

**Métodos de refinamento local em  
modelos globais de previsão  
numérica do tempo**

**Claudia Inés Garcia**

TESE APRESENTADA  
AO  
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA  
DA  
UNIVERSIDADE DE SÃO PAULO  
PARA  
OBTENÇÃO DO GRAU DE DOUTOR  
EM  
MATEMÁTICA APLICADA

Área de Concentração: **Análise Numérica**  
Orientador: **Prof. Dr. Saulo Rabello Maciel de Barros**

– São Paulo, março de 2001 –

# Métodos de refinamento local em modelos globais de previsão numérica do tempo

Este exemplar corresponde à redação  
final da tese devidamente corrigida  
e defendida por Claudia Inés Garcia  
e aprovada pela comissão julgadora.

São Paulo, 3 de maio de 2001.

Banca examinadora:

- Prof. Dr. Saulo Rabello Maciel de Barros (orientador) (IME-USP)
- Prof. Dr. Nelson Mugayar Kuhl (IME-USP)
- Prof. Dr. Pedro Leite da Silva Dias (IAG-USP)
- Prof. Dr. Julio Romano Meneghini (EP-USP)
- Prof. Dr. André Nachbin (IMPA)

## Agradecimentos

Meu agradecimento mais profundo a todas aquelas pessoas que fizeram possível a conclusão desta tese.

Quero agradecer ao CNPQ e à FAPESP pelas bolsas recebidas.

Ao pessoal do Departamento de Matemática Aplicada, da Secretaria da CPG e da Biblioteca.

Ao pessoal do IAG, em especial ao professor Pedro pelas sugestões e incentivo.

Ao Conselho Diretor do LCCA em particular ao Francisco pela ajuda inestimável que significou para nosso trabalho o uso dos recursos do LCCA.

Aos professores Alexandre, Nelson e Luis Carlos pelas sugestões e comentários para a realização deste trabalho.

Um agradecimento muito especial a meu orientador Saulo, pela amizade, conselhos, competência e sugestões fundamentais durante a elaboração desta tese.

À minha ex-orientadora Cristina pelo apoio e carinho.

Aos inúmeros amigos que ganhei durante estes anos, peruanos, colombianos, chilenos, brasileiros... e que me ajudaram a tornar esta jornada muito mais agradável. Foram tantas as mãos amigas que fica difícil citar nomes.

A meus colegas de análise numérica, em particular ao Raul e à Alice pelo companheirismo, e a Walter pela paciência infinita.

Aos meus amigos argentinos Carolina, Hector, Karina, Marcel, Miguel, Susana, Viviana, Walter e Washi.

A meus queridos amigos Viviana e Daniel, sempre generosos, com quem compartilhei momentos muito importantes. A Luana pela alegria.

A meu querido amigo Juan, pelo companheirismo, alegria e ajuda nos momentos mais difíceis.

A Laura, Maria de los Angeles e Emilio pela força durante estes anos.

A meus pais Lucy e Jesús e meus irmãos pelo apoio e carinho constante. A meu irmão Jesús pela ajuda e orientação quando cheguei ao Brasil.

A Juan Carlos, pelo carinho.

## Resumo

O objetivo central deste trabalho foi demonstrar a viabilidade de uso de técnicas de refinamento local para a obtenção de alta resolução em uma região de interesse, empregando um modelo global de previsão numérica do tempo. Desta forma visa-se atingir precisões locais que seriam obtidas com o uso de malhas uniformemente finas nos modelos globais, mas a um custo computacional comparável ao de modelos locais de previsão. A vantagem de trabalhar com um modelo global é evitar a introdução de condições artificiais de fronteira e ainda ter um modelo único para previsões locais ou globais. Nosso estudo foi baseado em uma hierarquia de modelos, iniciando por um simples modelo de advecção na esfera, passando por um modelo de vorticidade barotrópica até chegarmos a um modelo global para as equações de água-rasa. Nosso desenvolvimento foi centrado no uso de métodos semi-Lagrangeanos de dois níveis no tempo, dando uma unidade ao tratamento dos modelos, que foram incorporando novos aspectos com seus crescentes graus de dificuldade (interpolações, cálculos de trajetórias, métodos multigrid para solução de equações elípticas lineares e não lineares). Ao final, ainda mostramos que as técnicas empregadas se adaptam à incorporação de adaptatividade nos refinamentos e estudamos casos com refinamentos móveis, acompanhando fenômenos como centros de baixa pressão.

## Abstract

The central objective of this work was to prove the feasibility of the use of local refinement techniques in order to obtain high resolution in a region of interest, using a global numerical weather prediction model. In this way, we aim to achieve local accuracies comparable to that of fine resolution global models, but at considerably lower computational cost. The advantage of working with a global model, instead of using local models, is that we avoid the introduction of artificial boundary conditions. Furthermore, we may have a single model for local and global predictions. Our study was based on a sequence of models, beginning with a simple advection equation model on the sphere, followed by a barotropic vorticity model until we got to a global model for the shallow-water equations. Our development was based on the use of two-time-level semi-Lagrangian schemes, unifying the treatment of the models, while they gradually incorporated new aspects according to their growing complexity (interpolations, computation of trajectories, multigrid methods for solutions of linear and non linear elliptic equations). Finally, we have shown that the methods developed are suited to incorporate some degree of adaptiveness in the local refinement. We investigated cases with moving refinements, following low-pressure centers.

---

# Índice

<b>Introdução</b>	<b>1</b>
<b>1 Advecção Passiva sobre a esfera.</b>	<b>5</b>
1.1 Advecção Semi-Lagrangeana . . . . .	5
1.2 Problema teste. . . . .	7
1.2.1 Descrição do teste. . . . .	7
1.2.2 Sistemas de coordenadas. . . . .	8
1.2.3 Cálculo do dado inicial . . . . .	9
1.3 Trajetória . . . . .	9
1.4 Malhas com refinamento local . . . . .	10
1.5 Interpolação . . . . .	12
1.6 Teste numérico e resultados . . . . .	13
1.6.1 Comportamento do modelo em malhas uniformes . . . . .	13
1.6.2 Comportamento do modelo em malhas localmente refinadas . . . . .	17
<b>2 Um modelo de Vorticidade Barotrópica Não Divergente</b>	<b>21</b>
2.1 Trajetória na Advecção Semi-Lagrangeana (exemplo uni-dimensional) . . . . .	22
2.2 Método Semi-Lagrangeano para a equação da vorticidade barotrópica . . . . .	24
2.3 Discretização da equação de Poisson na esfera . . . . .	25
2.3.1 Discretização nos polos . . . . .	26
2.4 Método Multigrid . . . . .	27
2.4.1 Full-Multigrid . . . . .	31
2.4.2 FAS Multigrid em malhas com refinamento local . . . . .	32

2.5	Cálculo da trajetória Lagrangeana na esfera . . . . .	32
2.6	Resultados . . . . .	36
2.6.1	Multigrid . . . . .	36
2.6.2	Vorticidade barotrópica semi-Lagrangeana . . . . .	39
<b>3</b>	<b>Equações de Água-Rasa</b>	<b>47</b>
3.1	Aspectos gerais do modelo . . . . .	48
3.2	Discretização temporal . . . . .	49
3.3	Malha alternada . . . . .	54
3.4	Discretização espacial da equação não linear para o geopotencial . . . . .	55
3.4.1	Discretização nos polos . . . . .	56
3.5	Cálculo das trajetórias . . . . .	58
3.6	Resolução da equação não linear via Multigrid . . . . .	60
3.7	Resultados . . . . .	60
3.7.1	Multigrid não linear . . . . .	61
3.7.2	Modelo de água-rasa . . . . .	62
<b>4</b>	<b>Refinamento adaptativo</b>	<b>81</b>
4.1	Do refinamento fixo ao adaptativo . . . . .	82
4.2	Algoritmos . . . . .	83
4.2.1	Vorticidade barotrópica . . . . .	83
4.2.2	Água-rasa . . . . .	84
4.3	Resultados . . . . .	85
4.3.1	Vorticidade barotrópica . . . . .	85
4.3.2	Água-rasa . . . . .	89
<b>5</b>	<b>Conclusões</b>	<b>97</b>
	<b>Apêndices</b>	<b>99</b>
<b>A</b>	<b>Advecção unidimensional</b>	<b>99</b>
A.1	Interpoladores que preservam a forma . . . . .	101
A.2	Teste numérico e resultados . . . . .	104
<b>B</b>	<b>Transformação do vetor horizontal para o plano tangencial</b>	<b>111</b>

Bibliografía



---

# Introdução

Este trabalho tem por objetivo o estudo do uso de malhas com refinamento local em modelos globais de previsão do tempo, como uma forma eficiente de se obter alta resolução local em uma dada região de interesse. A idéia de usar malhas com refinamento local é obter essa melhor resolução na região de interesse, sem que precisemos utilizar a mesma resolução no restante do globo, e assim, com um custo computacional muito menor.

As malhas com refinamento local se apresentam como opção ao uso de modelos de área limitada (por exemplo Gustafsson 1991, Cullen 1991), que são alimentados por condições de fronteira geradas por modelos globais. Como essas condições de fronteira não são naturais para o problema, em modelos de área limitada (que nessa forma não são nem bem postos), costuma-se adotar técnicas empíricas, como relaxação em zonas próximas à fronteira, para minimizar os efeitos da prescrição dos campos meteorológicos na fronteira.

A vantagem do uso de um modelo global com malhas localmente refinadas, em contraposição a modelos de área limitada, é que se evita a introdução de condições artificiais de fronteira. Além disso, a possibilidade do uso de resoluções variáveis torna em princípio possível o uso de um único modelo para previsões locais, regionais e globais. O uso de resoluções variáveis em modelos de água-rasa já foi proposto para métodos espectrais, de diferenças finitas e elementos finitos (Courtier e Geleyn 1988, Barros et al.1990, Staniforth e Mitchell 1978). Todos estes fazem uso de stretching de coordenadas e as malhas resultantes, razoavelmente uniformes sobre a região de interesse, apresentam faixas onde a resolução em uma coordenada é bem mais alta que na outra, gerando anisotropias nas equações. Algumas destas idéias vêm sendo aproveitadas em modelos globais de previsão (Geley 1998, Staniforth et al. 1998).

Propomos aqui o emprego de técnicas de refinamento local. A partir de uma malha global de menor resolução vamos gradativamente introduzindo malhas de resolução mais fina, todas elas uniformes, em torno da região de interesse. Para o teste de tal técnica, nosso principal objetivo foi desenvolver um modelo global de água-rasa. Para tal adotamos uma estratégia incremental,

escolhendo uma hierarquia de modelos com crescentes graus de complexidade.

Nesta direção, primeiro trabalhamos com um modelo simples de advecção com velocidade constante sobre a esfera em malhas localmente refinadas, onde também testamos o uso de métodos semi-Lagrangeanos. Nosso principal objetivo foi o estudo dos efeitos da interpolação (referente ao método semi-Lagrangeano). Usamos um interpolador cúbico de Lagrange e nossos resultados, apresentados no capítulo 1, indicaram que o tratamento semi-Lagrangeano da advecção na esfera adapta-se muito bem a malhas localmente refinadas e os efeitos da interpolação nos métodos semi-Lagrangeanos mantêm-se como em malhas uniformes, produzindo alguma dissipação na solução. Ainda estudamos técnicas de interpolação preservando a forma da função (ver Williamson e Rasch, 1989) e verificamos que os mesmos tratamentos funcionam bem em malhas localmente refinadas. (Estas técnicas foram estudadas em um modelo unidimensional e os resultados encontram-se no apêndice A).

O passo seguinte foi estudar o comportamento de malhas localmente refinadas numa equação mais complexa, a equação de vorticidade barotrópica com vento não divergente. Nesta equação adiciona-se a necessidade de usar um método iterativo no cálculo das posições de partida das trajetórias Lagrangeanas, para o que usamos as idéias de Ritchie 1987. Surge ainda a necessidade de se resolver a equação de Poisson na esfera em malhas com refinamento local, para o que adaptamos um método multigrid de Barros 1991. Nossos resultados mostram que o uso de refinamento local na equação de vorticidade barotrópica causa uma melhora na precisão da solução. As técnicas utilizadas e os resultados obtidos na equação de vorticidade barotrópica são apresentados no capítulo 2.

Como última etapa do desenvolvimento tomamos as equações de água-rasa na esfera. Desenvolvemos inicialmente um modelo semi-implícito, semi-Lagrangeano de dois níveis no tempo, o qual pode ser visto como uma variação do modelo proposto em Bates et al. 1990.

Um primeiro trabalho propondo o uso de técnicas de refinamento local em modelos globais de previsão foi desenvolvido por Ruge et al. 1995. Eles adotaram, no entanto, como ponto de partida um modelo de Barros et al. 1990, que não empregava o método semi-Lagrangeano. Este modelo necessitava de Filtragem de Fourier nas regiões dos polos e de um filtro de Shapiro para a supressão de altas frequências. Na implementação que fizeram, Ruge et al. mantiveram uma malha uniformemente fina para estas filtragens. Eles demonstraram que seria possível o uso de refinamento local, mas não estudaram a precisão da técnica e não forneceram resultados sobre eficiência computacional.

