}} : G_q \rightarrow G_{q+1}. \quad (3.8)$$

Um exemplo de operador de interpolação é dado pela interpolação linear, definida como segue.

Seja u uma função em coordenadas esféricas discretizada sobre a malha esférica icosaédrica em um nível grosso q com $N_q = 10(2^{2q}) + 2$ pontos, sendo a malha $G_q = \{P_0^q, P_1^q, \dots, P_{N_q}^q\}$. Queremos obter aproximações para os valores de u nos pontos da malha G_{q+1} , denotamos estes valores por $u_j^{q+1} = u(P_j^{q+1})$. As aproximações u_j^{q+1} são obtidas da seguinte maneira:

- (i) os valores da função nos pontos de G_{q+1} que estão contidos em G_q permanecerão os mesmos de G_q . Na Figura 3.7 os pontos verdes representam os pontos da malha $G_q \supset G_{q+1}$.
- (ii) os valores da função nos pontos de G_{q+1} que não estão contidos em G_q serão obtidos como a média aritmética entre os valores da função em dois pontos de células vizinhas em G_q . Na Figura 3.7 os quadrados representam os pontos da malha G_{q+1} que serão interpolados. Por exemplo, o valor a ser aproximado em P_j^{q+1} é obtido a partir dos valores da função nos pontos P_0^q e P_1^q por $u_j^{q+1} = \frac{1}{2}(u_0^q + u_1^q)$.

Figura 3.7: Representação da interpolação linear. A malha verde representa o nível mais grosso q e a malha azul o nível mais fino $q+1$. Pontos verdes: valores da função permanecem os mesmos da malha verde. Quadrados vermelhos: valores da função interpolados nesses pontos.

Para um teste numérico considermos u uma função em coordenadas esféricas discretizada na malha icosaédrica em um nível de resolução q . A partir desses valores discretizados faremos a interpolação linear a fim de obter aproximações para os valores da função nos pontos da malha do nível de resolução $q+1$. No teste a seguir vamos apresentar os erros E_2 e E_∞ dados pelas equações

(3.6), sendo e_i a diferença entre o valor exato da função em cada ponto da malha no nível $q + 1$ e o valor aproximado pelo esquema de interpolação neste mesmo ponto, deste mesmo nível. Os resultados são apresentados para diversos níveis de resolução. Seja

$$u(\lambda, \theta) = \text{sen}(\theta). \quad (3.9)$$

Mostramos na Tabela 3.3 os erros associados à interpolação linear para diferentes níveis de resolução da malha. Na primeira coluna da tabela representamos o nível em que os valores foram interpolados, na segunda o número de pontos da malha icosaédrica (nós) referente ao nível de interpolação e na terceira e quarta colunas os erros E_2 e E_∞ , na norma dois e norma do máximo, respectivamente.

Nível	Nós(N)	Erro (E_2)	Erro (E_∞)
1	42	0.07287494	0.14934919
2	162	0.02178501	0.04654802
3	642	0.00569402	0.01280475
4	2562	0.00143941	0.00335732
5	10242	0.00036085	0.00084918
6	40962	0.00009028	0.00021291
7	163842	0.00002257	0.00005327
8	655362	0.00000564	0.00001332
9	2621442	0.00000141	0.00000333
10	10485762	0.00000035	0.00000083

Tabela 3.3: Erros associados à interpolação linear considerando a função dada em (3.9). Primeira coluna: nível em que a função está sendo interpolada. Segunda coluna: número de pontos da malha. Terceira e quarta colunas: erros nas normas dois e infinito.

Com os resultados apresentados na Tabela 3.3 observa-se que os erros diminuem na medida em que aumentamos o nível de resolução da malha e fica comprovado a ordem dois de convergência da interpolação linear.

A Figura 3.8 mostra o erro calculado pelo valor absoluto da diferença entre o valor exato em cada ponto da malha e o valor aproximado pela interpolação linear, em quatro diferentes níveis de resolução da malha. Observemos a semelhança dos resultados apresentados na tabela acima. Os erros diminuem ao se interpolar em malhas mais finas.

Figura 3.8: Erro associado à interpolação linear considerando a função $u(\lambda, \theta) = \sin(\theta)$.

3.5.2 Restrição

O operador de restrição transfere informações da malha mais fina G_{q+1} para a mais grossa G_q , sendo denotado por:

$$I_{q+1}^q = I_{G_{q+1}}^{G_q} : G_{q+1} \rightarrow G_q. \quad (3.10)$$

O processo de restrição mais simples é chamado de *injeção*. Para entendê-lo consideremos a malha geodésica icosaédrica em dois níveis consecutivos, por exemplo, G_0 e G_1 . O processo de injeção transfere os valores de G_1 , que estão nas mesmas posições da malha G_0 , para G_0 . A Figura 3.9 ilustra este processo. Em vermelho as células da malha G_0 e em preto células da malha G_1 . À esquerda estão representados dois níveis de malhas, G_1 e G_0 e à direita mostramos as células do nível fino (células em preto pontilhadas) contidas nas células de G_0 . Portanto, neste processo de injeção, os pontos que são os centros de G_1 e também de G_0 são simplesmente projetados em G_0 . O mesmo acontece para os valores nesses pontos.

Figura 3.9: Processo de restrição da malha G_1 para G_0 .

Outro processo de restrição que pode ser aplicado à malha geodésica icosaédrica é o *full weighting* [Zhou e Fulton, 2009], dado por:

$$(\mathcal{I}_{q+1}^q \psi_q)_0 = \frac{A_0(\psi_q)_0 + \frac{1}{2} \sum_{i=1}^n A_i(\psi_q)_i}{A_0 + \frac{1}{2} \sum_{i=1}^n A_i}, \quad (3.11)$$

sendo A_0, A_i correspondente às áreas das células na grade fina e n os pontos da grade fina ($n = 5$ para pentágono ou $n = 6$ para hexágono). ψ_q corresponde ao resíduo e $(\psi_q)_i$ ao resíduo em cada célula vizinha da célula na qual o resíduo está sendo restringido, no nosso caso, esta célula é representada pelo subíndice zero.

Vejamos agora como funciona o processo de correção. Este tem como objetivo obter uma aproximação para o erro na malha mais grossa utilizada para corrigir a solução na malha mais fina. O processo de correção é realizado pelos seguintes passos.

- 1) Calcular o resíduo $r_N = f_N - \mathcal{L}_N \tilde{u}_N$.
- 2) Transferir o resíduo r_N para a malha do nível mais grosso, G_{N-1} , do seguinte modo

$$r_{N-1} = I_N^{N-1} r_N,$$

sendo I_N^{N-1} o operador de restrição (injeção ou *full weighting*).

- 3) Resolver a equação residual $\mathcal{L}_{N-1} e_{N-1} = r_{N-1}$ para obter uma aproximação e_{N-1} para o erro.
- 4) Interpolar a correção, e_{N-1} para a malha G_N , utilizando o operador de interpolação I_{N-1}^N , ou seja,

$$e_N = I_{N-1}^N e_{N-1}.$$

- 5) Corrigir a aproximação $\tilde{u}_N$ dada em G_N usando o vetor e_N resultante da interpolação, ou seja,

$$\tilde{u}_N = \tilde{u}_N + e_N.$$

A terceira etapa do esquema, que corresponde à resolução da equação residual em G_{N-1} , pode ser realizada de diversas formas. Uma possibilidade é resolver a equação residual apenas aplicando um método iterativo estacionário qualquer. Neste caso, a eficiência da correção fica dependente, não apenas da qualidade da interpolação, mas também da precisão da solução obtida para a equação residual em G_{N-1} . Outra alternativa é aplicar um método direto, como, por exemplo, o método de eliminação de Gauss para resolver a equação residual. Dessa forma, a correção estaria a cargo apenas da interpolação dos valores do vetor e_{N-1} . Entretanto, uma das dificuldades deste procedimento diz respeito ao tamanho e à estrutura da matriz que, em problemas reais, é geralmente muito grande e esparsa, de forma que o uso de um método direto é inadequado. Existe ainda uma terceira opção para resolver a equação residual na qual consiste em criar uma outra malha, mais grossa que G_{N-1} , definida por G_{N-2} , e então aplicar novamente o procedimento de correção descrito acima, agora com G_{N-1} e G_{N-2} . Esse processo pode então ser repetido de uma forma recursiva, para várias malhas, quantos forem necessárias para que a equação residual na malha mais grossa, G_0 , possa ser resolvida com o uso de um método direto. Esse conjunto de operações envolvendo um suavizador, uma hierarquia de malhas e transferências de vetores entre essas malhas compõem o método multigrid.

3.6 Métodos multigrid propostos na literatura

O primeiro método multigrid em malhas esféricas geodésicas icosaédricas foi proposto por [Heikes e Randall, 1995a], como parte de um esquema para a solução das equações de águas rasas. Estas equações, derivadas das equações de Euler a partir de algumas simplificações, descrevem razoavelmente bem a evolução de campos atmosféricos a 500 mb e são comumente empregadas como um primeiro teste real a métodos propostos na literatura. Parte importante em diversos métodos é a solução de equações elípticas resultantes nos esquemas. O segundo método apresentado para a solução destas equações foi proposto por [Thuburn, 1997], empregando diversas malhas aninhadas no processo de solução. Mais recentemente [Heikes *et al.*, 2013] apresentaram um novo método multigrid no qual é utilizado como operador de restrição o *full weighting* e interpolação linear. Implementaremos este método para a solução de uma equação do tipo Poisson e para verificar sua eficiência utilizaremos a Análise Local de Fourier descrita na próxima seção.

3.6.1 Método proposto por [Heikes e Randall, 1995a]

O método multigrid sugerido por [Heikes e Randall, 1995a] para a solução das equações elípticas utiliza um operador de interpolação da malha grossa para a malha fina, baseado em polinômios quadráticos em coordenadas de latitude e longitude. Cada ponto da malha $P_i = (\theta_i, \lambda_i)$ é dado em coordenadas esféricas. Para cada ponto da malha fina, por exemplo P_0 (veja a Figura 3.10) os autores associam seis pontos da malha grossa (os pontos P_1, P_2, P_3, P_4, P_5 e P_6 da Figura 3.10).

Note que a escolha dos pontos P_2 e P_6 é relativamente arbitrária, devido a uma falta de simetria em sua localização relativa a P_0 .

Figura 3.10: *Stencil de interpolação do método proposto por HR.*

O problema de interpolação é formulado a partir dos valores f_1, f_2, f_3, f_4, f_5 e f_6 da função na malha grossa em cada respectivo ponto da malha P_1, P_2, P_3, P_4, P_5 e P_6 . Então fica determinada uma função quadrática $p(\lambda, \theta) = c_1\theta^2 + c_2\theta\lambda + c_3\lambda^2 + c_4\theta + c_5\lambda + c_6$ que avaliada nos pontos $P_i, i = 1, \dots, 6$ assume os valores f_i . A função p é determinada através da resolução do sistema:

$$\begin{bmatrix} \theta_1^2 & \theta_1\lambda_1 & \lambda_1^2 & \theta_1 & \lambda_1 & 1 \\ \theta_2^2 & \theta_2\lambda_2 & \lambda_2^2 & \theta_2 & \lambda_2 & 1 \\ \theta_3^2 & \theta_3\lambda_3 & \lambda_3^2 & \theta_3 & \lambda_3 & 1 \\ \theta_4^2 & \theta_4\lambda_4 & \lambda_4^2 & \theta_4 & \lambda_4 & 1 \\ \theta_5^2 & \theta_5\lambda_5 & \lambda_5^2 & \theta_5 & \lambda_5 & 1 \\ \theta_6^2 & \theta_6\lambda_6 & \lambda_6^2 & \theta_6 & \lambda_6 & 1 \end{bmatrix} \cdot \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{bmatrix}. \quad (3.12)$$

Resolvendo o sistema linear dado em (3.12), obtemos as constantes $c_i = c_i(\theta_1, \dots, \theta_6, \lambda_1, \dots, \lambda_6, f_1, \dots, f_6)$ para $1 \leq i \leq 6$.

Para o ponto P_0 na malha fina, tem-se

$$c_1\theta_0^2 + c_2\theta_0\lambda_0 + c_3\lambda_0^2 + c_4\theta_0 + c_5\lambda_0 + c_6 = f_0. \quad (3.13)$$

Após algumas manipulações algébricas pode-se reescrever a interpolação na forma

$$w_1f_1 + w_2f_2 + w_3f_3 + w_4f_4 + w_5f_5 + w_6f_6 = f_0, \quad (3.14)$$

sendo $w_i = w_i(\theta_1, \dots, \theta_6, \lambda_1, \dots, \lambda_6)$. Estes pesos w_i dependem da posição dos pontos da esfera. [Heikes e Randall, 1995a], antes de resolver o sistema (3.12), transladam cada grupo de sete pontos, preservando suas posições relativas, de modo que $P_0 = (0, 0)$. Em seguida $P_1, P_2, \dots, P_6$ são levados para mais perto de P_0 substituindo cada ponto P_i , $i = 1, \dots, 6$, pelo ponto médio do arco entre P_0 e P_i . Este processo é repetido até que os pontos estejam quase no mesmo plano.

Para calcular os pesos w_i , os autores utilizam o programa *Mathematica* [Wolfram, 1988]. Os valores são armazenados em um arquivo que é lido pelo modelo antes da execução. Além disso, é necessário utilizar uma alta precisão numérica para superar o mal condicionamento da matriz de Vandermonde, dada em (3.12). [Heikes e Randall, 1995a] não apresentam resultados sobre a performance deste método multigrid. O método de interpolação utilizado é desnecessariamente complicado e inadequado. Requer alto custo de memória para armazenamento dos pesos para cada ponto da malha, além de intenso pré-processamento. O operador de restrição não é descrito no artigo (possivelmente emprega-se injeção simples).

3.6.2 Método proposto por [Thuburn, 1997]

Em [Thuburn, 1997] é proposto um método para a solução de equações elípticas em um novo modelo tipo multigrid. Primeiramente, o lado direito da equação de Poisson $\nabla^2 u = f$ será projetado em malhas mais grossas por uma média ponderada da área. Seja N o nível mais fino de resolução, escrevemos

$$Z_k^N = f_k^N A_k^N, \quad (3.15)$$

sendo A_k a área da célula k e f_k^N a função f avaliada em um ponto P_k (centro da célula k) da malha.

Os valores Z_k^{N-1} , do nível mais grosso $N - 1$ serão dados por

$$Z_k^{N-1} = Z_k^N + \sum_j Z_j^N / 2, \quad (3.16)$$

sendo j o índice que percorre os vizinhos da célula k em N , representado pelos seis pontos $Z_1^N, \dots, Z_6^N$ na Figura 3.11. A indexação é escolhida de forma que os centros das células do nível N e $N - 1$ coincidam.

Figura 3.11: Representação dos índices das células utilizadas para obter Z_k^{N-1} .

Para os outros níveis da hierarquia da malha, Z^q é calculado em termos de Z^{q+2} por

$$Z_k^q = \sum_j Z_j^{q+2} + \sum_l Z_l^{q+2}/2, \quad (3.17)$$

sendo j o índice que percorre as células completamente dentro da célula k . Na Figura 3.12 a célula k que pertence ao nível mais grosso q está representada em vermelho e o conjunto de índices j está representado pelas células pintadas de rosa. Já o índice l percorre as células em $q+2$ que sobrepõe a aresta da célula k sobre a grade q (células pintadas, pela metade, em azul na Figura 3.12).

Figura 3.12: Representação dos índices das células utilizadas para obter Z_k^q .

Depois de projetar a quantidade f_k nas malhas dos níveis mais grossos, utilizando o processo descrito anteriormente, o algoritmo proposto por [Thuburn, 1997] para resolver a equação de Poisson segue os seguintes passos:

- (I) A solução é aproximada na malha mais grossa utilizando como esquema de relaxação o método de Jacobi.
- (II) Os valores relaxados são interpolados utilizando a interpolação linear, descrita na Seção 3.5.1, neste capítulo.
- (III) Apenas os valores interpolados são relaxados.
- (IV) Todos os valores são relaxados.
- (V) O processo é repetido até um nível de resolução mais fino.

O autor relata que o custo da CPU é dominado pela solução da equação de Poisson e pelo problema elíptico que surge em dar um passo de tempo semi-implícito, juntos correspondem em

cerca de 70 % do custo total do seu modelo. O número de iterações requeridas no método "multigrid" proposto por ele variou entre 15 e 30 por nível. Observemos que o algoritmo de Thuburn na verdade não se constitui em um método multigrid, uma vez que as malhas grossas não são utilizadas para a correção dos erros nas malhas finas. Seu algoritmo seria mais adequadamente chamado de *nested iteration*, uma vez que apenas utiliza os valores de malha grossa para obtenção de aproximações iniciais para as iterações de Jacobi nas malhas finas subsequentes.

3.6.3 Método escolhido para implementação: proposto por [Heikes et al., 2013]

Podemos resumir o método multigrid como uma combinação de um método de suavização com um problema de correção em malha mais grossa. O método multigrid proposto por [Heikes et al., 2013] utiliza como operador de restrição o *full weighting* apresentado, anteriormente, na Seção 3.5.2 e faz uso da interpolação linear, apresentada na Seção 3.5.1 deste capítulo.

O esquema mais básico do método multigrid e utilizado por [Heikes et al., 2013] é o chamado *ciclo em V*, ou *V-ciclo*. Ele recebe este nome em virtude da trajetória que percorre entre as diferentes malhas construídas para o problema.

Antes de descrevermos o algoritmo do V-ciclo, com objetivo de simplificar a notação, vamos definir o vetor do lado direito da equação residual r_{N-1} por f_{N-1} e ao invés de utilizar e_{N-1} para a solução da equação residual, vamos utilizar $\tilde{u}_{N-1}$. O V-ciclo é apenas um membro de uma família de ciclos do multigrid e é definido de forma recursiva como segue:

Ciclo multigrid - V-ciclo: $V(\nu_1, \nu_2)$.

- (i) Aplicar ν_1 vezes um método de relaxação em $L_N u_N = f_N$ na malha G_N para obter uma aproximação $\tilde{u}_N^{(k)}$.
- (ii) Se G_{N-1} é a malha do nível mais grosso, ou seja, $G_{N-1} = G_0$, então vá para o item (v) tomando $\tilde{u}_{N-1}^{(k+1)} = \tilde{u}_{N-1}^{(k)}$, caso contrário continue.
- (iii) Calcular o resíduo $r_N = f_N - L_N \tilde{u}_N^{(k)}$ e o transferir para G_{N-1} fazendo $f_{N-1} = I_N^{N-1} r_N$.
- (iv) Retornar para (i) considerando a malha G_{N-1} , $\gamma = 1$ vezes.
- (v) Corrigir $\tilde{u}_N^{(k+1)} = \tilde{u}_N^{(k)} + I_{N-1}^N \tilde{u}_{N-1}$
- (vi) Aplicar ν_2 passos de relaxação em $L_N u_N = f_N$ na malha G_N com chute inicial $\tilde{u}_N^{(k+1)}$.

Teoricamente, o número de passos de relaxação ν_1 , ν_2 , aplicados nas etapas (i) e (vi) do procedimento devem ser suficientes para tornar o erro suave. Os valores de ν_1 e ν_2 usados na prática são, frequentemente, 1, 2 ou 3.

O V-ciclo será denotado por $V(\nu_1, \nu_2)$, sendo ν_1 o número de relaxações realizadas no início do processo (observe os itens (i) do processo acima) e ν_2 o número de relaxações após a correção (observe os itens (v) do processo acima).

A Figura 3.6.3 representa um V-ciclo, começando o processo da malha fina G_3 . O V-ciclo multigrid é apenas um de uma família de ciclos multigrid que são denominados γ -ciclos multigrid. Na prática, utiliza-se $\gamma = 1$ para V-ciclo e $\gamma = 2$, denominado W-ciclo. Veja como fica o ciclo multigrid, fazendo a escolha de $\gamma=2$ na Figura 3.14, começando o processo novamente da malha G_3 . Em

ambas as figuras as flechas entre as malhas representam os operadores de transferência de restrição (descida) e interpolação (subida).

Figura 3.13: Representação do V-ciclo partindo de uma malha com 642 pontos.

Figura 3.14: Representação do W-ciclo partindo de uma malha com 642 pontos.

3.7 Análise Local de Fourier

Nesta seção, estamos interessados em analisar a eficiência e precisão do método multigrid. A Análise Local de Fourier (para simplificar LFA, que vem da sigla do nome em inglês *Local Fourier Analysis*) é uma ferramenta essencial para este estudo. Ela fornece estimativas de taxas de convergência, o que facilita verificar se o método está implementado corretamente. Para isso, primeiramente vamos trabalhar com ferramentas teóricas para prever as taxas de convergência que podem ser esperadas pelo método de relaxação. Esta análise vai medir os efeitos dos componentes do multigrid nas componentes de Fourier do erro.

Faremos a Análise de Fourier para o método multigrid utilizando a malha uniforme hexagonal sobre o plano que, segundo [Zhou e Fulton, 2009] é uma boa aproximação para a malha geodésica icosaédrica, já que esta consiste de células hexagonais quase uniformes juntamente com doze células pentagonais.

A análise realizada neste trabalho é similar àquela feita em [Trottenberg *et al.*, 2001] para grades retangulares, já que grades hexagonais são retangulares em coordenadas oblíquas, como veremos a seguir.

3.7.1 Grades hexagonais e operadores

A malha uniforme hexagonal G_h é definida como uma coleção de pontos x_j^h como segue

$$G_h = \{x_j^h = (j_1 a_1 + j_2 a_2)h : j = (j_1, j_2) \in \mathbb{Z}^2\},$$

em que os x_j^h são os centros das células hexagonais (representados pelos pontos na Figura 3.15), sendo h a distância entre os centros das células e a_1 e a_2 vetores unitários definidos no sistema de coordenadas oblíquas por

$$a_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, a_2 = \begin{pmatrix} 1/2 \\ \sqrt{3}/2 \end{pmatrix}.$$

A Figura 3.15 representa a malha hexagonal considerada para fazer a análise.

Vamos considerar G_h como sendo a malha do nível mais fino e a malha do nível consecutivo mais grosso será denotada por G_{2h} com $x_j^{2h} = x_{2j}^h$ para $j = (j_1, j_2) \in \mathbb{Z}^2$. As malhas dos próximos níveis mais grossos $G_{4h}, G_{8h}, \dots$ são definidas do mesmo modo.

Para fazer a Análise de Fourier, vamos trabalhar com funções da forma

$$\varphi_h(\theta, x) = e^{i\theta x/h},$$

em que $\theta \in \mathbb{R}^2$ caracteriza a frequência da função φ_h ou o número de onda de Fourier relativo à malha G_h . Estas funções $\varphi_h(\theta, x)$ são chamadas de *modos* ou *funções de Fourier*.

Na malha hexagonal é conveniente escrevermos $\theta = \theta_1 b_1 + \theta_2 b_2$, sendo $\{b_1, b_2\}$ base ortogonal a $\{a_1, a_2\}$, com b_1 e b_2 dados por

$$b_1 = \begin{pmatrix} 1 \\ -1/\sqrt{3} \end{pmatrix}, b_2 = \begin{pmatrix} 0 \\ 2/\sqrt{3} \end{pmatrix},$$

Figura 3.15: Grade hexagonal uniforme G_h . Vetores a_1 e a_2 definidos no sistema de coordenadas oblíquas. Os pontos representam os centros das células da malha G_h .

satisfazendo $a_i b_j = \delta_{ij} = \begin{cases} 1 & \text{se } i = j \\ 0 & \text{se } i \neq j \end{cases}$. Dessa forma, em G_h temos

$$\varphi_h(\theta, x_j^h) = e^{i\theta x_j^h/h} = e^{i\theta(j_1 a_1 + j_2 a_2)} = e^{i(j_1 \theta_1 + j_2 \theta_2)}, \quad j \in \mathbb{Z}^2.$$

Além disso,

$$\varphi_h(\theta, x) = \varphi_h(\theta', x), \quad x \in G_h$$

se, e somente se, a diferença entre θ_1 e θ'_1 e a diferença entre θ_2 e θ'_2 são ambas múltiplas de 2π . Portanto, é suficiente considerarmos $\varphi_h(\theta, x)$ com $\theta \in \Theta = [-\pi, \pi)^2$ o qual vamos nos referir como frequências visíveis em G_h .