Técnicas espectrais têm sido escolhidas para a discretização horizontal de modelos globais de previsão do tempo em boa parte dos modelos operacionais. Modelos de diferenças finitas semi-implícitos semi-Lagrangeanos, com métodos otimais para a resoluções das equações elípticas tem uma potencial vantagem computacional, uma vez que têm complexidade ótima, enquanto

que os espectrais são assintoticamente mais custosos devido à transformada de Legendre.

Métodos semi-Lagrangeanos para modelos globais de água-rasa foram desenvolvidos em conjunto com métodos espectrais (Ritchie 1988, Côté e Staniforth 1988), diferenças finitas (McDonald e Bates 1989, Bates et al. 1990) e elementos finitos (Côté e Staniforth 1990). Um ótimo apanhado sobre métodos semi-Lagrangeanos se encontra em Staniforth e Côté 1991.

O modelo de diferenças finitas Bates et al. (1990) produziu resultados bem melhores que os de McDonald e Bates (1989) com significativa melhora na precisão para passos no tempo maiores. Tinham no entanto que usar um termo de amortecimento do divergente em certas situações, pois alguns problemas de estabilidade ainda persistiam.

O modelo que propomos para a equação de água-rasa neste trabalho difere do modelo proposto por Bates et al. (1990), principalmente no tratamento totalmente implícito do termo de divergência de massa na equação da continuidade, o qual escrevemos em função do logaritmo do geopotencial. Desta maneira, eliminamos o termo de amortecimento do divergente utilizado por eles. Com essa formulação obtemos um método estável sem precisar da adição de termos de filtragem seletiva. O preço a pagar por esta formulação totalmente implícita é a necessidade de solução de uma equação não linear para o geopotencial. Para tal, adaptamos o método multigrid usado na equação de vorticidade barotrópica para a equação de Poisson. Aqui temos uma equação elíptica não linear, com outros termos de primeira ordem além do Laplaciano da equação de Poisson. A flexibilidade do método multigrid se faz valer aqui. Conseguimos uma eficiência equivalente à da solução da equação de Poisson ao resolver a equação não linear.

A adaptação do esquema a malhas com refinamento local é feita como no caso da equação de vorticidade barotrópica e obtemos um esquema igualmente eficiente. Fizemos diversos testes com o modelo que se mostrou plenamente estável. Fizemos várias experiências com dados reais e testamos a capacidade do modelo gerar boas soluções locais, mesmo empregando uma resolução global relativamente pobre. Os resultados mostram a viabilidade da técnica. A curto prazo, obtemos soluções com refinamento local com precisão comparável à obtida com uma resolução uniformemente fina. Os custos computacionais são no entanto reduzidos em mais de uma ordem de magnitude.

Por estarmos empregando um método semi-Lagrangeano, vimos que seria bastante simples e natural experimentar o uso de algum refinamento adaptativo. Um refinamento adaptativo, é um refinamento local que não precisa ser mantido fixo, tanto a posição como o tamanho podem variar durante a integração. Temos que considerar algum critério para decidir onde ficará o novo refinamento. Podemos nos guiar pelo erro de truncamento (como em Skamarock 1989), ou seja, refinamos onde o erro de truncamento assume valores maiores em cada passo da integração, ou podemos procurar refinar onde a solução tem maiores gradientes.

O uso de refinamento adaptativo, em tese, não acarreta maiores complicações ao algoritmo,

apenas tem-se que calcular a nova posição do refinamento e guardar informações sobre a localização das duas malhas.

No capítulo 4, testamos a idéia de um refinamento móvel, fazendo a região do refinamento perseguir algum fenômeno interessante.

Consideramos primeiro o modelo de vorticidade barotrópica, no qual verificamos a viabilidade da técnica, conseguindo ganhos em precisão ao variar o refinamento, com mínimo custo extra. Também para as equações de água-rasa mostramos ser esta uma possibilidade interessante. Não nos aprofundamos, no entanto, nos critérios para adaptatividade, sendo este um tema para futuras investigações. Também não investigamos a possibilidade de criação dinâmica de várias regiões refinadas.

Nosso trabalho está estruturado em 4 capítulos, sendo os três primeiros dedicados a cada um dos modelos que nos propusemos a investigar. Começamos pelo modelo de advecção no capítulo 1, seguido pela equação de vorticidade barotrópica no capítulo 2 e pelo modelo de água-rasa no capítulo 3. O capítulo 4 é dedicado aos experimentos com malhas moveis, seguido por uma seção com conclusões. Incluímos ainda dois apêndices, um contemplando interpolações preservando forma e o outro com detalhes técnicos do cálculo das trajetórias.

# Advecção Passiva sobre a esfera.

Como um primeiro modelo, visando o uso de malhas localmente refinadas em modelos globais de previsão do tempo tomamos uma simples equação de advecção passiva sobre a esfera, tratada através de métodos semi-Lagrangeanos.

Este modelo permite um estudo separado dos efeitos da interpolação cúbica nos métodos semi-Lagrangeanos em malhas com refinamento local. Adotamos neste capítulo o mesmo teste utilizado por (Ritchie 1987), onde as trajetórias das partículas correspondentes ao método semi-Lagrangeano podem ser calculadas exatamente. Assim isolamos os efeitos da interpolação sem ter a presença dos erros que poderiam advir do cálculo aproximado da trajetória.

Comparamos os resultados com os obtidos em malhas uniformes. Nossos resultados indicam que o tratamento semi-Lagrangeano da advecção na esfera adapta-se muito bem a malhas localmente refinadas. Em particular, não observamos reflexão de ondas na transição entre malhas com diferentes níveis de refinamento. Os efeitos da interpolação nos métodos semi-Lagrangeanos mantêm-se como em malhas uniformes, produzindo alguma dissipação na solução.

## 1.1 Advecção Semi-Lagrangeana

Nesta seção apresentamos a idéia básica do método semi-Lagrangeano (veja A. Staniforth e J. Côté 1991 e referências lá contidas) aplicado à equação de advecção passiva unidimensional:

$$\frac{d}{dt}F(r(t), t) = 0, \quad (1.1)$$

com

$$\frac{dr}{dt} = V(x, t), \quad (1.2)$$

onde  $V(x, t)$  é uma função dada (a velocidade de advecção do campo  $F$ ) e  $r(t)$  é o vetor posição como função do tempo. A equação (1.1) implica que a função  $F$  permanece constante ao longo das trajetórias do fluido. Nossa descrição segue de perto as idéias de Ritchie 1987.

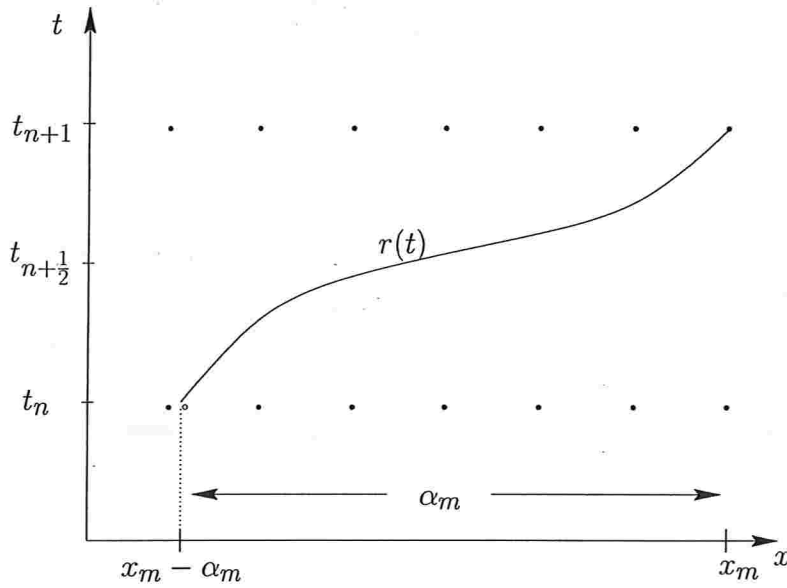


Figura 1.1: Esquema de dois níveis.

Descrevemos a seguir um método semi-Lagrangeano de dois níveis no tempo. Para ilustrar a idéia do método, na figura 1.1 representamos por  $r(t)$  a trajetória exata de uma partícula que depois de um intervalo de tempo  $\Delta t$  chega no ponto de malha  $x_m$ .

Supondo conhecidos os valores de  $F$  no tempo  $t_n$  para todos os pontos  $x_m$  de uma malha fixa, nós queremos conhecer os valores da função  $F$  nos mesmos pontos da malha  $x_m$  no tempo  $t_{n+1}$ . A idéia do método semi-Lagrangeano é integrar (1.1), ao longo da trajetória do fluido. Por conseguinte, sendo  $r$  a trajetória de uma partícula do fluido tal que  $r(t_n)$  é a posição da partícula no tempo  $t_n$ , que chega ao ponto de malha  $x_m$  no tempo  $t_{n+1}$ , discretizamos (1.1) como

$$\frac{F(x_m, t_{n+1}) - F(r(t_n), t_n)}{\Delta t} = 0, \quad (1.3)$$

ou

$$F(x_m, t_{n+1}) = F(r(t_n), t_n). \quad (1.4)$$

Logo, para determinar o valor de  $F(x_m, t_{n+1})$  precisamos determinar o valor de  $r(t_n)$  correspondente a cada ponto de malha  $x_m$ , o que no caso de advecção passiva com a trajetória do fluido conhecida é um problema simples. Em geral, é necessário calcular uma aproximação para a trajetória do fluido, durante o processo de integração semi-Lagrangeana. Abordaremos este problema no próximo capítulo, ao tratar das equações de Vorticidade Barotrópica.

Conhecido o ponto de partida  $r(t)$  para todo  $x_m$ , através de (1.4) podemos calcular o valor de  $F$  em todos os pontos da malha no tempo  $t_{n+1}$ . Geralmente o ponto de partida da partícula

$r(t)$  não é um ponto da malha, sendo necessário o uso de interpolação (polinomial em nosso caso) para obter o valor de  $F$  nos pontos  $r(t_n)$  para todo  $m$ .

## 1.2 Problema teste.

Como caso teste para as malhas com refinamento local em modelos meteorológicos, adotamos o problema sugerido por Ritchie 1987, descrito a seguir.

### 1.2.1 Descrição do teste.

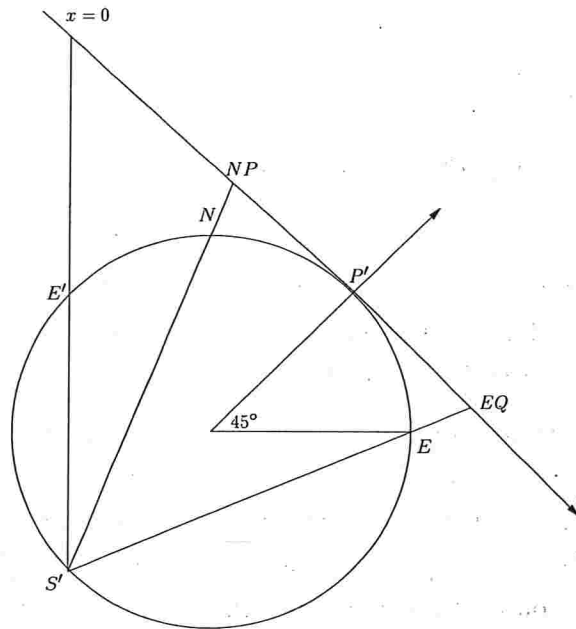


Figura 1.2: Diagrama do teste.

A equação de advecção em duas dimensões é dada por

$$\frac{d}{dt}F(r(t), t) = 0, \quad (1.5)$$

onde

$$\frac{d}{dt}(\cdot) = \frac{\partial}{\partial t}(\cdot) + V \cdot \nabla(\cdot), \quad (1.6)$$

$r(t)$  neste caso é um vetor posição em função do tempo e  $V$  é o vetor de velocidades.

Analogamente ao caso unidimensional, o valor do campo  $F$  no novo passo no tempo será dado por (1.4)

O teste consiste num movimento de rotação de corpo rígido. Sendo  $P'$  um ponto na superfície da esfera, o movimento consiste na rotação ao redor de um eixo que passa pelo centro da esfera e por  $P'$ , com velocidade angular constante  $\omega$  (ver figura 1.2).

Temos o sistema de coordenadas esféricas  $\{(\lambda, \theta) : (\lambda, \theta) \in [-\pi, \pi] \times [-\pi/2, \pi/2]\}$  no qual o ponto  $P'$  tem coordenadas  $(\lambda_0, \theta_0)$ . Designaremos  $(\lambda', \theta')$  um outro sistema de coordenadas esféricas que tem como polo norte o ponto  $P'$ . Neste novo sistema de coordenadas o movimento é simplesmente de rotação em uma coordenada.