Para a suavização e análise em duas grades, precisamos distinguir componentes de baixas e altas frequências de G_h com relação a G_{2h} . Note que para pontos na grade G_{2h} , temos a seguinte igualdade:

$$\varphi_h(\theta, x_{2j}^h) = \varphi_{2h}(2\theta, x_j^{2h}),$$

ou seja, a função com número de onda θ relativo a G_h tem número de onda 2θ relativo a G_{2h} . Assim, somente

$$\varphi_h(\theta, \cdot) \text{ com } -\frac{\pi}{2} \leq \theta < \frac{\pi}{2},$$

são visíveis em G_h . Para cada $\theta' \in [-\frac{\pi}{2}, \frac{\pi}{2})^2$ três outras componentes de frequência $\varphi_h(\theta, \cdot)$ com $\theta \in [-\pi, \pi)^2$ coincidem em G_{2h} com $\varphi_h(\theta', \cdot)$ e não são visíveis em G_{2h} .

Dessa maneira, definiremos as componentes de baixas e altas frequências como segue:

A função φ é uma componente de baixa frequência se, e somente se, $\theta \in \Theta_{\text{low}} := \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)^2$ e é uma componente de alta frequência se, e somente se, $\theta \in \Theta_{\text{high}} := \Theta \setminus \Theta_{\text{low}} = [-\pi, \pi)^2 \setminus \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)^2$. A Figura 3.16 ilustra as regiões de baixas e altas frequências. A região do quadrado pequeno representa as baixas frequências, já a região mais escura e sem o quadrado pequeno ilustra a de altas frequências.

Figura 3.16: Baixas frequências (região em branco), altas frequências (região mais escura).

Os operadores trabalhados aqui são assumidos lineares com coeficientes constantes. Seja L_h correspondente ao stencil $[s_k]_h$ indexado por $k = (k_1, k_2) \in \mathbb{Z}^2$, o que significa que para qualquer função u_h em G_h definimos

$$L_h u_h(x_j^h) = \sum_{k \in K} s_k u_h(x_{j+k}^h), x_j^h \in G_h, k = (k_1, k_2) \in \mathbb{Z}^2, \quad (3.18)$$

sendo $K \subset \mathbb{Z}^2$ um conjunto finito de índices. Para o operador L_h segue o seguinte lema.

Lema 3.7.1. [Zhou e Fulton, 2009]. Para qualquer operador discreto L_h em G_h descrito na forma (3.18), todos os modos de Fourier $\varphi_h(\theta, x)$ são autofunções, com

$$L_h \varphi_h(\theta, x) = \tilde{L}_h(\theta) \varphi_h(\theta, x), \quad x \in G_h,$$

sendo o autovalor $\tilde{L}_h(\theta) := \sum_{k \in K} s_k e^{i(k_1 \theta_1 + k_2 \theta_2)}$ chamado o *símbolo* de L_h .

Exemplo 3.7.1. O símbolo do operador laplaciano, $-\nabla^2$, é dado por

$$\tilde{L}_h(\theta) = \frac{4}{3h^2} [3 - \cos(\theta_1) - \cos(\theta_2) - \cos(\theta_2 - \theta_1)]. \quad (3.19)$$

Veja no Apêndice A.2.1 que o operador $-\nabla^2$ pode ser discretizado utilizando sete pontos da seguinte

maneira

$$L_h u_0 = \frac{2}{3h^2} \left(6u_0 - \sum_{i=1}^6 u_i \right), \quad (3.20)$$

sendo conveniente usar a indexação: o subíndice zero denota o ponto central e os subíndices de 1 a 6 representam os vizinhos. Na Figura 3.17 fica clara essa indexação.

Figura 3.17: Representação da indexação do stencil utilizado para obter a discretização de $-\nabla^2$ apresentada em (3.20). Sendo a_1 e a_2 os vetores unitários definidos no sistema de coordenadas oblíquas.

Em coordenadas oblíquas, temos o stencil representado pela seguinte matriz

$$\frac{2}{3h^2} \begin{bmatrix} -1 & -1 & 0 \\ -1 & 6 & -1 \\ 0 & -1 & -1 \end{bmatrix}_h$$

que geometricamente pode ser representado pela Figura 3.18.

Figura 3.18: Representação do stencil em coordenadas oblíquas utilizado para obter o símbolo dado em (3.19).

Utilizando os pontos do stencil acima no Lema 3.7.1, temos o símbolo dado em (3.19) como mostrado em A.2.1.

Para obter o fator de suavização e os fatores de convergência em duas malhas, além do operador L_h também precisamos definir os seguintes operadores:

- S_h - operador de suavização,
- I_h^{2h}, I_{2h}^h - operadores de transferências,
- K_h^{2h} - operador de correção da malha grossa,
- M_h^{2h} - operador de duas malhas.

O objetivo da LFA é determinar o *fator de convergência*, $\mu_2(M_h^{2h})$, que nos fornece a taxa de redução do erro, para o método de relaxação, a cada ciclo do multigrid.

Primeiramente vamos obter o fator de suavização $\mu_1(S_h)$ que estima o quanto o esquema de relaxação reduz as componentes de altas frequências do erro.

3.7.2 Análise de suavização

O desempenho do multigrid depende em grande parte da eficiência do esquema de relaxação em suavizar o erro em cada malha. Consideremos o operador $L_h = -\nabla^2$ e vamos mostrar como LFA é utilizada para fazer a análise da suavização dos esquemas de relaxação e calcular o fator de suavização $\mu_1(S_h)$. Nesta análise, as funções de Fourier $\varphi(\theta, x)$ são autofunções dos correspondentes operadores de suavização.

Consideremos a discretização da equação diferencial parcial $L_h u_h = f_h$, e assumimos que o esquema de relaxação pode ser escrito como

$$L_h^+ \bar{u}_h + L_h^- \tilde{u}_h = f_h, \quad (3.21)$$

sendo $\tilde{u}_h$ a velha aproximação da solução discreta verdadeira u_h (aproximação antes de aplicar o esquema de relaxação) e $\bar{u}_h$ a nova aproximação para u_h (depois de aplicar o esquema de relaxação). Assim, a relaxação é caracterizada pela partição

$$L_h = L_h^+ + L_h^-.$$

Exemplo 3.7.2. Consideremos a equação de Poisson $-\nabla^2 u = f$ com a discretização para $-\nabla^2$ dada em (3.20). Uma formulação para o método Weighted Jacobi considerando malha hexagonal estudada, pode ser dada por

$$\bar{u}_0 = (1 - \omega)\tilde{u}_0 + \frac{\omega}{6} \left(\frac{3h^2}{2} f_0 + \sum_{i=1}^6 \tilde{u}_i \right), \quad (3.22)$$

sendo w o parâmetro de relaxação. Para esta formulação, obtemos os operadores L_h^+ e L_h^- como seguem

$$L_h^+ = \frac{2}{3h^2} \begin{bmatrix} 0 & 0 \\ 0 & 6/w & 0 \\ & 0 & 0 \end{bmatrix}, \quad L_h^- = \frac{2}{3h^2} \begin{bmatrix} -1 & -1 \\ -1 & 6(1 - 1/w) & -1 \\ & -1 & -1 \end{bmatrix}.$$

Exemplo 3.7.3. Consideremos novamente o problema $-\nabla^2 u = f$. Para o método de Weighted Gauss-Seidel (SOR) os operadores L_h^+ e L_h^- são dados por

$$L_h^+ = \frac{2}{3h^2} \begin{bmatrix} 0 & 0 \\ 0 & 6/w & -1 \\ & -1 & -1 \end{bmatrix}, \quad L_h^- = \frac{2}{3h^2} \begin{bmatrix} -1 & -1 \\ -1 & 6(1 - 1/w) & 0 \\ & 0 & 0 \end{bmatrix},$$

sendo $0 < w < 2$ o parâmetro de relaxação.

Subtraindo (3.21) de $L_h u_h = f_h$, temos

$$L_h u_h - (L_h^+ \bar{u}_h + L_h^- \tilde{u}_h) = 0.$$

Como podemos representar L_h por $(L_h^+ + L_h^-)$ obtemos

$$L_h^+ u_h + L_h^- u_h - (L_h^+ \bar{u}_h + L_h^- \tilde{u}_h) = 0.$$

Assim,

$$(L_h^+)^{-1} L_h^+ u_h + (L_h^+)^{-1} L_h^- u_h - ((L_h^+)^{-1} L_h^+ \bar{u}_h + (L_h^+)^{-1} L_h^- \tilde{u}_h) = 0,$$

o que acarreta

$$u_h - \bar{u}_h = -(L_h^+)^{-1} L_h^-(u_h - \tilde{u}_h). \quad (3.23)$$

Escrevendo os novos e os velhos erros $\bar{v}_h = u_h - \bar{u}_h$ e $\tilde{v}_h = u_h - \tilde{u}_h$, respectivamente, a equação (3.23) torna-se

$$\bar{v}_h = S_h \tilde{v}_h \quad (3.24)$$

sendo $S_h = -(L_h^+)^{-1} L_h^-$ o *operador de suavização*. Aplicando L_h^+ , L_h^- e S_h nas funções de Fourier $\varphi_h(\theta, x)$ temos o seguinte lema.

Lema 3.7.2. [Zhou e Fulton, 2009]. Para qualquer esquema de relaxação da forma (3.21), todos os modos de Fourier $\varphi_h(\theta, x)$ para o qual $\tilde{L}_h^+(\theta) \neq 0$ são autofunções do operador de suavização S_h , com

$$S_h \varphi_h(\theta, x) = \tilde{S}_h(\theta) \varphi_h(\theta, x), \quad x \in G_h,$$

sendo $\tilde{S}_h(\theta) := -\tilde{L}_h^-(\theta)/\tilde{L}_h^+(\theta)$ chamado de *fator de amplificação*.

Do Lema 3.7.2, podemos imediatamente definir o *fator de suavização* $\mu_1(S_h)$ por

$$\mu_1(S_h) := \sup \left\{ \left| \tilde{S}_h(\theta) \right| : \theta \in \Theta_{\text{high}} \right\}. \quad (3.25)$$

Assumindo que os erros de baixas frequências são eliminados por correções calculadas nas grades grossas, o fator de suavização $\mu_1(S_h)$ nos fornece uma estimativa para a eficiência global do método multigrid.

Exemplo 3.7.4. Consideremos o problema $-\nabla^2 u = f$ com a discretização dada em (3.20). Usando Weighted Jacobi (WJ) como esquema de relaxação escrevemos

$$\bar{u}_0 = (1 - \omega) \tilde{u}_0 + \frac{\omega}{6} \left(\frac{3h^2}{2} f_0 + \sum_{i=1}^6 \tilde{u}_i \right), \quad (3.26)$$

sendo w o parâmetro de relaxação. O correspondente operador de suavização é

$$(S_h^J(\omega) v_h)_0 = (1 - \omega) v_0 + \frac{\omega}{6} \sum_{i=1}^6 v_i, \quad (3.27)$$

com símbolo

$$\tilde{S}_h^J(\theta, \omega) = 1 - \frac{2\omega}{3} \left[\sin^2 \left(\frac{\theta_1}{2} \right) + \sin^2 \left(\frac{\theta_2}{2} \right) + \sin^2 \left(\frac{\theta_1 - \theta_2}{2} \right) \right]. \quad (3.28)$$

Considerando w positivo temos, como calculado no Apêndice A.2.2, o seguinte fator de suavização

$$\mu_1(S_h^J(\omega)) = \max \left\{ 1 - \omega \left(1 - \frac{\sqrt{2}}{3} \right), \frac{3\omega}{2} - 1 \right\}, \quad (3.29)$$

sendo que o primeiro valor corresponde ao extremo da fronteira interior e o segundo valor, ao interior da região de alta frequência Θ_{high} .