### 1.2.2 Sistemas de coordenadas.

Neste novo sistema teremos algumas simplificações, pois

$$\frac{d\lambda'}{dt} = \omega \quad (1.7)$$

$$\frac{d\theta'}{dt} = 0. \quad (1.8)$$

Além disso, estes dois sistemas satisfazem as seguintes relações

$$\sin \theta' = \sin \theta \sin \theta_0 + \cos \theta \cos \theta_0 \cos(\lambda - \lambda_0), \quad (1.9)$$

$$\sin \theta = \sin \theta' \sin \theta_0 + \cos \theta' \cos \theta_0 \cos \lambda', \quad (1.10)$$

$$\cos \theta' \sin \lambda' = \cos \theta \sin(\lambda - \lambda_0). \quad (1.11)$$

Da equação (1.9) obtemos

$$\theta'(\lambda, \theta) = \sin^{-1}(\sin \theta \sin \theta_0 + \cos \theta \cos \theta_0 \cos(\lambda - \lambda_0)), \quad (1.12)$$

e de (1.9) e (1.10) que

$$\cos \theta \cos(\lambda - \lambda_0) = \sin \theta' \cos \theta_0 + \cos \theta' \sin \theta_0 \cos \lambda'.$$

Logo, usando (1.11) temos que

$$\lambda'(\lambda, \theta) = \tan^{-1}(\sin(\lambda - \lambda_0) / [\sin \theta_0 \cos(\lambda - \lambda_0) - \cos \theta_0 \tan \theta]). \quad (1.13)$$

Analogamente obtemos a relação inversa

$$\theta(\lambda', \theta') = \sin^{-1}(\sin \theta' \sin \theta_0 + \cos \theta' \cos \theta_0 \cos \lambda'), \quad (1.14)$$

e

$$\lambda(\lambda', \theta') = \tan^{-1}(\sin(\lambda') / [\tan \theta' \cos \theta_0 - \sin \theta_0 \cos \lambda']) + \lambda_0. \quad (1.15)$$



### 1.2.3 Cálculo do dado inicial

Para comparação dos resultados vamos considerar o plano tangente à esfera no ponto  $P'$ , considerando a projeção estereográfica da esfera no ponto  $P'$  sobre esse plano, e transformando as coordenadas esféricas  $(\lambda', \theta')$  em coordenadas  $(X, Y)$  sobre o plano tangente por meio de

$$X = 2a \frac{\cos(\lambda') \cos(\theta')}{1 + \sin(\theta')},$$

$$Y = 2a \frac{\sin(\lambda') \cos(\theta')}{1 + \sin(\theta')}$$

onde  $P'$  será a origem no plano  $(X, Y)$ . O movimento de rotação de corpo rígido com velocidade angular constante  $\omega$  ao redor do eixo que passa pelo centro da esfera e o ponto  $P'$  corresponde no plano a um movimento de rotação do campo ao redor da projeção estereográfica do ponto  $P'$ .

Escolhemos como campo inicial uma função 'Gaussiana' com altura máxima de 100 unidades no centro, e definida por

$$F(r) = 100e^{-\frac{\pi r^2}{L^2}} \quad (1.16)$$

onde  $r$  é a distância do ponto ao centro da função no plano  $(X, Y)$ , e  $L$  define a 'largura' da Gaussiana.

Tomamos  $P' = (0^\circ, 45^\circ)$  no sistema de coordenadas  $(\lambda, \theta)$  e escolhemos como a posição inicial do centro da Gaussiana o ponto  $EQ$  que é a projeção do ponto  $E$  do Equador no plano tangente (ver figura 1.2). A solução analítica vai estar inicialmente centrada no ponto  $EQ$ , depois de meia volta vai passar sobre o polo norte (o ponto  $NP$  no plano) para voltar ao ponto  $EQ$  ao concluir uma volta. Faremos a comparação dos resultados numéricos com a solução analítica.

## 1.3 Trajetória

Um dos passos importantes em algoritmos semi-Lagrangeanos é o cálculo da trajetória. Aqui, no entanto, aproveitaremos o fato de que a posição de partida da partícula pode ser calculada exatamente, visto que o movimento é uma rotação constante no tempo. Desta forma, poderemos isolar os efeitos da interpolação (cúbica).

Para cada ponto da malha  $(\lambda_m, \theta_m)$  no sistema de coordenadas  $(\lambda, \theta)$  usando (1.12) e (1.13) obtemos  $(\lambda'_m, \theta'_m)$ , sua representação no sistema  $(\lambda', \theta')$ .

O movimento do campo neste sistema consiste de uma rotação com velocidade constante  $\omega$  ao redor do eixo que passa pelo centro da esfera e o ponto  $P'$  (ver figura 1.2)). Neste sistema

de coordenadas  $(\lambda', \theta')$  temos um movimento com velocidade constante  $\omega$  na longitude. Por conseguinte, as coordenadas do ponto de partida (num intervalo de tempo  $\Delta t$ ) são dadas por

$$\lambda'_{m_p} = \lambda'_m - \omega \Delta t$$

e

$$\theta'_{m_p} = \theta'_m.$$

O ponto de partida da partícula é então transformado para o sistema de coordenadas  $(\lambda, \theta)$  usando (1.14) e (1.15).

## 1.4 Malhas com refinamento local

Nosso principal interesse no uso de malhas localmente refinadas é poder obter uma melhor resolução (bastante fina) numa região específica (a qual chamamos de região de interesse) sem utilizar a mesma resolução no restante do globo. Desta forma, o custo computacional é bastante reduzido em relação a um modelo global com malha uniforme. Com o uso de refinamento local em um modelo global evitamos a necessidade de tratamentos especiais junto à fronteira, como em modelos regionais (por exemplo Gustafsson 1991, Cullen 1991).

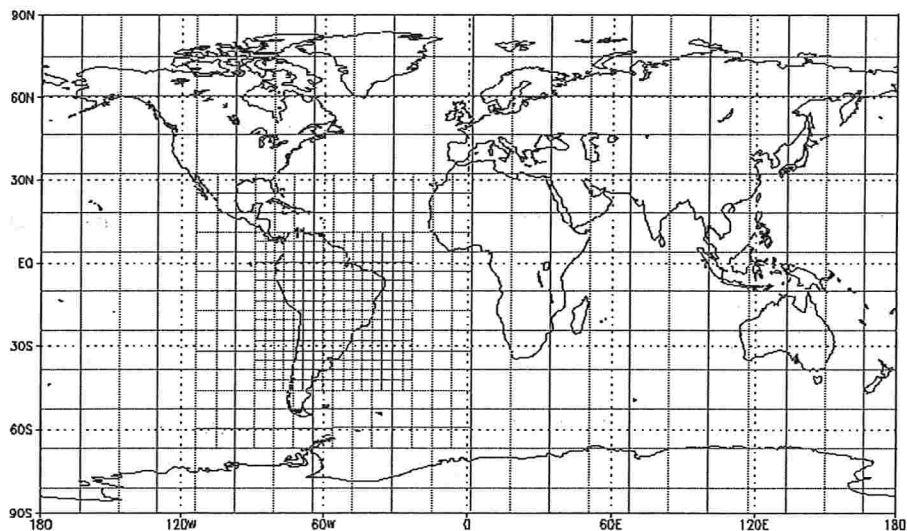


Figura 1.3: Malhas localmente refinadas.

Adotaremos uma malha uniforme (relativamente grossa) sobre toda a esfera, a qual chamaremos malha base. Adotamos então refinamentos gradativos, com passos de malha duas vezes mais finos a cada passo. Estes refinamentos são uniformes, englobando a região de interesse de

forma a permitir uma transição suave entre a malha base e os diferentes níveis de refinamento (ver figura 1.3). A partir de uma malha global de menor resolução vamos gradativamente introduzindo malhas de resolução mais fina, em torno da região de interesse.

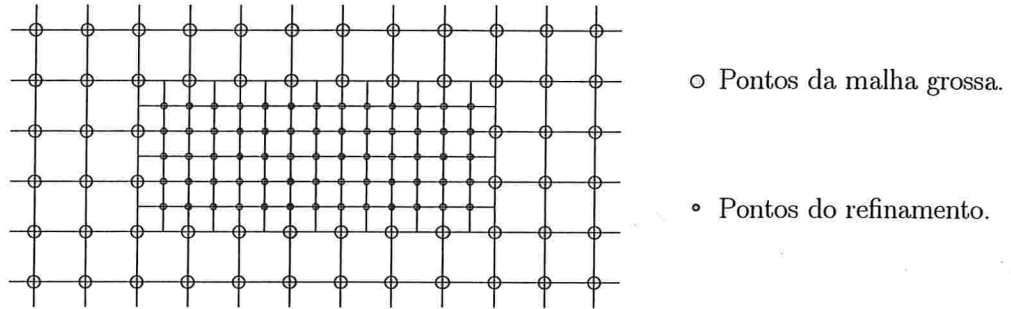


Figura 1.4: Pontos pertencentes a malha base e ao refinamento.

Escolhidos a posição do refinamento e a quantidade de pontos que ele terá nas duas direções, na figura 1.4 podemos observar como são tomados os pontos do refinamento com respeito à malha base. O uso de um fator dois de refinamento e a posição relativa dos pontos da malha mais fina e mais grossa foram escolhidos assim por razões de eficiência, ligadas ao uso de Multigrid (a ser descrito no próximo capítulo).

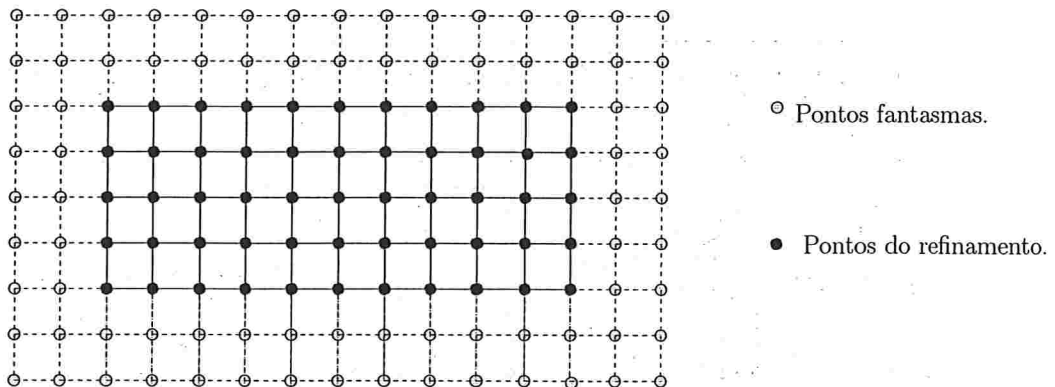


Figura 1.5: Pontos fantasmas nas malhas refinadas.

Em nossa implementação (em Fortran 77) os campos referentes a cada malha serão representados em matrizes bidimensionais uniformes, nas rotinas mais internas ao código. Na estrutura global, os valores de  $F$  nas diferentes malhas serão guardados num mesmo vetor, iniciando pela malha base e continuando pelos refinamentos, do mais grosso até o de melhor resolução. O acesso aos valores de  $F$  nas diferentes malhas será feito através de ponteiros, que indicarão a posição inicial de cada uma das diferentes malhas.

Para o campo  $F$  em cada malha corresponderá uma matriz, na qual reservaremos posições extras ('os pontos fantasmas') para armazenar valores auxiliares para os cálculos. Nós reservaremos duas linhas no início da matriz e duas no final assim como duas colunas no início e duas no final, para o armazenamento destes pontos fantasmas em cada malha (ver figura 1.5).

Mais detalhe sobre o uso e a estrutura das malhas localmente refinadas serão fornecidos nas seções seguintes.

## 1.5 Interpolação

Uma vez calculado o ponto de partida da partícula, precisaremos normalmente interpolar os valores de  $F$  usando os pontos da malha onde  $F$  está definida. Empregaremos um polinômio interpolador de Lagrange bicúbico, com erro de interpolação da ordem  $O(\Delta x^4)$ , que será suficiente para não afetar a ordem de discretização dos esquemas empregados nos próximos capítulos.

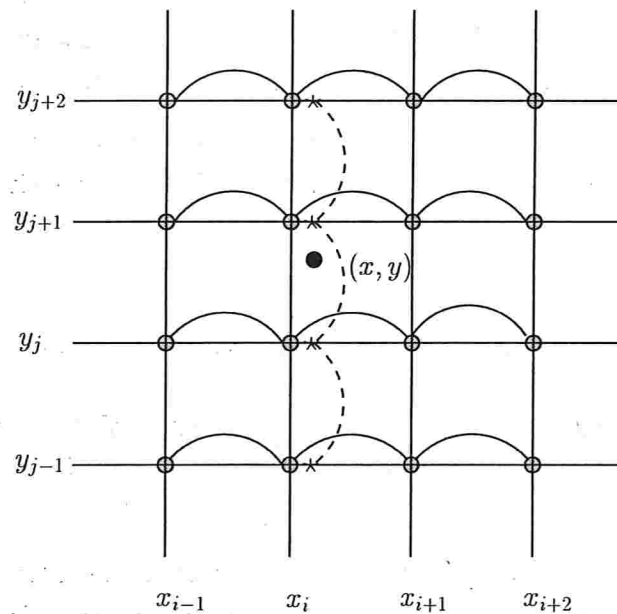


Figura 1.6: Diagrama da interpolação cúbica.