Para o método de Weighted Gauss-Seidel o símbolo correspondente é dado por

$$\tilde{S}_h^G(\theta, \omega) = -\frac{\tilde{L}_h^-(\theta)}{\tilde{L}_h^+(\theta)} = \frac{6(1-w) + w(e^{i\theta_1} + e^{i\theta_2} + e^{i(\theta_2-\theta_1)})}{6 - w(e^{-i\theta_1} + e^{-i\theta_2} + e^{-i(\theta_2-\theta_1)})}, \quad (3.30)$$

sendo o fator de suavização $\mu_1(S_h^G(w, \theta))$ dado por

$$\mu_1(S_h^G(w, \theta)) := \sup \left\{ \left| \tilde{S}_h^G(w, \theta) \right| : \theta \in \Theta_{\text{high}} \right\}. \quad (3.31)$$

Na Figura 3.19, observemos o comportamento dos fatores de suavização μ_1^J (Jacobi) e μ_1^G (Gauss-Seidel). No eixo x representamos a variação do parâmetro de relaxação w e no eixo y estão representados os valores de μ_1^J e μ_1^G para o w correspondente. O valor ótimo do fator de suavização para Jacobi é obtido com $w = 0.98$. Para este w o fator de suavização μ_1^J é aproximadamente 0.4789. Já para Gauss-Seidel o valor ótimo é $\mu_1^G = 0.5396$ e ocorre com $w = 1.02$. Para $w = 1$ temos um fator de suavização de 0.5 com o Jacobi, e de 0.5399 com Gauss-Seidel.

Figura 3.19: Representação dos fatores de suavização μ_1^J (Jacobi) e μ_1^G (Gauss-Seidel). Em verde está a representação para o fator de suavização de Jacobi e em vermelho, para Gauss-Seidel.

3.7.3 Análise em duas grades

A análise em duas grades melhora a análise de suavização por incluir os efeitos das transferências das malhas. Sabemos que no método multigrid, no ciclo de duas grades, a correção na malha grossa é composta pelo cálculo do resíduo na grade fina G_h , que é transferido para a grade grossa G_{2h} . Resolve-se a equação residual e transfere-se o resultado da correção de volta para a malha fina, que é adicionada à aproximação anterior. O operador de correção, K_h^{2h} é dado por

$$K_h^{2h} = I_h - I_{2h}^h (L_{2h})^{-1} I_h^{2h} L_h,$$

sendo I_h^{2h} o operador de restrição, I_{2h}^h o operador de interpolação e I_h o operador identidade em G_h .

O ciclo em duas grades consiste de ν_1 e ν_2 passos de relaxação sobre a malha fina, antes e depois da correção na malha grossa, respectivamente. Representamos este ciclo pelo operador

$$M_h^{2h} := S_h^{\nu_2} K_h^{2h} S_h^{\nu_1}.$$

Para calcular os fatores de convergência de M_h^{2h} , vamos analisar como os operadores L_h , L_{2h} , I_h^{2h} , I_{2h}^h e S_h atuam nas funções de Fourier $\varphi(\theta, \cdot)$. Para qualquer frequência baixa $\theta = (\theta_1, \theta_2) \in \Theta_{\text{low}} = [-\frac{\pi}{2}, \frac{\pi}{2}]^2$, consideremos as frequências

$$\theta^{(0)} := (\theta_1, \theta_2), \quad \theta^{(1)} := (\tilde{\theta}_1, \theta_2), \quad \theta^{(2)} := (\theta_1, \tilde{\theta}_2), \quad \theta^{(3)} := (\tilde{\theta}_1, \tilde{\theta}_2),$$

sendo

$$\tilde{\theta}_i = \begin{cases} \theta_i + \pi, & \text{se } \theta_i < 0 \\ \theta_i - \pi, & \text{se } \theta_i \geq 0 \end{cases}$$

Observemos que $\varphi_h(\theta^{(\ell)}, x_{2j}^h) = \varphi_h(\theta, x_{2j}^h) = \varphi_{2h}(2\theta, x_j^{2h})$, $\ell = 0, 1, 2, 3$. Estas quatro funções $\varphi_h(\theta^{(\ell)}, \cdot)$ são os *harmônicos*, relacionados à uma frequência θ . Para um dado $\theta = \theta^{(0)} \in \Theta_{\text{low}}$ definimos seu *espaço de harmônicos (de dimensão 4)* por

$$E_h^\theta = \text{span}\{\varphi_h(\theta^{(\ell)}, \cdot) : \ell = 0, 1, 2, 3\}.$$

Para qualquer $\psi_h \in E_h^\theta$ podemos representar ψ_h da forma

$$\psi_h(x) = \sum_{\ell=0}^3 A^{(\ell)} \varphi_h(\theta^{(\ell)}, x), \quad x \in G_h. \quad (3.32)$$

com coeficientes ou amplitudes $A^{(\ell)}$ definidos unicamente. Uma vez que $e^{i\tilde{\theta}_\ell} = -e^{i\theta_\ell}$ então

$$\psi_h(x) = \Psi_h(x) \varphi_h(\theta, x), \quad x \in G_h, \quad (3.33)$$

sendo Ψ_h dada pela forma *four-color* como segue:

$$\Psi_h(x_j^h) = \begin{cases} \Psi^{(0)}, & \text{se } j_1 \text{ par e } j_2 \text{ par,} \\ \Psi^{(1)}, & \text{se } j_1 \text{ ímpar e } j_2 \text{ par,} \\ \Psi^{(2)}, & \text{se } j_1 \text{ par e } j_2 \text{ ímpar,} \\ \Psi^{(3)}, & \text{se } j_1 \text{ ímpar e } j_2 \text{ ímpar.} \end{cases} \quad (3.34)$$

A Figura 3.20 mostra as posições dos $\Psi^{(\ell)}$ na malha hexagonal uniforme, com os valores $\Psi^{(\ell)}$

dados em termos dos coeficientes $A^{(\ell)}$ por

$$\begin{pmatrix} \Psi^{(0)} \\ \Psi^{(1)} \\ \Psi^{(2)} \\ \Psi^{(3)} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} A^{(0)} \\ A^{(1)} \\ A^{(2)} \\ A^{(3)} \end{pmatrix}. \quad (3.35)$$

Figura 3.20: Posições dos $\Psi^{(\ell)}$ na malha de acordo com (3.34).

Todos os modos $\varphi(\theta, \cdot)$ são autofunções de L_h (Lema 3.7.1) e pelo Lema 3.7.2 também são autofunções de S_h . Estes fatos são úteis para os seguintes resultados.

- **Operadores discretos**

Dado que um operador L_h é representado por um stencil $[s_k(h)]_h$, ele é um operador linear em E_h^θ . Do Lema 3.7.1 a matriz de representação é:

$$\widehat{L}_h(\theta) = \begin{pmatrix} \widetilde{L}_h(\theta^{(0)}) & & & \\ & \widetilde{L}_h(\theta^{(1)}) & & \\ & & \widetilde{L}_h(\theta^{(2)}) & \\ & & & \widetilde{L}_h(\theta^{(3)}) \end{pmatrix}. \quad (3.36)$$

Por exemplo, para o operador $-\nabla^2$ discretizado como em (3.20) a matriz $\widehat{L}_h(\theta)$ é construída utilizando o símbolo dado em (3.19).

Do mesmo modo, o operador L_{2h} , representado pelo stencil $[s_k(2h)]_{2h}$, atuando em $\varphi_{2h}(2\theta, x)$ produz

$$L_{2h}\varphi_{2h}(2\theta, x) = \widetilde{L}_{2h}(2\theta)\varphi_{2h}(2\theta, x), \quad x \in G_{2h}.$$

A análise em duas grades assume que a equação é resolvida exatamente na grade grossa, portanto, omite-se o espaço nulo de $\tilde{L}_{2h}$, ou seja,

$$\Theta_0 = \{\theta \in \Theta_{\text{low}} : \tilde{L}_{2h}(2\theta) = 0\}.$$

Por exemplo, para o operador de Laplace discretizado como em (3.20) e com símbolo dado pela equação (3.19), o espaço nulo Θ_0 contém somente a frequência $\theta = (0, 0)$.

• Suavização

Como o espaço dos harmônicos E_h^θ é invariante sobre S_h , então S_h pode ser representado por $\hat{S}_h(\theta)$ (matriz 4×4) [Zhou e Fulton, 2009].

No caso de $\varphi(\theta, x)$ ser autofunção de S_h pelo Lema 3.7.2

$$\hat{S}_h(\theta) = \begin{pmatrix} \tilde{S}_h(\theta^{(0)}) & & & \\ & \tilde{S}_h(\theta^{(1)}) & & \\ & & \tilde{S}_h(\theta^{(2)}) & \\ & & & \tilde{S}_h(\theta^{(3)}) \end{pmatrix}. \quad (3.37)$$

Por exemplo, para o método de Jacobi utilizamos o fator de suavização dado em (3.28) para construir a matriz $\hat{S}_h(\theta)$. E para o método de Gauss-Seidel o fator de suavização utilizado é dado na equação (3.30).

• Restrição

Assumimos I_h^{2h} representado por um stencil $[\hat{t}_k]_h^{2h}$, indexado por k em algum conjunto finito de índices $k \in K_R \subset \mathbb{Z}^2$. Então

$$\left(I_h^{2h} \psi_h\right)(x_j^{2h}) = \sum_{k \in K_R} \hat{t}_k \psi_h(x_{2j+k}^h), \quad K_R \subset \mathbb{Z}^2. \quad (3.38)$$

Tal restrição satisfaz o lema a seguir.

Lema 3.7.3. [Zhou e Fulton, 2009]. Se $\psi_h \in E_h^\theta$ com coeficientes $A^{(\ell)}$ dado em (3.32), então a restrição em (3.38) produz $\psi_{2h} = I_h^{2h} \psi_h$ satisfazendo $\psi_{2h}(x) = A_{2h} \varphi_{2h}(2\theta, x)$ para $x \in G_{2h}$ com

$$A_{2h} = \left(\tilde{I}_h^{2h}(\theta^{(0)}), \tilde{I}_h^{2h}(\theta^{(1)}), \tilde{I}_h^{2h}(\theta^{(2)}), \tilde{I}_h^{2h}(\theta^{(3)})\right) \begin{pmatrix} A^{(0)} \\ A^{(1)} \\ A^{(2)} \\ A^{(3)} \end{pmatrix}, \quad (3.39)$$

onde $\tilde{I}_h^{2h}(\theta) := \sum_{k \in K_R} \hat{t}_k e^{ik \cdot \theta}$ é o símbolo do operador de restrição.

Assim, a restrição I_h^{2h} é representada pela matriz 1×4 formada pelos termos $\tilde{I}_h^{2h}(\theta^{(\ell)})$, $l = 0, \dots, 3$ do lado direito de (3.39) e será denotada por $\hat{I}_h^{2h}(\theta)$.

A restrição mais natural é o *full weighting* que sobre grades hexagonais tem stencil da forma

$$\frac{1}{8} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}_h^{2h}, \quad (3.40)$$

sendo o símbolo correspondente $\tilde{I}_h^{2h}(\theta) = \frac{1}{4}[1 + \cos(\theta_1) + \cos(\theta_2) + \cos(\theta_2 - \theta_1)]$. Se considerarmos como operador de restrição *injeção*, o símbolo correspondente é $\tilde{I}_h^{2h}(\theta) = 1$.