Na prática, a interpolação bicúbica é implementada por sucessivas interpolações cúbicas unidimensionais, alternando as direções. Por exemplo, suponhamos que o ponto no qual queremos procurar o valor de  $F$  é  $(x, y)$  tal que  $x_i \leq x \leq x_{i+1}$ ,  $y_j \leq y \leq y_{j+1}$  onde  $(x_k, y_n)$  com  $k = i, i+1$  e  $n = j, j+1$  são pontos da malha (ver figura 1.6). Então empregamos interpolação cúbica dos valores de  $F$  em cada linha  $x_{i+k}$  com  $k = -1, \dots, 2$  para obter valores de  $F$  nos pontos  $(x, y_{j+k})$

com  $k = -1, \dots, 2$ , e a partir destes interpolamos cubicamente na coluna  $x$  para obter  $F$  em  $(x, y)$ .

Nas malhas localmente refinadas temos pontos na região de transição dos diferentes níveis de refinamento. Antes de entrar na interpolação atualizaremos os pontos fantasmas descritos na seção anterior. Nós transferimos os valores de  $F$  para os pontos fantasmas através de interpolação cúbica dos valores da malha de resolução grossa imediatamente anterior. Uma vez atualizados estes pontos fantasmas a interpolação cúbica no refinamento se realiza como descrito anteriormente, sem que seja necessário verificar se os pontos estão junto à fronteira, pois os pontos fantasmas, contendo os valores necessários para interpolação, permitem um tratamento uniforme para todos os pontos.

## 1.6 Teste numérico e resultados

O método semi-Lagrangeano, não coloca restrições de estabilidade na escolha do passo no tempo  $\Delta t$ , podendo esta escolha ser norteada apenas por razões de precisão. Já que neste modelo particular a trajetória da partícula é calculada exatamente, a escolha do passo no tempo  $\Delta t$  será importante só para determinar o número de interpolações a serem feitas.

Nos testes escolhemos  $\omega$  tal que a 'Gaussiana' complete uma volta depois de 20 dias. Empregamos  $\Delta t = 2h$ , assim a cada 20 dias estaremos observando efeitos cumulativos de 240 interpolações.

### 1.6.1 Comportamento do modelo em malhas uniformes

Nestes testes usamos  $\Delta t = 2h$  e três funções iniciais diferentes, obtidas variando a largura  $L$  da 'Gaussiana' em (1.16). Em todos eles empregamos uma malha uniforme na esfera de  $124 \times 65$  pontos.

Na figura 1.7 podemos observar a função aproximada para diferentes tempos e valores de  $L$ . Pela escolha de  $\omega$ , a 'Gaussiana' que inicialmente se encontra centrada no equador, deve dar uma volta completa a cada 20 dias.

Na figura 1.7 temos na coluna (a) os resultados após uma volta, aos 20 dias, e na coluna (b), após 40 dias. Na primeira linha os resultados foram obtidos com  $L = 10000km$ , na segunda linha com  $L = 5000km$  e na terceira com  $L = 2500km$ .

Relembrando que a altura máxima da função inicial é de 100 unidades, observa-se nos resultados, um rebaixamento do máximo da 'Gaussiana'. Quanto menor o suporte da 'Gaussiana' o rebaixamento é mais intenso. Este comportamento é consequência das sucessivas interpolações.

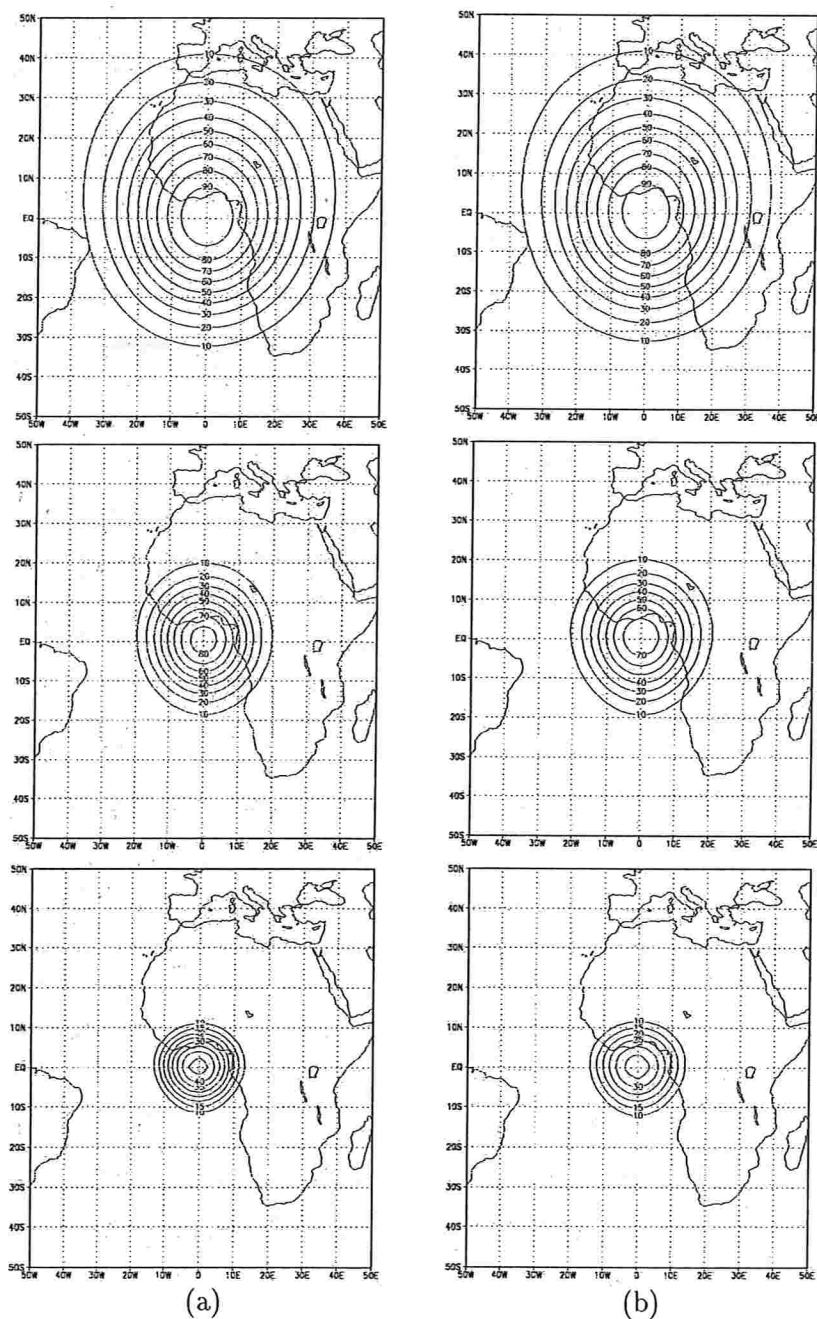


Figura 1.7: Função aproximada para diferentes valores de  $L$ . Na coluna (a) temos o resultado depois de 20 e na coluna (b) depois de 40 dias de integração.

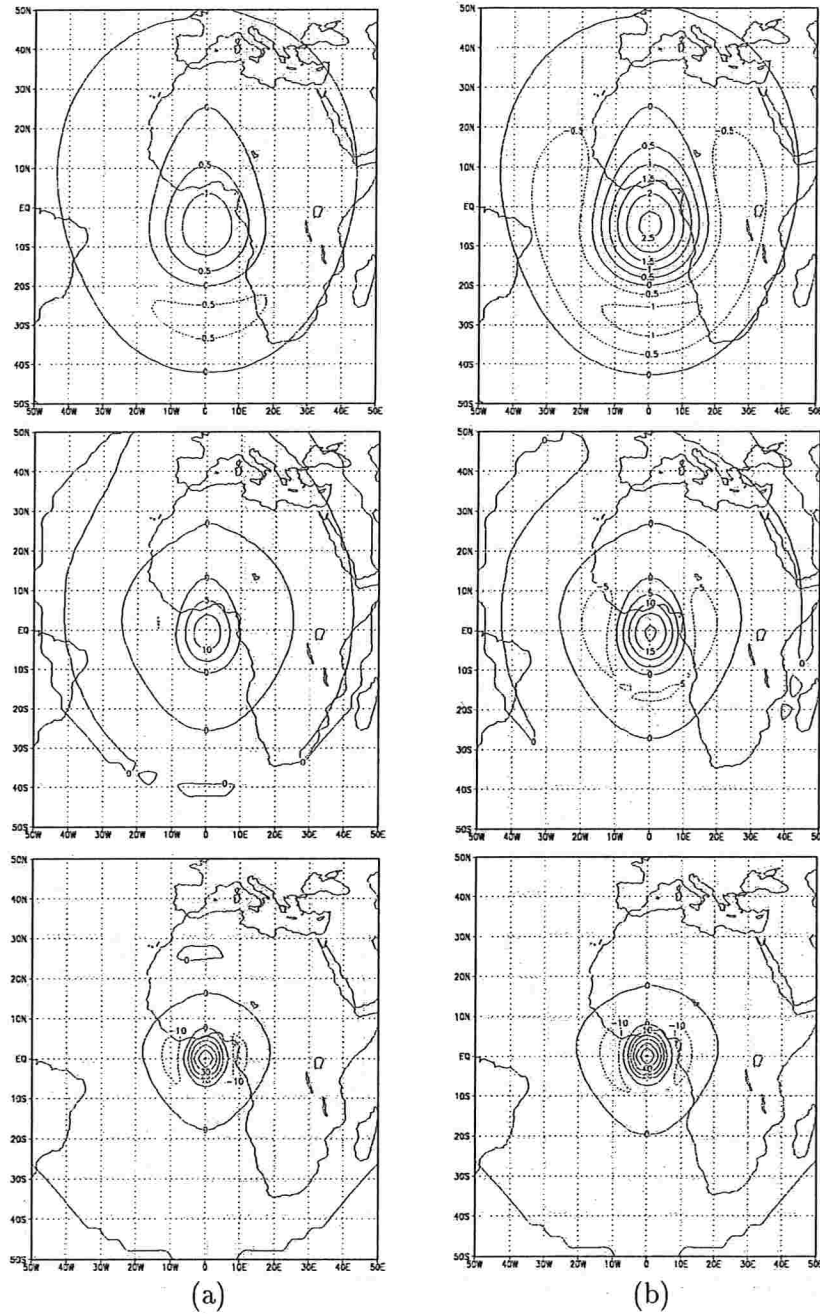


Figura 1.8: Diferença entre a solução analítica e a função aproximada para diferentes valores de  $L$ . Na coluna (a) temos o resultado depois de 20 e na coluna (b) depois de 40 dias de integração.

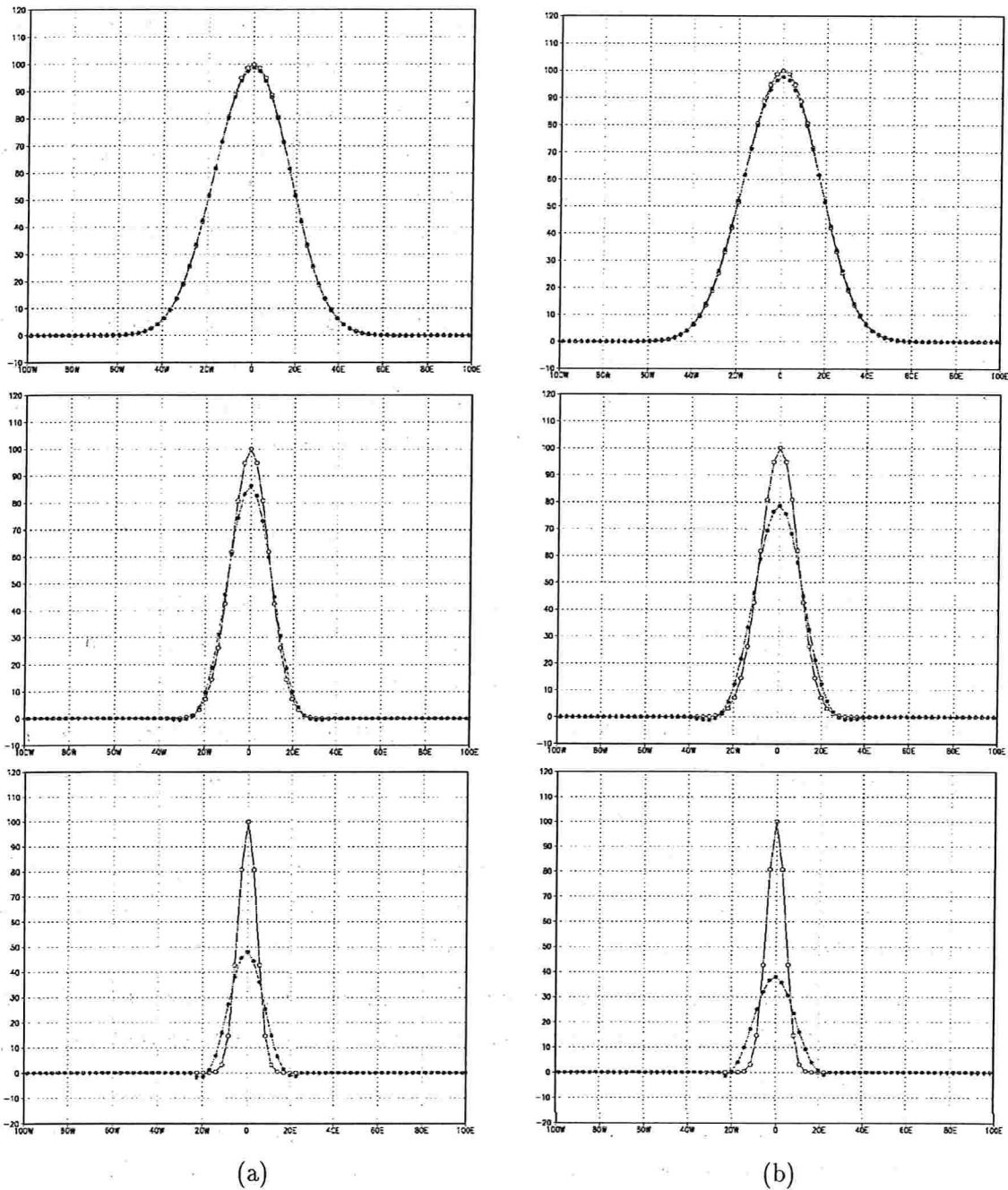


Figura 1.9: Corte transversal da solução analítica e a função aproximada para diferentes valores de  $L$ . Na coluna (a) temos o resultado depois de 20 e na coluna (b) depois de 40 dias de integração.



Se interpolação linear fosse empregada os resultados seriam inaceitáveis.

Na figura 1.8 apresentamos a diferença entre a solução analítica e a função aproximada. Novamente mostramos na coluna (a) os resultados depois de 20 dias de integração e em (b) depois de 40 dias, com três valores diferentes de  $L$  como na figura 1.7.

Nestes gráficos podemos observar que a diferença toma valores positivos no centro da 'Gaussiana', refletindo o efeito de rebaixamento da altura máxima da 'Gaussiana'. Também podemos observar valores negativos, nos pontos que ficam mais longe do centro. Isto é mais notório após 40 dias de integração e também é devido à interpolação cúbica, que não preserva a monotonicidade das funções.

Finalmente na figura 1.9 mostramos um corte transversal, comparando a solução analítica e a função aproximada na altura do equador. Nestes gráficos se pode ver com clareza os dois efeitos observados nas figuras anteriores. Pode se notar valores ligeiramente negativos da aproximação, mais notórios nos gráficos após 40 dias e quando o suporte é menor.

### 1.6.2 Comportamento do modelo em malhas localmente refinadas

O objetivo destes testes é verificar primeiramente, a aparição de algum efeito indesejado, que seja causado pela presença dos refinamentos, como reflexões de onda. Além disso, devem evidenciar as vantagens de uma melhor resolução numa determinada região do globo.

Nestes casos utilizaremos a função inicial dada por (1.16) com  $L = 5000km$ . A malha base, uniforme contém  $124 \times 65$  pontos, e acrescentamos regiões gradativamente refinadas, por um fator 2 a cada novo nível de refinamento.

A região refinada é mantida fixa durante a integração e no caso da figura 1.10 e 1.11, a área da região refinada é relativamente pequena. Após poucas iterações, a 'Gaussiana' sai do refinamento para retornar completamente após quase 20 dias de integração.

Na figura 1.10 temos os resultados obtidos com  $L = 5000km$ , a região refinada está marcada por uma linha pontilhada. Como nas figuras anteriores, a coluna (a) mostra os resultados depois de 20 e (b) depois de 40 dias de integração.

Na primeira linha vemos a posição do refinamento e a função aproximada para os diferentes tempos. Comparando com a figura 1.7, observamos que aos 40 dias, a nova função aproximada conserva melhor a altura máxima.

Na segunda linha temos os gráficos da diferença entre a solução analítica e a função aproximada. Neste caso podemos observar melhor o fato da função aproximada não ter sido tão suavizada em comparação com a figura 1.8, e os efeitos de uma melhor resolução.

Notemos que o efeito de se ter uma melhor resolução durante algum tempo em que a 'Gaus-

siana' permanece na região refinada se faz sentir, mesmo após a 'Gaussiana' dar duas voltas completas, deixando a região refinada e mais tarde retornando por duas vezes. Não aparece reflexão de ondas nas interfaces da região refinada. Os valores negativos na base da 'Gaussiana' ainda aparecem, mas mais atenuados. Isto é melhor observado na ultima linha da 1.10, onde mostramos um corte transversal da solução analítica e da função aproximada na altura do equador.

Na figura 1.11 apresentamos resultados obtidos com dois níveis de refinamento (e  $L = 5000km$ ). As colunas (a) e (b) têm o mesmo significado que nas figuras anteriores.

Na primeira linha mostramos a função aproximada nos dois tempos além da posição dos dois níveis de refinamento. Comparando com a figura anterior vemos que a função aproximada apresenta mais uma melhoria na altura máxima.

Na segunda linha mostramos novamente a diferença entre a solução analítica e a função aproximada. Observamos claramente que a adição de mais um nível de refinamento contribuiu para melhorar ainda mais a resolução. O surgimento de valores negativos na aproximação é mais atenuado, como mostram os cortes na ultima linha da 1.11.

A melhora na função aproximada é devida à resolução mais fina nas áreas localmente refinadas. Novamente não se observa a aparição de reflexão de ondas na transição entre malhas com diferentes níveis de refinamento, embora os efeitos da interpolação permaneçam como no caso da malha uniforme.

Nossos resultados indicam que o tratamento semi-Lagrangeano da advecção na esfera adapta-se muito bem a malhas localmente refinadas. Em particular, não há problema na transição entre malhas com diferentes níveis de refinamento. Os efeitos da interpolação nos métodos semi-Lagrangeanos mantêm-se como em malhas uniformes, produzindo alguma dissipação na solução. No caso de advecção de quantidades estritamente positivas (como umidade por exemplo), o aparecimento de 'undershooting' (produzindo quantidades negativas) é indesejável e pode ser tratado com técnicas de interpolação preservando a forma da função. No apêndice A, verificamos que os mesmos tratamentos funcionam bem em malhas localmente refinadas.

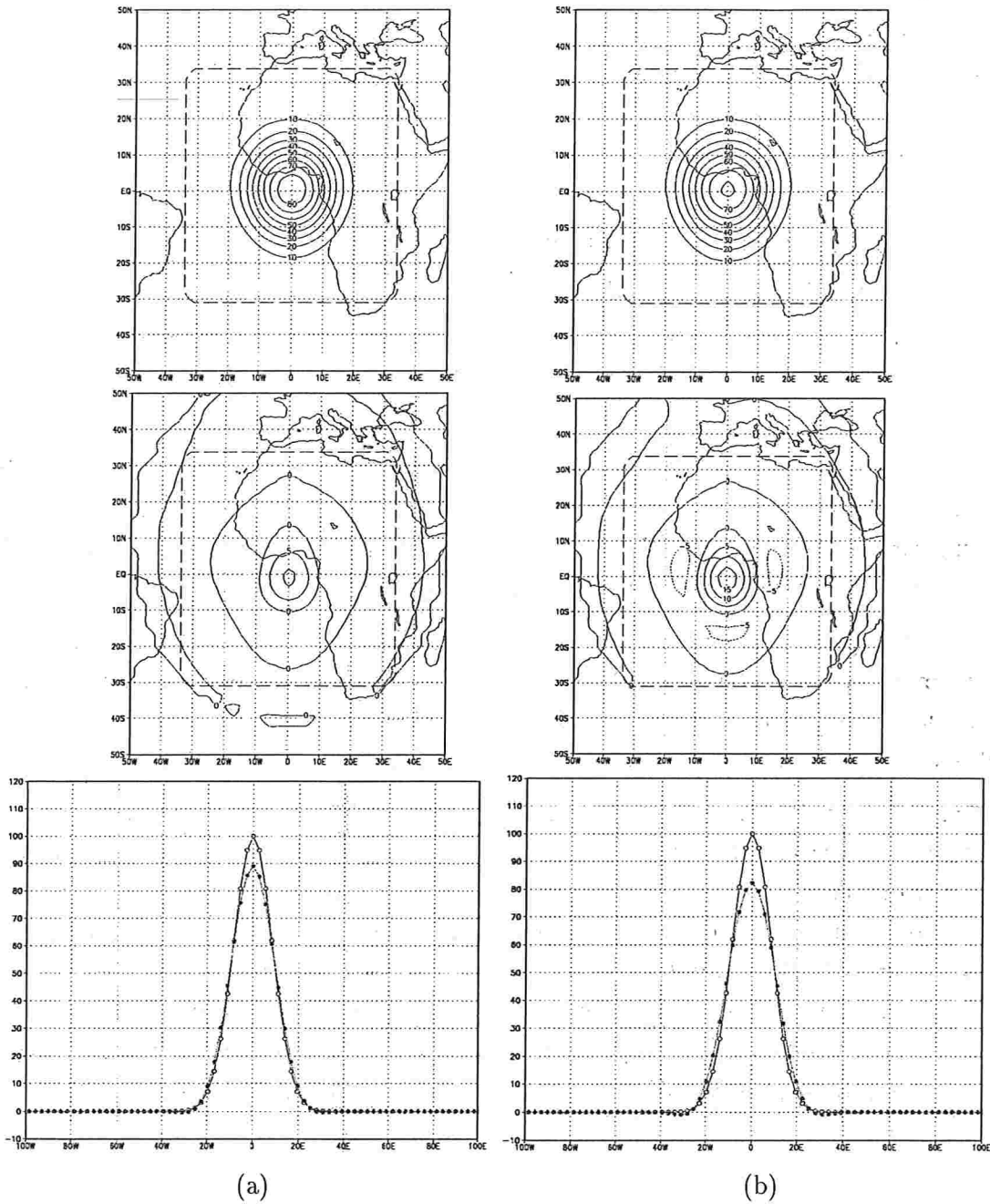


Figura 1.10: Resultados em uma malha com um nível de refinamento local (ver texto), a região refinada está marcada pela linha pontilhada no gráfico. Foi usado o valor  $L = 5000km$ . Na coluna (a) temos o resultado após 20 dias e na coluna (b) após 40 dias de integração.

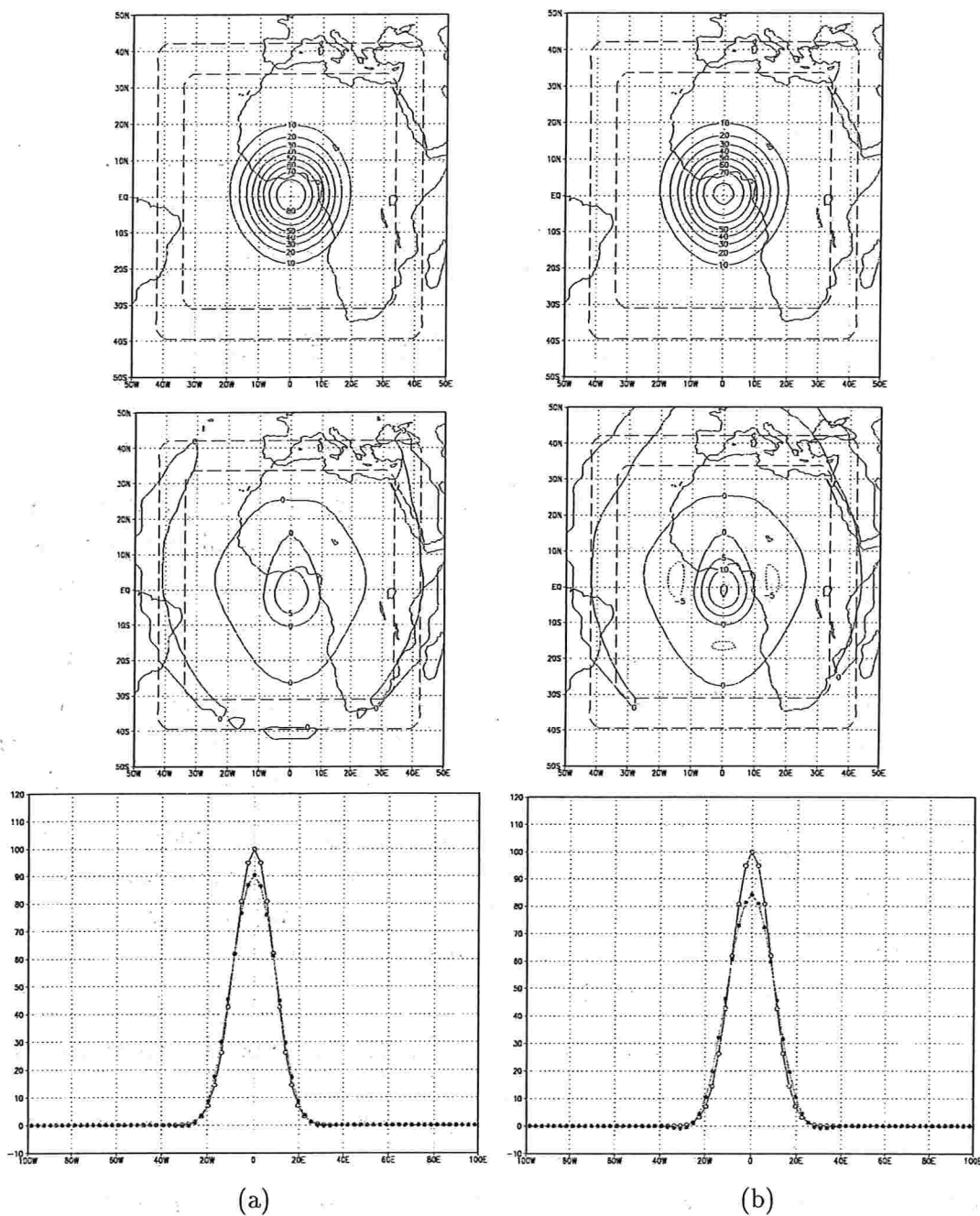


Figura 1.11: Resultados em uma malha com dois níveis de refinamento local, delimitados pelas linhas pontilhadas no gráfico. Foi empregado o valor  $L = 5000km$ . Na coluna (a) temos o resultado após 20 dias e na coluna (b) após 40 dias de integração.