• Prolongação

Igualmente assumimos que o operador de prolongação I_{2h}^h indexado por k em algum conjunto finito de índices $K_P \subset \mathbb{Z}^2$ é representado por um stencil $]t_k[_{2h}^h$. Então

$$(I_{2h}^h \psi_{2h})(x_{2j+k}^h) = \sum_{k'} t_{(2k'+k)} \psi_{2h}(x_{j-k'}^{2h}), \quad 2k' + k \in K_P \quad (3.41)$$

Usando a forma four-color mostrada em (3.34), temos o seguinte lema.

Lema 3.7.4. [Zhou e Fulton, 2009]. Se $\psi_{2h}(x) = A_{2h} \varphi_{2h}(2\theta, x)$ para $x \in G_{2h}$, então a equação em (3.41) produz $\psi_h = I_{2h}^h \psi_{2h} \in E_h^\theta$ e os coeficientes $A^{(\ell)}$ em (3.32) satisfazem

$$\begin{pmatrix} A^{(0)} \\ A^{(1)} \\ A^{(2)} \\ A^{(3)} \end{pmatrix} = \begin{pmatrix} \tilde{I}_{2h}^h(\theta^{(0)}) \\ \tilde{I}_{2h}^h(\theta^{(1)}) \\ \tilde{I}_{2h}^h(\theta^{(2)}) \\ \tilde{I}_{2h}^h(\theta^{(3)}) \end{pmatrix} A_{2h}, \quad (3.42)$$

sendo $\tilde{I}_h^{2h}(\theta) := \frac{1}{4} \sum_{k \in K_P} t_k e^{-ik\theta}$ o símbolo do operador prolongação.

Portanto, o operador de prolongação I_{2h}^h é representado pela matriz 4×1 formada pelos termos $\tilde{I}_{2h}^h(\theta^{(\ell)})$, $l = 0, \dots, 3$ do lado direito de (3.42) e a denotamos por $\hat{I}_{2h}^h(\theta)$.

A prolongação mais natural é a interpolação linear entre os vizinhos, com stencil

$$\frac{1}{2} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}_{2h}^h, \quad (3.43)$$

sendo o símbolo correspondente $\tilde{I}_h^{2h}(\theta) = \frac{1}{4}[1 + \cos(\theta_1) + \cos(\theta_2) + \cos(\theta_2 - \theta_1)]$.

Unindo os resultados acima, temos o Teorema 3.7.1 de [Zhou e Fulton, 2009], a seguir.

Teorema 3.7.1. Suponha que os operadores discretos L_h e L_{2h} e os “transferidores” I_h^{2h} e I_{2h}^h são representados por stencils. Então para cada $\theta \in \Theta_{\text{low}} \setminus \Theta_0$ ($\Theta_0 = \{\theta \in \Theta_{\text{low}} : \tilde{L}_{2h}(2\theta) = 0\}$), o espaço E_h^θ é invariante sobre o operador correção na grade grossa K_h^{2h} , o qual é representado pela matriz 4×4

$$\widehat{K}_h^{2h}(\theta) = \widehat{I}_h - \widehat{I}_{2h}^h(\theta) \left(\widehat{L}_{2h}(2\theta) \right)^{-1} \widehat{I}_h^{2h}(\theta) \widehat{L}_h(\theta),$$

sendo $\widehat{I}_h$ a matriz identidade 4×4 . Além disso, se E_h^θ é invariante sobre o operador de suavização S_h , então ele é também invariante sobre o operador M_h^{2h} , o qual é representado pela matriz 4×4

$$\widehat{M}_h^{2h}(\theta) = \widehat{S}_h(\theta)^{\nu_2} \widehat{K}_h^{2h}(\theta) \widehat{S}_h(\theta)^{\nu_1}.$$

Os efeitos do ciclo em duas grades é então medido pelo *fator de convergência assintótico*

$$\rho_2(M_h^{2h}) = \sup\{\rho(\widehat{M}_h^{2h}(\theta)) : \theta \in \Theta_{\text{low}} \setminus \Theta_0\}$$

sendo $\rho(\cdot)$ o raio espectral. E o fator de convergência μ_2 obtido com a análise em duas malhas que é dado por

$$\mu_2 = \rho_2(M_h^{2h}). \quad (3.44)$$

Mostremos na Tabela 3.4 os valores ótimos para μ_2^J e μ_2^G obtidos ao utilizar o operador de restrição *full weighting*. Já na Tabela 3.5 são apresentados os valores ótimos para μ_2^J e μ_2^G utilizando *injeção* como operador de restrição. Em ambas tabelas representamos na primeira coluna os V-ciclos $V(\nu_1, \nu_2)$. Na segunda coluna mostramos o valor ótimo de μ_2^J e seu respectivo w e na última coluna o valor ótimo de μ_2^G e seu respectivo w .

Fatores de convergência analíticos				
Full weighting				
$V(\nu_1, \nu_2)$	μ_2^J ótimo	w	μ_2^G ótimo	w
V(1,1)	0.1600000	0.9	0.12621424	0.9
V(1,2)	0.09056206	0.9	0.07057802	1.0
V(2,1)	0.09056206	0.9	0.07057802	1.0
V(2,2)	0.06662572	0.8	0.04021900	1.1

Tabela 3.4: Valores ótimos para μ_2^J e μ_2^G usando *full weighting*.

Fatores de convergência analíticos				
Injeção				
$V(\nu_1, \nu_2)$	μ_2^J ótimo	w	μ_2^G ótimo	w
V(1,1)	0.22893599	0.9	0.18634964	1.0
V(1,2)	0.09764480	0.9	0.07296427	1.0
V(2,1)	0.09764480	0.9	0.07296427	1.0
V(2,2)	0.05308321	0.9	0.05142315	1.0

Tabela 3.5: Valores ótimos para μ_2^J e μ_2^G usando *injeção*.

Observemos na Tabela 3.4 que tanto para Jacobi quanto para Gauss-Seidel, utilizando *full weighting* o menor valor para o fator de suavização é obtido com $V(2, 2)$ ciclos em $w = 0.8$ para Jacobi e $w = 1.1$ para Gauss-Seidel. Utilizando *injeção* os valores ótimos também são obtidos com $V(2, 2)$ ciclos. Além disso, comparando os resultados obtidos por *full weighted* (Tabela 3.4) e *injeção* (Tabela 3.5) podemos observar que os menores valores dos fatores de convergência são obtidos por *full weighting*, exceto em $V(2, 2)$ para Jacobi. Dessa maneira, apresentaremos resultados numéricos para os esquemas de Jacobi e Gauss-Seidel utilizando *full weighting* como operador de restrição.

Agora vamos comparar esses valores de convergência obtidos pela Análise Local de Fourier com um fator μ_N que vai medir a redução do resíduo em cada ciclo do multigrid ao utilizar um algoritmo com repetidos $V(\nu_1, \nu_2)$ ciclos. Este fator μ_N é calculado por

$$\mu_N = \frac{\|\bar{r}_h\|}{\|\tilde{r}_h\|}, \quad (3.45)$$

sendo $\tilde{r}_h = f_h - L_h \tilde{u}_h$ o resíduo antes do V-ciclo e $\bar{r}_h = f_h - L_h \bar{u}_h$ o resíduo depois do V-ciclo. A norma $\|\cdot\|$ é a norma euclidiana usual, definida por

$$\|r_h\| = \left[\sum_j (r_j^h)^2 \right]^{1/2}. \quad (3.46)$$

3.8 Resultados numéricos

Os resultados que apresentaremos a seguir foram obtidos utilizando o seguinte problema do tipo Poisson:

$$-\nabla^2 u(\lambda, \theta) + u(\lambda, \theta) = f(\lambda, \theta), \quad (3.47)$$

sendo

$$f(\lambda, \theta) = \cos(\lambda) \cos^2(\theta) [-4 \sin^2(\theta) + 17 \cos^2(\theta) - 11], \quad (3.48)$$

com condição inicial dada pela seguinte função escalar, em coordenadas esféricas:

$$CI(\lambda, \theta) = \sin(35\theta) + \cos(10\lambda). \quad (3.49)$$

Ilustramos a função $CI(\lambda, \theta)$ na Figura 3.21, considerando o nível sete de resolução da malha.

Figura 3.21: Função trigonométrica dada em (3.49).

Mostramos na Figura 3.22 a solução exata para o problema dado em (3.47). Esta solução é dada pela equação

$$u(\lambda, \theta) = \cos(\lambda) \cos^4(\theta). \quad (3.50)$$

Figura 3.22: Função trigonométrica dada em (3.50).

Vamos verificar a eficiência do método multigrid implementado através da Análise Local de Fourier estudada na Seção 3.7. Para isso, vamos comparar os fatores de convergência obtidos ao utilizar o método multigrid aplicado à equação (3.47) com os fatores de convergência obtidos teoricamente para o problema idealizado em malhas hexagonais regulares.

Para um primeiro teste vamos utilizar como esquema de relaxação o método Weighted Jacobi e o operador de restrição *full weighting*. Para o operador de prolongamento, faremos uso da interpolação linear apresentada na Seção 3.5.1, deste capítulo. A seguir consideramos os mesmos operadores de restrição e interpolação, porém utilizaremos como esquema de relaxação o método de Gauss-Seidel.

Para cada um dos gráficos da Figura 3.23 e da Figura 3.24 mostramos, no eixo y , o comportamento dos fatores de convergência, μ_2 e μ_N e no eixo x representamos os valores de w . Na Figura 3.23 os fatores foram obtidos utilizando o método de Jacobi com relaxação. O fator numérico μ_N foi obtido após a realização de repetidos V-ciclos (entre 5 e 10 ciclos) para V(1, 1) (Figura 3.23a), V(1, 2) (Figura 3.23b), V(2, 1) (Figura 3.23c) e V(2, 2) (Figura 3.23d). Estes resultados foram obtidos considerando uma malha do nível cinco com 10242 pontos.

Já na Figura 3.24, novamente considerando o nível cinco de resolução da malha, mostramos o comportamento dos fatores de convergência μ_2 e μ_N utilizando Gauss-Seidel ao realizar repetidos $V(\nu_1, \nu_2)$ ciclos, considerando *full weighting* como o operador de restrição. Mostramos o fator numérico μ_N para V(1, 1) (Figura 3.24a), V(1, 2) (Figura 3.24b), V(2, 1) (Figura 3.24c) e V(2, 2) (Figura 3.24d).

(a) Fatores de convergência para $V(1,1)$. (b) Fatores de convergência para $V(1,2)$. (c) Fatores de convergência para $V(2,1)$. (d) Fatores de convergência para $V(2,2)$.**Figura 3.23:** Fatores de convergência para Jacobi e restrição full weighting.

(a) Fatores de convergência para $V(1,1)$. (b) Fatores de convergência para $V(1,2)$. (c) Fatores de convergência para $V(2,1)$. (d) Fatores de convergência para $V(2,2)$.**Figura 3.24:** Fatores de convergência para Gauss-Seidel e restrição full weighting.

Observemos na Figura 3.23 que todos os valores ótimos de μ_N são obtidos em $w = 0.9$. Além disso, o menor valor μ_N é obtido ao realizar $V(2,2)$ ciclos assim como o valor analítico ótimo μ_2 . Além disso, não há uma diferença significativa nos fatores ao utilizar $V(\nu_1, \nu_2)$, para $\nu_1 = 1, 2$ e $\nu_2 = 1, 2$.

Na Figura 3.24 observemos que os menores valores para o fator de convergência, tanto numérico quanto analítico encontram-se no intervalo de 0.8 a 1.1 realizando $V(2,2)$ ciclos. O valor ótimo $\mu_N = 0.0430832$ obtido em $w = 0.9$ é aproximadamente o valor ótimo analítico $\mu_2 = 0.040219$.