# Um modelo de Vorticidade Barotrópica Não Divergente

No primeiro capítulo, trabalhamos com um modelo simples de advecção constante sobre a esfera em malhas localmente refinadas, onde testamos o uso de métodos semi-Lagrangeanos. Nosso principal objetivo foi o estudo dos efeitos da interpolação. Usamos um interpolador cúbico de Lagrange e nossos resultados indicaram que o tratamento semi-Lagrangeano da advecção na esfera adapta-se bem a malhas localmente refinadas e os efeitos da interpolação nos métodos semi-Lagrangeanos mantêm-se como em malhas uniformes, produzindo alguma dissipação na solução. Também estudamos técnicas de interpolação preservando a forma da função (ver apêndice A) e verificamos que os mesmos tratamentos funcionam bem em malhas localmente refinadas.

Neste capítulo, através do estudo da equação de vorticidade barotrópica iremos introduzir novos aspectos no desenvolvimento dos modelos. O tratamento semi-Lagrangeano das equações de vorticidade barotrópica irá requerer o cálculo das trajetórias do modelo, bem como a resolução de uma equação elíptica a cada passo no tempo. Iremos então usar este modelo para estudar estas questões em malhas localmente refinadas.

A equação da vorticidade barotrópica com vento não divergente pode ser escrita como:

$$\frac{d(\xi + f)}{dt} = 0 \quad (2.1)$$

onde  $f$  é o termo de Coriolis, e  $\xi$  a vorticidade relativa (eg. Haltiner e Williams 1980).

A equação garante a preservação da vorticidade total  $\eta = \xi + f$  ao longo de trajetórias Lagrangeanas. Estas são determinadas com o auxílio de uma função corrente  $\psi$ , através da relação:

$$\xi = \Delta\psi, \quad (2.2)$$

da qual deriva-se o vento não divergente, segundo o qual a vorticidade total é transportada.

Estas equações podem ser derivadas das equações de água-rasa (que consideraremos no próximo capítulo) impondo divergente nulo.

A seguir explicaremos as técnicas utilizadas no tratamento do modelo de vorticidade barotrópica.

## 2.1 Trajetória na Advecção Semi-Lagrangeana (exemplo unidimensional)

No capítulo anterior, introduzimos a idéia geral do método semi-Lagrangeano num exemplo simples de advecção na esfera, no qual as trajetórias das partículas correspondentes ao método semi-Lagrangeano eram conhecidas e portanto não precisamos calculá-las. No modelo para a equação de vorticidade barotrópica as trajetórias das partículas correspondentes ao método semi-Lagrangeano não são conhecidas a priori e portanto precisamos de algum método para calculá-las.

Nesta seção descrevemos o cálculo das trajetórias para uma equação unidimensional. O caso da esfera será tratado na seção 2.5.

Consideremos a equação de advecção unidimensional:

$$\frac{d}{dt}F(r(t), t) = 0, \quad (2.3)$$

com

$$\frac{dr}{dt} = V(x, t), \quad (2.4)$$

onde  $V(x, t)$  é uma função dada (a velocidade de advecção do campo  $F$ ) e  $r(t)$  é o vetor posição como função do tempo. A equação (2.3) implica que a função  $F$  permanece constante ao longo das trajetórias.

Queremos resolver a equação fazendo uso de um método semi-Lagrangeano de dois níveis no tempo. Na figura 2.1 representamos por uma curva contínua  $r(t)$  a trajetória exata de uma partícula que depois de um intervalo de tempo  $\Delta t$  chega no ponto de malha  $x_m$ , a reta tracejada que une os pontos  $A'C$  representa a trajetória aproximada.  $\alpha_m$  é a distância percorrida pela partícula nesse intervalo de tempo.

Supondo conhecidos os valores de  $F$  no tempo  $t$  para todos os pontos  $x_m$  de uma malha fixa, queremos determiná-los no instante  $t + \Delta t$ . Portanto, sendo  $r$  a trajetória de uma partícula do fluido tal que  $r(t)$  é a posição da partícula no tempo  $t$ , que chega ao ponto da malha  $x_m$  no tempo  $t + \Delta t$ , da equação (2.3) temos

$$\frac{F(x_m, t + \Delta t) - F(r(t), t)}{\Delta t} = 0, \quad (2.5)$$

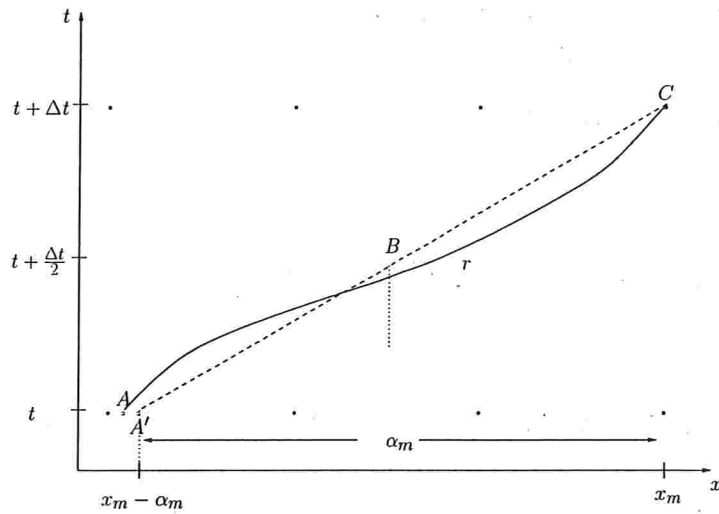


Figura 2.1: Esquema de dois níveis.

ou seja

$$F(x_m, t + \Delta t) = F(r(t), t). \quad (2.6)$$

Portanto, para determinar o valor de  $F(x_m, t + \Delta t)$  precisamos primeiro, encontrar o ponto de partida correspondente  $r(t)$  para cada ponto de malha  $x_m$ . Para tal aproximamos (2.4) usando diferenças finitas centradas. Como  $r(t) = x_m - \alpha_m$  e  $r(t + \Delta t) = x_m$  (ver figura 1.1), temos

$$r(t + \Delta t) - r(t) = \alpha_m = \Delta t \cdot V(x_m - \frac{\alpha_m}{2}, t + \frac{\Delta t}{2}).$$

Esta equação pode ser resolvida iterativamente para o deslocamento  $\alpha_m$

$$\alpha_m^{k+1} = \Delta t \cdot V(x_m - \frac{\alpha_m^k}{2}, t + \frac{\Delta t}{2}),$$

para algum dado inicial  $\alpha^0$ . Para tal precisamos de  $V$  no instante  $t + \frac{\Delta t}{2}$  e na posição  $x_m - \frac{\alpha_m^k}{2}$ . Sendo  $V$  conhecido apenas na malha e nos instantes anteriores este será extrapolado temporalmente e interpolado espacialmente a seguir. Empregamos interpolação linear em ambos os casos, o que é suficiente para uma aproximação de segunda ordem (Temperton e Staniforth 1987):

Conhecidos os pontos de partida  $r(t)$  para toda a malha, os valores de  $F$  podem ser determinados segundo (2.6), após interpolação (cúbica) nos pontos de partida.

## 2.2 Método Semi-Lagrangeano para a equação da vorticidade barotrópica

A discretização semi-Lagrangeana de dois níveis no tempo para a equação de vorticidade barotrópica que usaremos é a seguinte:

$$(\xi + f)_{\bullet}^{n+1} = (\xi + f)_{*}^n, \quad (2.7)$$

onde o símbolo  $*$  denota a posição de partida de uma partícula no instante  $n$  que chega a um ponto de malha  $\bullet$  no instante  $n + 1$ . Conhecidos os pontos de partida, pode-se calcular o valor da vorticidade total nestes pontos e assim, obter seu valor no novo tempo. Como os pontos de partida das partículas em geral não pertencem à malha, necessitaremos interpolar os valores da vorticidade disponíveis na malha. Empregamos interpolação cúbica de Lagrange, como na equação de advecção tratada no capítulo 1.

Para o cálculo da posição de partida da partícula precisamos conhecer o valor do vento no tempo  $n$ . Para isso temos primeiramente que resolver a equação para a função corrente:

$$\xi^n = \Delta\psi^n. \quad (2.8)$$

Determinada a função corrente, obtemos as componentes do vento não divergente (empregando diferenças centradas):

$$\begin{aligned} u^n &= -\psi_y^n \\ v^n &= \psi_x^n. \end{aligned}$$

Portanto em cada passo no tempo será necessário:

- i) Resolver a equação de Poisson (2.8) na esfera (em malhas localmente refinadas). Para isso vamos utilizar um método multigrid na esfera, baseado em Barros 1991, o qual adaptamos para malhas com refinamento local (veja também Brandt 1977 e Brandt e Diskin 1994).
- ii) Conhecida a função corrente, calcular o vento não divergente e determinar as trajetórias das partículas, de uma maneira eficiente em malhas com refinamento local (nos baseamos em Ritchie 1987).
- iii) Calcular o valor da vorticidade total no novo tempo nos pontos da malha, usando interpolação cúbica, para avaliar a vorticidade no tempo anterior nos pontos de partida (2.7).

Detalharemos estas etapas nas seções seguintes.



## 2.3 Discretização da equação de Poisson na esfera

Para a discretização da equação de Poisson na esfera empregaremos diferenças finitas e uma formulação integral (cf. Barros 1991) para o tratamento das singularidades nos polos.

Em coordenadas esféricas temos o domínio

$$\Omega^* = \{(\lambda, \theta) : 0 \leq \lambda \leq 2\pi, -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}\},$$

e a equação de Poisson (supondo o raio constante igual a 1) escreve-se como

$$\Delta u = \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} (\cos \theta) \frac{\partial u}{\partial \theta} + \frac{1}{\cos^2 \theta} \frac{\partial^2 u}{\partial \lambda^2} = f. \quad (2.9)$$

Definimos a malha computacional  $\Omega_h^*$  com o mesmo espaçamento nas duas direções ( $\Delta \lambda = \Delta \theta = h$ ):

$$\Omega_h^* = \{(\lambda_i, \theta_j) : 1 \leq i \leq N_\lambda, 1 \leq j \leq N_\theta\},$$

com  $\lambda_i = (i-1)h$  e  $\theta_j = -\frac{\pi}{2} + (j-1)h$  onde  $h = \frac{2\pi}{N_\lambda}$ . Como tomamos a malha de forma que os polos pertencem a ela, então  $N_\theta = \frac{N_\lambda}{2} + 1$ . (Para  $\theta = -\frac{\pi}{2}$  e  $\theta = \frac{\pi}{2}$ , toda a linha da malha corresponde a um único ponto, o polo norte ou o polo sul).

A discretização da equação em  $(\lambda_i, \theta_j)$ , um ponto da malha diferente do polo, é baseada na seguinte discretização dos termos da equação (2.9)

$$\begin{aligned} \left( \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} (\cos \theta) \frac{\partial u}{\partial \theta} \right)_{(i,j)} &\approx \frac{1}{h^2 \cos \theta_j} \left( \cos \theta_{j+\frac{1}{2}} (u_{ij+1} - u_{ij}) - \cos \theta_{j-\frac{1}{2}} (u_{ij} - u_{ij-1}) \right), \\ \left( \frac{1}{\cos^2 \theta} \frac{\partial^2 u}{\partial \lambda^2} \right)_{(i,j)} &\approx \frac{1}{\cos^2 \theta_j} \left( \frac{u_{i+1j} - 2u_{ij} + u_{i-1j}}{h^2} \right), \end{aligned}$$

onde  $\theta_{j+\frac{1}{2}} = \theta_j + \frac{h}{2}$ ,  $\theta_{j-\frac{1}{2}} = \theta_j - \frac{h}{2}$  e  $u_{ij} = u(\lambda_i, \theta_j)$ .