Agora vamos considerar valores para w de 0.8 a 1.0 e vejamos na Tabela 3.6 os valores para os fatores de convergência para os níveis de malha 5, 6, 7 e 8, considerando Jacobi e *full weighted*. E na Tabela 3.7 apresentamos os valores para os fatores, considerando Gauss-Seidel e *full weighted* para os níveis 5, 6, 7 e 8 de resolução da malha.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
WJ	$V(1, 1)$	0.8	0.21777777	0.25029112	0.26441015	0.27373996	0.27424629
		0.9	0.16000000	0.21524114	0.23105154	0.24394877	0.25309779
		1.0	0.23551392	0.38511353	0.38236475	0.38271178	0.39016447
	$V(1, 2)$	0.8	0.10566588	0.16883976	0.17843008	0.18377160	0.18377160
		0.9	0.09056206	0.13252771	0.14183274	0.15418461	0.16133998
		1.0	0.11667287	0.22157232	0.23121699	0.23438785	0.23598664
	$V(2, 1)$	0.8	0.10566588	0.15561607	0.16787377	0.18203776	0.18920957
		0.9	0.09056206	0.13566077	0.14107420	0.15454839	0.16165353
		1.0	0.11667287	0.22138209	0.22983952	0.23197645	0.23394368
	$V(2, 2)$	0.8	0.06662572	0.11805433	0.12440337	0.13682967	0.14385172
		0.9	0.07055050	0.10447857	0.11081349	0.12302361	0.12984159
		1.0	0.12307975	0.15713289	0.16412344	0.16541436	0.16822828

Tabela 3.6: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi e full weighting.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
GS	$V(1, 1)$	0.8	0.16178232	0.16045781	0.17670369	0.18172893	0.18589568
		0.9	0.12621424	0.11628591	0.11637128	0.12416537	0.13059983
		1.0	0.12878920	0.06653615	0.06986767	0.09070342	0.09275603
	$V(1, 2)$	0.8	0.08339880	0.09793637	0.10427082	0.11401839	0.12067874
		0.9	0.07407520	0.06743251	0.07461554	0.09577540	0.09023327
		1.0	0.07057802	0.06617240	0.05404952	0.05949616	0.06521385
	$V(2, 1)$	0.8	0.07407520	0.09844595	0.05404952	0.10834952	0.11540671
		0.9	0.07407520	0.06631253	0.07010827	0.07769222	0.08376625
		1.0	0.07057802	0.07428607	0.05124757	0.05353211	0.058446062
	$V(2, 2)$	0.8	0.06069380	0.06897435	0.07304600	0.08147104	0.08754891
		0.9	0.05242854	0.06053901	0.05509990	0.06024418	0.06582992
		1.0	0.04021998	0.08348051	0.04821251	0.04322829	0.04750076

Tabela 3.7: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel e full weighting.

Pelos resultados da Tabelas 3.6 e da Tabela 3.7 é possível observar a semelhança entre os valores de convergência analíticos obtidos a partir da Análise Local de Fourier e os valores obtidos numericamente. Portanto o comportamento desses valores mostram que os resultados estão condizentes com a teoria. Em ambos os métodos os valores ótimos para os fatores de convergência numéricos são obtidos ao realizar $V(2,2)$ ciclos. Porém, podemos observar que os resultados para estes fatores de convergência, tanto para Jacobi quanto para Gauss-Seidel variam pouco ao realizar os diferentes ciclos $V(1,1)$, $V(1,2)$, $V(2,1)$ e $V(2,2)$. Dessa maneira, se torna mais eficiente utilizar $V(1,1)$ ciclos pelo fato do custo computacional ser menor ao realizar as iterações do método. A melhor escolha é

do uso de Gauss-Seidel (sem relaxação) no ciclo V(1,1), com taxas de convergência abaixo de 0.1, independentemente da malha.

Na Tabela 3.8 apresentamos os fatores de convergência para o método de Jacobi utilizando *injeção* e considerando valores de w entre 0.8 e 1.0. Já na Tabela 3.9 apresentamos os valores para os fatores de convergência considerando novamente *injeção* mas utilizando o método de Gauss-Seidel. Para ambos os testes consideremos os níveis 5, 6, 7 e 8 da malha icosaédrica.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
WJ	V(1, 1)	0.8	0.21777777	0.34636957	0.35709760	0.35791140	0.35101064
		0.9	0.22893599	0.28021193	0.36062402	0.44635546	0.46001396
		1.0	0.39252854	0.44692218	0.57491842	0.65396931	0.67152570
	V(1, 2)	0.8	0.10162962	0.15342853	0.15967543	0.16307989	0.16502028
		0.9	0.09764480	0.17291986	0.20155312	0.22533152	0.22568036
		1.0	0.21291331	0.32644363	0.37792383	0.38344078	0.36760687
	V(2, 1)	0.8	0.10162962	0.17472280	0.18510054	0.19389311	0.20188999
		0.9	0.09764480	0.17193402	0.19725024	0.21587714	0.23303248
		1.0	0.21291331	0.33755383	0.40625038	0.44965798	0.44072369
	V(2, 2)	0.8	0.05742716	0.09335755	0.09851157	0.07258579	0.07023871
		0.9	0.05308321	0.10599735	0.12889937	0.15546346	0.17074870
		1.0	0.12307975	0.22839272	0.25990259	0.29725000	0.30453885

Tabela 3.8: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi e *injeção*.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
GS	V(1, 1)	0.8	0.22162162	0.22865362	0.25232204	0.26700834	0.27876977
		0.9	0.18769230	0.25051962	0.28109200	0.31887143	0.32683882
		1.0	0.18634964	0.39303860	0.42081595	0.43811649	0.44300294
	V(1, 2)	0.8	0.10433215	0.10965361	0.11353216	0.11162665	0.12333438
		0.9	0.08131482	0.21137635	0.22896515	0.22774099	0.23945416
		1.0	0.07296427	0.43897486	0.46520450	0.45941246	0.47781006
	V(2, 1)	0.8	0.10433215	0.20240561	0.21251582	0.21762464	0.23177925
		0.9	0.08131482	0.25745506	0.27235006	0.28612270	0.29629411
		1.0	0.07296427	0.40644091	0.42696651	0.44930025	0.46947380
	V(2, 2)	0.8	0.04911614	0.14198709	0.14214987	0.14256982	0.14300959
		0.9	0.05522840	0.18393713	0.18463859	0.18579975	0.18679567
		1.0	0.05142315	0.27138807	0.27230182	0.27454688	0.27612414

Tabela 3.9: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel e *injeção*.

Notamos que os resultados com o operador de *injeção* são significativamente inferiores aos obtidos com o *full weighting*, não se justificando a ligeira economia computacional desta escolha. A combinação de Gauss-Seidel com *full weighting* leva aos melhores resultados.

A partir dos resultados apresentados, observemos que uma boa escolha para a resolução do problema apresentado em (3.47) é obtido utilizando o método de Gauss-Seidel, considerando *full weighting* como operador de restrição e $V(1,1)$ ciclos com $w=1.0$. Desse modo, mostramos na Tabela 3.10 na primeira coluna o número de recursão da malha e na segunda o número de nós. Já na terceira e quarta colunas, aparecem os erros E_2 e E_∞ respectivamente, obtidos através das equações dadas em (3.6).

Nível q	Nós(N_q)	Erro (E_2)	Erro (E_∞)
1	42	0.03894846395949859	0.087316437248675771
2	162	0.00908761156454505	0.022831572375772602
3	642	0.00222932433973176	0.005682488509134664
4	2562	0.00055466698263209	0.001442096361345734
5	10242	0.00013839358095530	0.000368461234086403
6	40962	0.00003452119311532	0.000097342623618157
7	163842	0.00000926762962937	0.000028960861851112
8	655362	0.00000216311799396	0.000006068779364154

Tabela 3.10: Erros na estimativa da solução para o problema dado em (3.47) utilizando o método multigrid, considerando Gauss-Seidel, *full-weighting* e $V(1,1)$ ciclos para $w = 1.0$.

Observa-se com os resultados da Tabela 3.10 que os erros diminuem a medida que aumentamos o nível de resolução da malha.

A Figura 3.25 mostra o erro calculado pelo valor absoluto da diferença entre o valor exato em cada ponto da malha, obtido utilizando a equação (3.50) e o valor aproximado pelo método multigrid, considerando Gauss-Seidel, *full-weighting* e $V(1,1)$ ciclos para $w = 1.0$. Observemos a semelhança dos resultados apresentados na Tabela 3.10, acima. Os erros diminuem ao considerar malhas mais finas.

Figura 3.25: Erro associado à aproximação da solução através do método multigrid para o problema dado em (3.47). Foram utilizados Gauss-Seidel para o esquema de relaxação, full weighting como operador de restrição e $V(1, 1)$ ciclos.

3.8.1 Testes com malhas otimizadas

Vamos analisar agora o comportamento dos fatores de convergência ao utilizar otimizações para a malha icosaédrica. Faremos os testes, inicialmente para a otimização chamada SCVT, do inglês, *Spherical Centroidal Voronoi Tessellation*. A ideia desta otimização é ajustar a malha para que os nós geradores das células de Voronoi sejam os centros de massa das células de Voronoi. A seguir utilizaremos a otimização HR95, proposta por [Heikes e Randall, 1995a]. Os autores utilizaram esta otimização para sanar o problema da inconsistência da discretização do laplaciano na malha icosaédrica (ver Seção 2.3 do Capítulo 2). O objetivo desta otimização é ajustar os nós fazendo com que os pontos de intersecção da malha primal e do diagrama de Voronoi sejam mais próximos possíveis do ponto médio da aresta que liga os verticés das células da malha icosaédrica.

Resultados e detalhes sobre estes dois tipos de otimizações podem ser vistos em [Peixoto, 2013].

Na Tabela 3.11 apresentamos os fatores de convergência utilizando o método de Jacobi como esquema de relaxação, full weighting como operador de restrição e a otimização SCVT. Já na Tabela 3.12 também apresentamos os fatores de convergência considerando Jacobi e full weighting, porém utilizamos HR95 para otimização. Nas Tabelas 3.13 e 3.14 foram considerados Gauss-Seidel, full weighting e respectivamente as otimizações SCVT e HR95.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
WJ	$V(1, 1)$	0.8	0.21777777	0.24452860	0.25659992	0.27178548	0.26263803
		0.9	0.16000000	0.19409093	0.21240586	0.21424719	0.21146531
		1.0	0.23551392	0.26694597	0.26306112	0.27274758	0.28252286
	$V(1, 2)$	0.8	0.10566588	0.15274145	0.16292316	0.16341072	0.16759534
		0.9	0.09056206	0.12004281	0.13217946	0.13378841	0.14087708
		1.0	0.11667287	0.15130788	0.14477050	0.14385326	0.15487533
	$V(2, 1)$	0.8	0.10566588	0.14840734	0.16220794	0.18144531	0.18210225
		0.9	0.09056206	0.12216047	0.13094450	0.14194037	0.14042756
		1.0	0.11667287	0.15250906	0.14282089	0.15378656	0.14563832
	$V(2, 2)$	0.8	0.06662572	0.10092446	0.11082438	0.12137692	0.12027567
		0.9	0.07055050	0.09235273	0.09899444	0.10225175	0.11069763
		1.0	0.12307975	0.09197872	0.09172352	0.10042611	0.10794536

Tabela 3.11: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi, full weighting e a otimização SCVT.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
WJ	$V(1, 1)$	0.8	0.21777777	0.29050674	0.30392089	0.31200953	0.31053955
		0.9	0.16000000	0.23082820	0.24148395	0.24335764	0.25092497
		1.0	0.23551392	0.3565460	0.36229482	0.36187688	0.38094212
	$V(1, 2)$	0.8	0.10566588	0.17653879	0.19094800	0.19360450	0.19167088
		0.9	0.09056206	0.14498281	0.15136887	0.15351056	0.16012110
		1.0	0.11667287	0.20482341	0.22079094	0.22056199	0.23167865
	$V(2, 1)$	0.8	0.10566588	0.17172011	0.19144125	0.20227847	0.21079771
		0.9	0.09056206	0.14238897	0.15172129	0.16181848	0.17003459
		1.0	0.11667287	0.20522697	0.22068667	0.22443503	0.22963983
	$V(2, 2)$	0.8	0.06662572	0.14318536	0.13123385	0.14097775	0.14041304
		0.9	0.07055050	0.13692702	0.11100450	0.12031784	0.12209071
		1.0	0.12307975	0.14121498	0.15261240	0.15093656	0.16082614

Tabela 3.12: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Jacobi, full weighting e a otimização HR95.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
GS	V(1, 1)	0.8	0.16178232	0.15033985	0.16763019	0.17633779	0.17249645
		0.9	0.12621424	0.08507477	0.09837020	0.11038073	0.11206194
		1.0	0.12878920	0.04834692	0.05782259	0.06206841	0.06781162
	V(1, 2)	0.8	0.08339880	0.08862131	0.09175726	0.09798302	0.09916400
		0.9	0.07407520	0.06023420	0.06296592	0.06922056	0.07335472
		1.0	0.07057802	0.04043061	0.04253691	0.04684801	0.05035777
	V(2, 1)	0.8	0.08339880	0.08243353	0.09011115	0.09945245	0.10336366
		0.9	0.07407520	0.05691598	0.05840227	0.06575482	0.06939937
		1.0	0.07057802	0.03683144	0.03833545	0.04252187	0.04524609
	V(2, 2)	0.8	0.06069380	0.05955018	0.06066478	0.06822139	0.07087341
		0.9	0.05242854	0.04312992	0.04338936	0.04906952	0.05184797
		1.0	0.04021998	0.02854402	0.02911435	0.03241574	0.03496456

Tabela 3.13: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel, full weighting e a otimização SCVT.

Relaxação	Ciclos	w	Analítico	Numérico			
			μ_2	Nível 5	Nível 6	Nível 7	Nível 8
GS	V(1, 1)	0.8	0.16178232	0.15670773	0.17903694	0.17449827	0.18492601
		0.9	0.12621424	0.09360230	0.11176636	0.11853958	0.12432230
		1.0	0.12878920	0.05997152	0.06515085	0.0695888	0.076888068
	V(1, 2)	0.8	0.08339880	0.08568421	0.10029162	0.10230695	0.10952320
		0.9	0.07407520	0.06676835	0.06573553	0.07397933	0.08027891
		1.0	0.07057802	0.05192579	0.0432587	0.04989028	0.05572976
	V(2, 1)	0.8	0.08339880	0.08981398	0.09861185	0.10399966	0.11564196
		0.9	0.07407520	0.08070947	0.06456718	0.07259296	0.07902550
		1.0	0.07057802	0.07531558	0.04692812	0.04698112	0.05213298
	V(2, 2)	0.8	0.06069380	0.063552563	0.06273674	0.07381630	0.07943034
		0.9	0.05242854	0.053332950	0.04530149	0.05311965	0.05862937
		1.0	0.04021998	0.042252524	0.03044675	0.03603399	0.04075875

Tabela 3.14: Fatores de convergência analíticos e numéricos para w entre 0.8 e 1.0. Foram considerados Gauss-Seidel, full weighting e a otimização HR95.

Observemos que tanto para Gauss-Seidel quanto para Jacobi há uma melhora nos fatores de convergência para ambas otimizações. Considerando a otimização SCVT esta melhora é maior do que ao utilizar HR95, principalmente para o método de Jacobi.

Capítulo 4

Conclusões

Ao longo dos últimos anos, diferentes malhas têm sido analisadas para uso em modelos atmosféricos globais. Tem-se buscado malhas isotrópicas, também com a motivação de eficiência no paralelismo dos modelos.

Uma alternativa para gerar malhas aproximadamente isotrópicas é o uso de sólidos platônicos. Estes são inscritos na esfera e suas faces projetadas na superfície, dando origem a uma malha geodésica. Uma das mais populares, gerada a partir do icosaedro, é a malha esférica geodésica icosaédrica, escolhida para o presente estudo.

Optamos por discretizar o operador de Laplace na malha icosaédrica por volumes finitos como usualmente proposto na literatura. A discretização, como verificamos em testes numéricos não é uniformemente consistente. No entanto, comprovamos numericamente que quando esta discretização é utilizada para aproximar a solução da equação de Poisson através do método de Jacobi ou Gauss-Seidel obtém-se convergência (de segunda ordem). Tal fato ainda está por merecer uma análise teórica mais aprofundada.

A fim de obter um método eficiente para a solução das equações elípticas em malhas icosaédricas, estudamos três métodos multigrid propostos na literatura. Para estimar a sua eficiência e comportamento realizamos um estudo baseado na Análise Local de Fourier (em malhas hexagonais planas), através do qual foram obtidas as taxas idealizadas de convergência analíticas, com os métodos de Jacobi e Gauss-Seidel como esquemas de relaxação para o multigrid. Foram realizados testes numéricos para comparar as taxas de convergência obtidas por esta análise e as taxas obtidas numericamente, utilizando ambos os esquemas de relaxação. Estes testes comprovaram que o método de Gauss-Seidel apresenta melhor performance para o multigrid, ratificando o que é apresentado pela teoria. Além disso, também apresentamos teste numéricos utilizando dois tipos de otimizações para a malha icosaédrica e verificamos que os fatores de convergência são melhores para ambos os métodos de Jacobi e Gauss-Seidel ao utilizar estas otimizações.

Os resultados obtidos mostram a eficiência do método multigrid para a resolução de problemas que envolvem o operador de Laplace em malhas icosaédricas. Dessa forma, é pertinente considerar em trabalhos futuros a utilização do algoritmo desenvolvido neste trabalho para a resolução de problemas que envolvem equações elípticas, como por exemplo, para a solução das equações de águas rasas .

Apêndice A

Definições e resolução de exemplos

A.1 Definições

Definição A.1.1. (Projeção Radial). Denotamos por projeção radial de um ponto Q da esfera sobre um plano tangente à esfera em um certo ponto P , o ponto de intersecção da semi-reta que parte do centro da esfera, e passa pelo ponto Q , com o plano tangente a esfera em P .

Definição A.1.2. (Polígono Alinhado). Um polígono é alinhado se tiver lados opostos paralelos e ainda os comprimentos de qualquer aresta for igual ao comprimento de sua aresta oposta.

Definição A.1.3. (Polígono Esférico Alinhado). Polígono esférico alinhado é um polígono esférico tal que sua projeção radial no plano tangente à esfera no seu centro de massa seja um polígono alinhado neste plano.

A.2 Resolução dos Exemplos

A.2.1 Resolução Exemplo 3.7.1

Sabemos que a aproximação discreta para o laplaciano ∇^2 de uma função u em um ponto P_0 é dada por

$$L_h u_0 = \frac{1}{A_c} \sum_{i=1}^n \frac{l_i}{L_i} (u_i - u_0). \quad (\text{A.1})$$

Como estamos trabalhando com células hexagonais uniformes e com stencil

$$\frac{2}{3h^2} \begin{bmatrix} -1 & -1 & 0 \\ -1 & 6 & -1 \\ 0 & -1 & -1 \end{bmatrix}_h$$

A_c = área do hexágono é dada por $A_{hex} = \frac{3\sqrt{3}}{2}h^2$. Além disso, $l_i = 1$ e $L_i = \sqrt{3}$. Assim, $-\nabla^2$ pode

ser discretizado usando sete pontos da forma

$$L_h u_0 = \frac{2}{3h^2} \left(6u_0 - \sum_{i=1}^6 u_i \right), \quad (\text{A.2})$$

logo

$$L_h \varphi_h(\theta, x) = \tilde{L}_h(\theta) := \sum_{k \in K} s_k e^{i(k_1 \theta_1 + k_2 \theta_2)}$$

fica da forma

$$\tilde{L}_h(\theta) = \frac{2}{3h^2} \left(6 - e^{i\theta_1} - e^{i\theta_2} - e^{i(-\theta_1 + \theta_2)} - e^{i(-\theta_1)} - e^{i(-\theta_2)} - e^{i(\theta_1 - \theta_2)} \right). \quad (\text{A.3})$$

Escrevendo as exponencias na forma polar obtemos

$$\begin{aligned} e^{i\theta_1} &= \cos(\theta_1) + i \sin(\theta_1) \\ e^{i\theta_2} &= \cos(\theta_2) + i \sin(\theta_2) \\ e^{-i\theta_1} &= \cos(\theta_1) - i \sin(\theta_1) \\ e^{-i\theta_2} &= \cos(\theta_2) - i \sin(\theta_2) \\ e^{i(-\theta_1 + \theta_2)} &= \cos(\theta_2 - \theta_1) + i \sin(\theta_2 - \theta_1) \\ e^{i(\theta_1 - \theta_2)} &= e^{-i(\theta_2 - \theta_1)} \cos(\theta_2 - \theta_1) - i \sin(\theta_2 - \theta_1) \end{aligned}$$

Substituindo as equações acima em A.2.1 temos

$$\tilde{L}_h(\theta) = \frac{2}{3h^2} [6 - 2 \cos(\theta_1) - 2 \cos(\theta_2) - 2 \cos(\theta_2 - \theta_1)],$$

$$\tilde{L}_h(\theta) = \frac{4}{3h^2} [3 - \cos(\theta_1) - \cos(\theta_2) - \cos(\theta_2 - \theta_1)].$$

A.2.2 Resolução Exemplo 3.7.4

Temos por definição $\mu_1(S_h^J) := \sup \left\{ \left| \tilde{S}_h^J(\theta) \right| : \theta \in \Theta_{\text{high}} \right\}$. Queremos verificar que

$$\mu_1(S_h^J(\omega)) = \max \left\{ 1 - \omega \left(1 - \frac{\sqrt{2}}{3} \right), \frac{3\omega}{2} - 1 \right\}. \quad (\text{A.4})$$

Como w é constante, consideremos $\tilde{S}_h^J(\theta, \omega) = S(\theta_1, \theta_2)$. Temos

$$S(\theta_1, \theta_2) = 1 - \frac{2\omega}{3} \left[\sin^2 \left(\frac{\theta_1}{2} \right) + \sin^2 \left(\frac{\theta_2}{2} \right) + \sin^2 \left(\frac{\theta_1 - \theta_2}{2} \right) \right]. \quad (\text{A.5})$$

A região Θ_{high} das altas frequências é o quadrado $-\pi \leq \theta_1 < \pi$, $-\pi \leq \theta_2 < \pi$ com o quadrado central $-\frac{\pi}{2} < \theta_1 \leq \frac{\pi}{2}$, $-\frac{\pi}{2} < \theta_2 \leq \frac{\pi}{2}$ removido, como mostra a Figura 3.16. Assim $S(\theta_1, \theta_2)$ é suave, logo o mínimo e o máximo ocorrem onde as derivadas parciais são iguais a zero, ou ao longo da fronteira da região.

1º Caso: Análise no interior da região Θ_{high} .

Igualando a zero as derivadas parciais de S em relação a θ_1 e θ_2 .

$$\frac{\partial S(\theta_1, \theta_2)}{\partial \theta_1} = 0 \Rightarrow \frac{\partial S(\theta_1, \theta_2)}{\partial \theta_1} = -\frac{w}{3} [\sin(\theta_1) + \sin(\theta_1 - \theta_2)]$$

$$\frac{\partial S(\theta_1, \theta_2)}{\partial \theta_2} = 0 \Rightarrow \frac{\partial S(\theta_1, \theta_2)}{\partial \theta_1} = -\frac{w}{3} [\sin(\theta_2) - \sin(\theta_1 - \theta_2)]$$

Agora vamos determinar os pontos críticos, já que precisamos calcular o supremo de $S(\theta_1, \theta_2)$.

$$\begin{cases} -\sin(\theta_1) = \sin(\theta_1 - \theta_2) \\ \sin(\theta_2) = \sin(\theta_1 - \theta_2) \end{cases}. \quad (\text{A.6})$$

Logo precisamos obter as soluções de $\sin(\theta_1) + \sin(\theta_2) = 0$. Temos duas soluções para esta equação:

1) $\theta_2 = \theta_1 - \pi$. Substituindo θ_2 na segunda equação de A.6 temos

$$\sin(\theta_2) - \sin(\pi) = 0 \Rightarrow \sin(\theta_2) = 0 \Rightarrow \theta_2 = 0 \text{ ou } \theta_2 = -\pi, \text{ já que } \pi \text{ não pertence ao domínio.}$$

Neste caso temos os seguintes pontos críticos $(\pi, 0)$ e $(0, \pi)$, porém, ambos não pertencem ao domínio.