Obtemos a seguinte discretização para a equação de Poisson

$$\begin{aligned} \frac{1}{h^2 \cos \theta_j} \left( \cos \theta_{j+\frac{1}{2}} (u_{ij+1} - u_{ij}) - \cos \theta_{j-\frac{1}{2}} (u_{ij} - u_{ij-1}) \right) + \\ \frac{1}{\cos^2 \theta_j} \left( \frac{u_{i+1j} - 2u_{ij} + u_{i-1j}}{h^2} \right) = f_{ij}. \end{aligned}$$

Multiplicando por  $h^2$  obtemos

$$\begin{aligned} & \frac{\cos \theta_{j+\frac{1}{2}}}{\cos \theta_j} u_{ij+1} + \frac{\cos \theta_{j-\frac{1}{2}}}{\cos \theta_j} u_{ij-1} + \frac{1}{\cos^2 \theta} u_{i+1j} + \\ & \frac{1}{\cos^2 \theta} u_{i-1j} - \left( \frac{\cos \theta_{j-\frac{1}{2}}}{\cos \theta_j} + \frac{\cos \theta_{j+\frac{1}{2}}}{\cos \theta_j} + \frac{2}{\cos^2 \theta_j} \right) u_{ij} = h^2 f_{ij} \end{aligned}$$

Em notação compacta, temos um operador linear de cinco pontos:

$$\begin{bmatrix} dn \\ dw & dc & de \\ ds \end{bmatrix}$$

onde

$$\begin{aligned} dn &= \frac{\cos \theta_{j+\frac{1}{2}}}{\cos \theta_j} \\ ds &= \frac{\cos \theta_{j-\frac{1}{2}}}{\cos \theta_j} \\ dw &= \frac{1}{\cos^2 \theta} \\ de &= \frac{1}{\cos^2 \theta} \\ dc &= -(dn + ds + dw + de) \end{aligned}$$

Este operador não está formalmente definido no polo (onde os termos são singulares). A discretização neste caso é apresentada a seguir.

### 2.3.1 Discretização nos polos

A discretização nos polos é obtida a partir da forma integral da equação. Integrando (2.9) numa calota polar de raio  $\rho = \frac{h}{2}$ , obtemos

$$\int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \Delta u \cos \theta d\theta d\lambda = \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} \left( \cos \theta \frac{\partial u}{\partial \theta} \right) \cos \theta d\theta d\lambda = \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} f \cos \theta d\theta d\lambda$$

(o outro termo do Laplaciano se anula pela periodicidade em  $\lambda$ ). Logo

$$\int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \frac{\partial}{\partial \theta} \left( \cos \theta \frac{\partial u}{\partial \theta} \right) d\theta d\lambda = \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} f \cos \theta d\theta d\lambda$$

Integrando com respeito a  $\theta$  obtemos:

$$-\cos\left(\frac{\pi}{2} - \rho\right) \int_{-\pi}^{\pi} \frac{\partial u}{\partial \theta} d\lambda = \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} f \cos \theta d\theta d\lambda.$$

Aproximando a integral no primeiro termo por n-trapézios, usando diferença finitas no cálculo da derivada e a regra do ponto médio para a integral no lado direito obtemos

$$\cos\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} h \frac{(u_{\lambda_i, \frac{\pi}{2}-\rho} - u_{P_N})}{h} = f_{P_N} \frac{\pi}{2} h \cos\left(\frac{\pi}{2} - \rho\right),$$

onde  $u_{P_N}$  representa o valor de  $u$  no polo norte. Assim, chegamos à seguinte discretização para a equação no polo norte:

$$\cos\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} (u_{\lambda_i, \frac{\pi}{2}-\rho} - u_{P_N}) = f_{P_N} \frac{\pi}{2} h \cos\left(\frac{\pi}{2} - \rho\right),$$

ou seja

$$u_{P_N} = \frac{1}{N_\lambda} \sum_{i=1}^{N_\lambda} u_{\lambda_i, \frac{\pi}{2}-\rho} - f_{P_N} \frac{h^2}{4}.$$

Analogamente para o polo sul  $P_S$  temos

$$u_{P_S} = \frac{1}{N_\lambda} \sum_{i=1}^{N_\lambda} u_{\lambda_i, \theta_2} - f_{P_S} \frac{h^2}{4}.$$

Este tratamento nos polos, combinado com diferenças centrais no restante da malha, nos leva a uma discretização global de segunda ordem.

## 2.4 Método Multigrid

Nesta seção apresentamos um método tipo FAS Multigrid (*Full Approximation Scheme*, Brandt, 1977), para a resolução da equação de Poisson na esfera, em malhas com refinamento local. Este método é baseado no algoritmo de Barros 1991 para malhas uniformes.

Introduzimos uma seqüência de malhas uniformes na esfera  $\Omega^1, \Omega^2, \dots, \Omega^M$  ( $\Omega^k = \Omega_{h_k}$ ), onde  $h_k = h_{k-1}/2$  e  $h_M = h$  (como na figura 2.2), sendo  $\Omega^M$  a malha mais fina onde queremos resolver a equação e  $\Omega^1$  uma malha grossa, com pequeno numero de pontos. Para cada malha  $\Omega^k$ , temos o correspondente problema discretizado

$$L^k u_i^k = f_i^k \quad i \in \Omega^k. \quad (2.10)$$

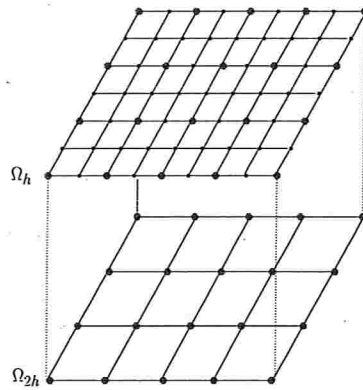


Figura 2.2: Níveis sucessivos de malhas para Multigrid.

onde  $L^k = L^{h_k}$  é o Laplaciano discretizado numa malha com espaçamento  $h_k$ ,  $u^k = u^{h_k}$  e  $f^k = f^{h_k}$ . Para  $k = 1$  o sistema (2.10) envolverá poucas incógnitas, sendo portanto fácil e barato de resolver.

Descrevemos a seguir o algoritmo FAS Multigrid, em forma geral, para a resolução da equação na malha mais fina  $\Omega^M$ , com o auxílio das malhas intermediárias (veja também Hackbusch para mais detalhes sobre métodos multigrid).

A partir de uma solução aproximada  $\tilde{u}^k$  para a solução na malha  $\Omega^k$ , cada ciclo de multigrid  $FAS_k(\nu_0, \nu_1, \nu_2; \gamma)$  é definido através das etapas (compare com a figura 2.3 para  $\gamma = 1$  (V-ciclo) e  $M = 4$ ,  $\gamma = 2$  corresponde a um W-ciclo) seguintes. Lembremos que a escolha correta dos componentes do método multigrid é fundamental para o desempenho e eficiência do mesmo.

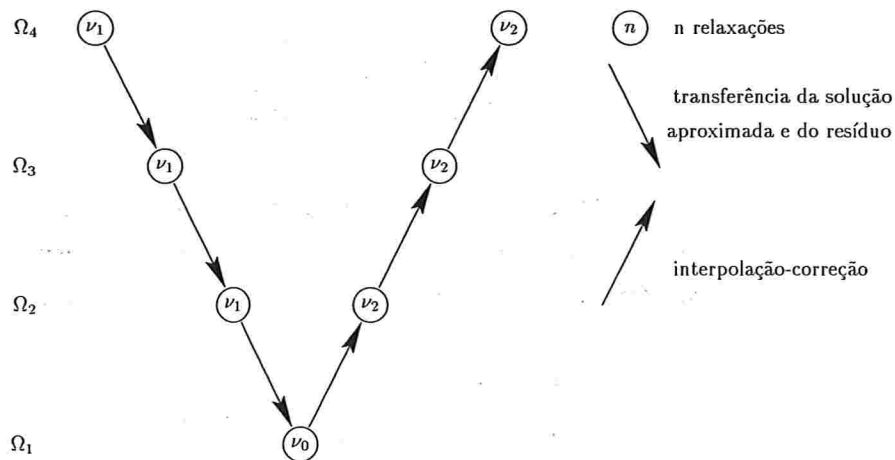


Figura 2.3: Diagrama de um  $V(\nu_1, \nu_2)$ -ciclo de FAS Multigrid ( $\gamma = 1$ ).

As etapas do nosso esquema são:

- i) *Pre-relaxação*. São aplicadas  $\nu_1$  relaxações à solução aproximada  $\tilde{u}^k$ .

Um dos principais componentes do Multigrid é a relaxação. Uma das características da equação de Poisson na esfera é a forte anisotropia do operador em coordenadas esféricas. É por isso que um método de relaxação padrão (como Gauss-Seidel, ponto a ponto) não proporciona a suavização necessária, sendo preciso recorrer a outros métodos neste caso. Nós trabalhamos com Gauss-Seidel por linhas, alternadas (relaxação tipo zebra), no qual os valores de toda uma linha de pontos de malha são atualizados simultaneamente (um sistema tridiagonal 'periódico' é resolvido para cada linha). Este método tem propriedades de relaxação muito boas para este problema, como analisado em Barros 1991.

- ii) *Transferência da solução*. Nós temos agora uma nova aproximação  $\tilde{u}^k$  na malha  $\Omega^k$  (com erro suavizado) e queremos melhorá-la através de uma correção empregando a próxima malha grossa  $\Omega^{k-1}$ . Na formulação FAS, transferimos a solução  $\tilde{u}^k$  para a malha grossa usando uma simples *injeção*, ou seja, os valores na malha grossa são os mesmos que nos pontos coincidentes da malha fina. Denotaremos esta aproximação na malha grossa por  $\hat{I}_k^{k-1}\tilde{u}^k$ , definida por

$$(\hat{I}_k^{k-1}\tilde{u}^k)_{i,j} = \tilde{u}_{2i,2j}^k.$$

- iii) *Transferência do resíduo*. Para a formação da equação de correção em  $\Omega^{k-1}$  necessitamos transferir para a malha grossa  $\Omega^{k-1}$  a função residual  $r^k = f^k - L^k\tilde{u}^k$ . Para a transferência do resíduo, usamos um operador *full-weighting* transformado, o qual denotamos por  $I_k^{k-1}$ . Pelo fato de estarmos trabalhando sobre a esfera, definimos o operador *full-weighting* em cada ponto da malha como

$$\frac{1}{4} \begin{bmatrix} \frac{\sin \theta_{j-1}}{\sin \theta_j} & 2 & \frac{\sin \theta_{j+1}}{\sin \theta_j} \\ 2 \frac{\sin \theta_{j-1}}{\sin \theta_j} & 4 & 2 \frac{\sin \theta_{j+1}}{\sin \theta_j} \\ \frac{\sin \theta_{j-1}}{\sin \theta_j} & 2 & \frac{\sin \theta_{j+1}}{\sin \theta_j} \end{bmatrix}.$$

Este operador leva em conta as diferentes áreas das células em torno de cada ponto de malha (cf. Barros 1991). Nos polos precisamos de uma definição especial do operador. Sendo  $P$  o polo, a transferência do resíduo para o polo na malha grossa é dada pela média ponderada:

$$2 \frac{\sin h_\theta/2}{\sin h_\theta} r(P) + \frac{4}{N_\lambda} \sum_{k=1}^{N_\lambda} r(\lambda_k, \bar{\theta}),$$

onde  $\bar{\theta} = -\frac{\pi}{2} + h_\theta$  se estamos considerando  $P$  o polo sul, e  $\bar{\theta} = \frac{\pi}{2} - h_\theta$  se  $P$  é o polo norte.  $N_\lambda$  é o número de pontos de malha por linha de latitude.

iv) *Equação na malha grossa.* Na malha grossa teremos a seguinte equação

$$L^{k-1}u^{k-1} = \tilde{f}^{k-1} \tag{2.11}$$

onde

$$\tilde{f}^{k-1} = L^{k-1}(\widehat{I}_k^{k-1}\tilde{u}^k) + I_k^{k-1}r^k.$$

v) *Iteração.* Se  $k - 1 = 1$  então (2.11) será resolvida. Na prática usaremos um número  $\nu_0$  de passos de relaxação para obter uma boa aproximação na malha mais grossa.

Se  $k - 1 > 1$  teremos que iniciar, tomando como aproximação inicial  $\widehat{I}_k^{k-1}\tilde{u}^k$ ,  $\gamma$  ciclos de  $FAS_{k-1}(\nu_0, \nu_1, \nu_2; \gamma)$  e obtemos uma solução melhorada para (2.11).