2) $\theta_2 = -\theta_1$. Substituindo θ_2 na primeira equação de A.6 obtemos

$$\sin(\theta_1) + \sin(2\theta_1) = 0 \Rightarrow \sin(\theta_1) + 2\cos(\theta_1)\sin(\theta_1) = 0 \Rightarrow \sin(\theta_1)(1 + 2\cos(\theta_1)) = 0.$$

$$* \sin(\theta_1) = 0 \Rightarrow \theta_1 = 0 \text{ ou } \theta_1 = \pm\pi, \text{ ambos valores para } \theta_1 \text{ não pertencem ao domínio.}$$

$$* 1 + 2\cos(\theta_1) = 0 \Rightarrow \theta_1 = \pm\frac{2\pi}{3}.$$

Logo os pontos críticos são $(\frac{2\pi}{3}, -\frac{2\pi}{3})$ e $(-\frac{2\pi}{3}, \frac{2\pi}{3})$. Vamos verificar se esses pontos são de máximo. Para isso calculemos

$$H(\theta_1, \theta_2) = \begin{bmatrix} \frac{\partial^2 S(\theta_1, \theta_2)}{\partial \theta_1^2} & \frac{\partial^2 S(\theta_1, \theta_2)}{\partial \theta_1 \partial \theta_2} \\ \frac{\partial^2 S(\theta_1, \theta_2)}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 S(\theta_1, \theta_2)}{\partial \theta_2^2} \end{bmatrix} = \begin{bmatrix} -\frac{w}{3} [\cos(\theta_1) + \cos(\theta_1 - \theta_2)] & \frac{w}{3} [\cos(\theta_1 - \theta_2)] \\ -\frac{w}{3} [-\cos(\theta_1 - \theta_2)] & -\frac{w}{3} [\cos(\theta_2) + \cos(\theta_1 - \theta_2)] \end{bmatrix}.$$

Para que um ponto crítico seja ponto de máximo precisamos que o determinante de $H(\theta_1, \theta_2)$ seja maior que zero e além disso que $\frac{\partial^2 S(\theta_1, \theta_2)}{\partial \theta_1^2} < 0$.

Temos que $\det(H(\theta_1, \theta_2)) = \left[-\frac{w}{3} \cos(\theta_1) + \cos(\theta_1 - \theta_2)\right]^2 > 0$. Além disso, no ponto $(\frac{2\pi}{3}, -\frac{2\pi}{3})$ temos $\frac{\partial^2 S(\frac{2\pi}{3}, -\frac{2\pi}{3})}{\partial \theta_1^2} = -\frac{w}{6} < 0$. Portanto $(\frac{2\pi}{3}, -\frac{2\pi}{3})$ é um ponto de máximo e assim substituindo este ponto em A.5 obtemos

$$S\left(\frac{2\pi}{3}, -\frac{2\pi}{3}\right) = 1 - \frac{2w}{3} \left[\sin^2\left(\frac{\pi}{3}\right) + \sin^2\left(\frac{\pi}{3}\right) + \sin^2\left(\frac{2\pi}{3}\right) \right].$$

$$S\left(\frac{2\pi}{3}, -\frac{2\pi}{3}\right) = 1 - \frac{2\omega}{3} \left(\frac{9}{4}\right).$$

Logo $S\left(\frac{2\pi}{3}, -\frac{2\pi}{3}\right) = 1 - \frac{3w}{2}$ que é o segundo valor de A.4. Para o ponto $(-\frac{2\pi}{3}, \frac{2\pi}{3})$ o processo é análogo e o resultado é o mesmo.

2º Caso: Análise ao longo da fronteira do quadrado interior.

Ao longo da fronteira, $S(\theta_1, \theta_2)$ torna-se uma função de uma variável (em função apenas de θ_1 ou θ_2 dependendo da fronteira).

Pontos críticos: $(\frac{\pi}{2}, \frac{\pi}{4})$ e $(-\frac{\pi}{2}, -\frac{\pi}{4})$.

Substituindo $(\frac{\pi}{2}, \frac{\pi}{4})$ em A.5 obtemos

$$S\left(\frac{\pi}{2}, \frac{\pi}{4}\right) = 1 - \frac{2\omega}{3} \left[\sin^2\left(\frac{\pi}{4}\right) + \sin^2\left(\frac{\pi}{8}\right) + \sin^2\left(\frac{\pi}{8}\right) \right] = 1 - \frac{2\omega}{3} \left[\sin^2\left(\frac{\pi}{4}\right) + 2\sin^2\left(\frac{\pi}{8}\right) \right].$$

Usando a relação trigonométrica

$$2 \sin^2\left(\frac{x}{2}\right) = 1 - \cos(x)$$

escrevemos

$$S\left(\frac{\pi}{2}, \frac{\pi}{4}\right) = 1 - \frac{2\omega}{3} \left[\sin^2\left(\frac{\pi}{4}\right) + 1 - \cos\left(\frac{\pi}{4}\right) \right] = 1 - \frac{2\omega}{3} \left[\left(\frac{\sqrt{2}}{2}\right)^2 + 1 - \left(\frac{\sqrt{2}}{2}\right) \right].$$

Logo $S\left(\frac{\pi}{2}, \frac{\pi}{4}\right) = 1 - w \left(1 - \frac{\sqrt{2}}{3}\right)$ que equivale ao primeiro termo de A.4. Para o ponto $(-\frac{\pi}{2}, -\frac{\pi}{4})$ o processo é análogo e obtém-se o mesmo resultado.

Referências Bibliográficas

- Aurenhammer e Klein(2000)** F. Aurenhammer e R. Klein. *Voronoi Diagrams*. Elsevier Science, 1ª edição. Citado na pág. [6](#)
- Barbeiro(2009)** S. Barbeiro. Supraconvergent cell-centered scheme for two dimensional elliptic problems. *Appl. Numer. Math.*, 59:56–72. Citado na pág. [19](#)
- Barros(1991)** S.R.M. Barros. Multigrid methods for two and three dimensional poisson-type equations on the sphere. *Journal of Computational Physics*, 92:313–348. Citado na pág. [27](#)
- Burden e Faires(2010)** R. L. Burden e J. D. Faires. *Numerical Analysis*. Brooks/Cole Publishing Company, 9ª edição. Citado na pág. [2](#)
- Delaunay(1934)** B. Delaunay. Sur la sphère vide. *Bull. Acad. Sci. USSR*, 7:793–800. Citado na pág. [5](#)
- Du et al.(2003)** Qiang Du, Max D. Gunzburger e Lili Ju. Constrained centroidal voronoi tessellations for surfaces. *SIAM J. Sci. Comput.*, 24:1488–1506. Citado na pág. [14](#)
- Fedorenko(1964)** R. P. Fedorenko. The speed of convergence of an iteration process. *USSR Comput. Math. And Math. Phys*, 4:227–235. Citado na pág. [2](#)
- Hackbusch(2010)** Wolfgang Hackbusch. *Elliptic Differential Equations. Theory and Numerical Treatment*. Springer, 2ª edição. Citado na pág. [22](#)
- Heikes e Randall(1995a)** Ross Heikes e David A. Randall. Numerical integration of the shallow-water equations on a twisted icosahedral grid. *American Meteorological Society*, 123:1862–1880. Citado na pág. [vii](#), [2](#), [5](#), [11](#), [34](#), [36](#), [62](#)
- Heikes e Randall(1995b)** Ross Heikes e David A. Randall. Numerical integration of the shallow-water equations on a twisted icosahedral grid. part ii: A detailed description of the grid and an analysis of numerical accuracy. *American Meteorological Society*, 123:1881–1887. Citado na pág. [16](#), [18](#)
- Heikes et al.(2013)** Ross Heikes, David A. Randall e Celal S. Konor. Optimized icosahedral grids: Performance of finite-difference operators and multigrid solver. *American Meteorological Society*, 141:4450–4469. Citado na pág. [vii](#), [2](#), [34](#), [39](#)
- Kaplan(2002)** Wilfred Kaplan. *Advanced Calculus*. Pearson, 5ª edição. Citado na pág. [13](#), [17](#)

- Peixoto(2013)** Pedro S. Peixoto. *Análise de discretizações e interpolações em malhas icosaédricas e aplicações em modelos de transporte semi-lagrangianos*. Tese de Doutorado, Instituto de Matemática e Estatística, Universidade de São Paulo, Brasil. Citado na pág. [2](#), [11](#), [13](#), [14](#), [62](#)
- Peixoto e Barros(2013)** Pedro S. Peixoto e Saulo R. M. Barros. Analysis of grid imprinting on geodesical spherical icosahedral grids. *Journal of Computational Physics*, 237:61–78. Citado na pág. [12](#), [13](#), [14](#), [15](#)
- Preparata e Shamos(1985)** F. P. Preparata e M. I. Shamos. *Computational Geometry, an Introduction*. Springer-Verlag. Citado na pág. [7](#), [8](#)
- Renka(1997)** Robert J. Renka. Algorithm 772: Stripack: Delaunay triangulation and voronoi diagram on the surface of a sphere. *ACM Transactions on Mathematical Software*, 23:416–434. Citado na pág. [5](#), [9](#)
- Sadourny et al.(1968)** R. Sadourny, A. Arakawa e Y. Mintz. Integration of the nondivergent barotropic vorticity equation with an icosahedral-hexagonal grid for the sphere. *Mon. Wea. Rev.*, páginas 351–356. Citado na pág. [1](#)
- Staniforth e Thuburn(2002)** Andrew Staniforth e John Thuburn. Horizontal grids for global weather and climate prediction models: a review. *Q. J. R. Meteorol. Soc.*, 138:1–26. Citado na pág. [1](#)
- Thuburn(1997)** John Thuburn. A pv-based shallow-water model on a hexagonal-icosahedral grid. *American Meteorological Society*, 125:2328–2347. Citado na pág. [vii](#), [2](#), [34](#), [36](#), [38](#)
- Trottenberg et al.(2001)** Ulrich Trottenberg, Cornelis Oosterlee e Anton Schuller. *Multigrid*. Academic Press. Citado na pág. [2](#), [21](#), [41](#)
- Voronoi(1908)** G. Voronoi. Nouvelles applications des parametres continus à la theorie des formes qudratiques: Deuxième mémoire: Recherches sur les paralléloèdres primitifs. *J. Reine Angew Math.*, 134:198–287. Citado na pág. [5](#)
- Williamson(1968)** David L. Williamson. Integration of the barotropic vorticity equation on aspherical geodesic grid. *Tellus XX*, 20:642–653. Citado na pág. [1](#), [5](#)
- Wolfram(1988)** S. Wolfram. *A System for Doing Mathematics by Computer*. Addison-Wesley, 1ª edição. Citado na pág. [36](#)
- Zhou e Fulton(2009)** Guohua Zhou e Scoot R. Fulton. Fourier analysis of multigrid methods on hexagonal grids. *Society for Industrial and Applied Mathematics*, 31:1518–1538. Citado na pág. [2](#), [3](#), [22](#), [33](#), [41](#), [43](#), [47](#), [51](#), [52](#)
- Øyvind Hjelle e Dæhlen(2006)** Øyvind Hjelle e Morten Dæhlen. *Triangulations and Applications*. Springer. Citado na pág. [7](#), [8](#)