Em quaisquer dos casos, denotaremos a solução aproximada final por  $\tilde{u}^{k-1}$ .

vi) *Transferência e correção.* Por meio de um interpolador linear  $I_{k-1}^k$  da malha  $\Omega^{k-1}$  para a malha  $\Omega^k$  transferimos a correção para  $\tilde{u}^k$  calculada na malha grossa para a malha fina mais próxima. A correção da solução aproximada em  $\Omega^k$  é dada por:

$$\tilde{u}^k \leftarrow \tilde{u}^k + I_{k-1}^k(\tilde{u}^{k-1} - \widehat{I}_k^{k-1}\tilde{u}^k).$$

vii) *Pós-relaxação.* Finalmente, melhoramos a solução aproximada  $\tilde{u}^k$ , através de  $\nu_2$  relaxações como no passo (i).

Esta formulação é adequada não só para problemas lineares, senão também para problemas não lineares, já que a correção foi formulada sem depender da linearidade de  $L$ . Além disso, esta formulação se adequa a malhas localmente refinadas.

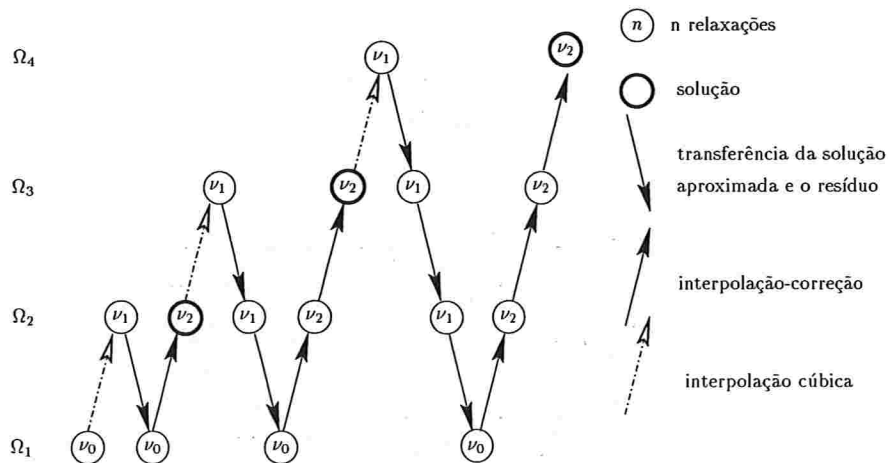


Figura 2.4: Diagrama de de Full-Multigrid.

## 2.4.1 Full-Multigrid

Para obter a aproximação inicial  $\tilde{u}^k$  com a qual começamos o algoritmo  $FAS_k(\nu_0, \nu_1, \nu_2; \gamma)$ , usamos um método full-multigrid (FMG).

Para  $k \geq 2$  o algoritmo  $FMG_k(\nu_0, \nu_1, \nu_2; \gamma)$  para a resolução de (2.10), pode ser descrito nos seguinte quatro passos (ver figura 2.4).

i) *Equação na malha grossa.* Temos a equação

$$L^{k-1}u^{k-1} = f^{k-1} \quad (2.12)$$

onde

$$f^{k-1} = I_k^{k-1} f^k$$

com  $I_k^{k-1}$  sendo o operador *injeção* de transferência.

ii) *Iteração.* Se  $k - 1 = 1$ , (2.12) é resolvida diretamente. (Em nosso caso estaremos usando  $\nu_0$  relaxações para obter uma solução aproximada suficientemente boa nessa malha).

Se  $k - 1 > 1$ , aplicamos o algoritmo  $FMG_{k-1}(\nu_0, \nu_1, \nu_2; \gamma; n)$  para resolver (2.12) e denotamos a solução obtida por  $\tilde{u}^{k-1}$ .

iii) *Solução interpolada.* Interpolamos  $\tilde{u}^{k-1}$  para  $\Omega^k$  usando um operador interpolação cúbica, o qual denotamos por  $\Pi_{k-1}^k$ .

iv) *Melhora e ciclos.* Melhoramos nossa solução aproximada na malha  $\Omega^k$ ,  $\tilde{u}^k = \Pi_{k-1}^k \tilde{u}^{k-1}$ , através de  $n$  ciclos de  $FAS_k(\nu_0, \nu_1, \nu_2; \gamma)$ .

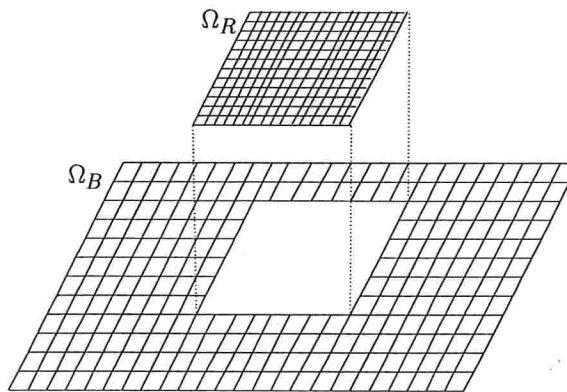


Figura 2.5: FAS multigrid em malhas com refinamento local.

### 2.4.2 FAS Multigrid em malhas com refinamento local

Para simplificar chamaremos  $\Omega_B$  à malha base e  $\Omega_R$  ao refinamento local como na figura 2.5.

Escolhemos a formulação FAS Multigrid, pois esta é facilmente adaptável a malhas com refinamento local. Tomando o algoritmo FAS Multigrid descrito, observemos quais os cuidados necessários para adaptá-lo a malhas com refinamento local.

Primeiro observemos o passo (iv) na descrição do algoritmo para FAS Multigrid. Na hora de construir a função  $\tilde{f}$  para a malha  $\Omega_B$ , ela só será modificada naqueles pontos, desta malha, que também pertençam ao refinamento local. Os outros não serão modificados. Quando temos que adicionar a correção na malha  $\Omega_B$ , no caso em que não são pontos dos refinamentos obteremos a solução aproximada. Nos pontos pertencentes ao refinamento, teremos a correção que depois será interpolada para a malha  $\Omega_R$ .

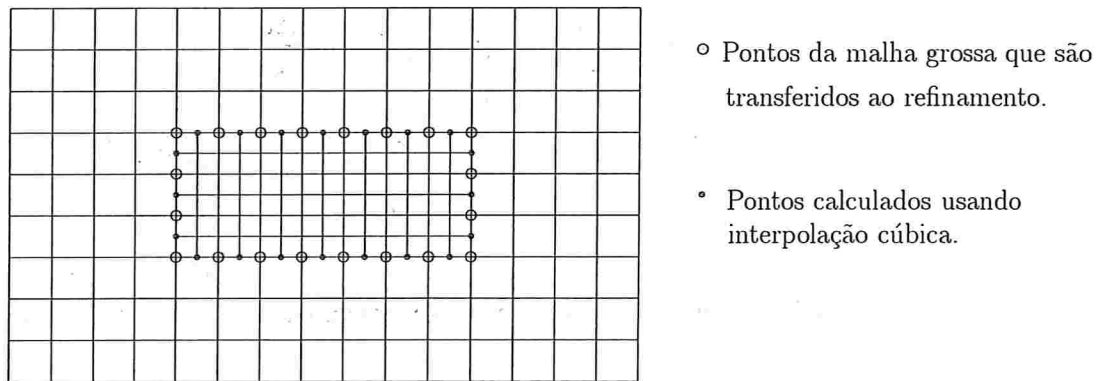


Figura 2.6: Pontos fantasmas no refinamento.

O principal cuidado deve ser tomado na relaxação nos refinamentos. As condições de fronteira (periódicas na malha base) são substituídas por condições de Dirichlet nas interfaces entre as malhas. Na figura 2.6 podemos observar como isso é feito; nós construímos uma fronteira no refinamento, adicionando linhas de 'pontos fantasmas'. Estes valores são obtidos da malha base por transferência nas posições onde existem pontos coincidentes na malha base e por interpolação cúbica nos restantes.

## 2.5 Cálculo da trajetória Lagrangeana na esfera

Para o cálculo das trajetórias implementamos o algoritmo de Ritchie (1987), adaptado para um método semi-Lagrangeano de dois níveis no tempo (e não de três como naquele trabalho). Incluímos aqui a descrição do método.



Seja  $r$  a trajetória da partícula que chega à posição  $r(t + \Delta t)$ , queremos calcular a posição da partícula no tempo  $t$ , ou seja  $r(t)$ . Para isso, primeiro vamos aproximar a posição da partícula  $r(t + \frac{\Delta t}{2})$  no ponto médio da trajetória usando expansão em série de Taylor

$$r(t + \Delta t) = r(t + \frac{\Delta t}{2}) + \frac{\Delta t}{2} \dot{r}(t + \frac{\Delta t}{2}) + O(\Delta t^2)$$

onde  $\dot{r}(t + \frac{\Delta t}{2})$  é o vetor velocidade de  $r$  no ponto médio da trajetória. Seja  $r(t + \Delta t) = g$  (um ponto da malha) então

$$r(t + \frac{\Delta t}{2}) = g - \frac{\Delta t}{2} \dot{r}(t + \frac{\Delta t}{2}) + O(\Delta t^2). \quad (2.13)$$

Esta equação pode levar a uma posição da partícula  $r(t + \frac{\Delta t}{2})$  ligeiramente deslocada da superfície da esfera (com erro  $O(\Delta t^2)$ ). Vamos então introduzir um fator de correção  $b$  no cálculo de  $r(t + \frac{\Delta t}{2})$  que assegure que  $|r(t + \frac{\Delta t}{2})| = a$ , onde  $a$  é o raio da esfera (em nossa aplicação, igual ao raio da terra). Ou seja, modificando a equação (2.13) por

$$r(t + \frac{\Delta t}{2}) = b \{ g - (\frac{\Delta t}{2}) \dot{r}(t + \frac{\Delta t}{2}) \} \quad (2.14)$$

onde

$$b = [1 + [\frac{\Delta t}{2}]^2 \dot{r}(t + \frac{\Delta t}{2})^2 / a^2 - 2(\frac{\Delta t}{2}) \dot{r}(t + \frac{\Delta t}{2}) \cdot g / a^2]^{-1/2},$$

garantimos uma trajetória sobre um grande círculo na esfera.

A equação (2.14) é resolvida iterativamente, uma vez que a velocidade  $\dot{r}(t + \frac{\Delta t}{2})$  depende da posição no ponto médio da trajetória. Podemos escrever a equação iterativa como

$$r_k = b_k (g - (\frac{\Delta t}{2}) \dot{r}_k).$$

Sejam  $(\lambda_g, \theta_g)$  as coordenadas esféricas do ponto de malha  $g$  aonde chega a partícula e  $(\lambda_k, \theta_k)$  as coordenadas de  $r_k$ , a posição media da trajetória na iteração  $k$ . Usamos as velocidades normalizadas da partícula para o cálculo do ponto médio da trajetória no passo  $k$ , ou seja

$$\begin{aligned} \tilde{u}_k &= u(\lambda_k, \theta_k, t + \frac{\Delta t}{2}) / a \\ \tilde{v}_k &= v(\lambda_k, \theta_k, t + \frac{\Delta t}{2}) / a \end{aligned}$$

Sendo  $(X, Y, Z)$  o sistema de coordenadas normalizado  $(x/a, y/a, z/a)$ , as componentes cartesianas da velocidade normalizada são dadas por

$$\begin{aligned} \dot{X}_k &= -\tilde{u}_k \sin(\lambda_k) - \tilde{v}_k \cos(\lambda_k) \sin(\theta_k) \\ \dot{Y}_k &= \tilde{u}_k \cos(\lambda_k) - \tilde{v}_k \sin(\lambda_k) \sin(\theta_k) \\ \dot{Z}_k &= \tilde{v}_k \cos(\theta_k). \end{aligned}$$

A posição do ponto da malha  $g$  em coordenadas cartesianas é

$$\begin{aligned} X_g &= \cos(\lambda_g) \cos(\theta_g) \\ Y_g &= \sin(\lambda_g) \cos(\theta_g) \\ Z_g &= \sin(\theta_g), \end{aligned}$$

portanto, o fator de correção  $b$  é dado por

$$b_k = \{1 + [\frac{\Delta t}{2}]^2 (\dot{X}_k^2 + \dot{Y}_k^2 + \dot{Z}_k^2) - \Delta t (\dot{X}_k X_g + \dot{Y}_k Y_g + \dot{Z}_k Z_g)\}^{-1/2}$$

e usando a posição do ponto médio da trajetória da partícula em coordenadas cartesianas temos:

$$\begin{aligned} X_{k+1} &= b_k (X_g - (\frac{\Delta t}{2}) \dot{X}_k) \\ Y_{k+1} &= b_k (Y_g - (\frac{\Delta t}{2}) \dot{Y}_k) \\ Z_{k+1} &= b_k (Z_g - (\frac{\Delta t}{2}) \dot{Z}_k). \end{aligned}$$

Daí obtemos

$$\begin{aligned} \lambda_{k+1} &= \tan^{-1}(Y_{k+1}/X_{k+1}) \\ \theta_{k+1} &= \sin^{-1}(Z_{k+1}), \end{aligned}$$

que são as coordenadas esféricas da posição da partícula no tempo  $t + \frac{\Delta t}{2}$ , ponto médio da trajetória que no tempo  $t + \Delta t$  chega ao ponto de malha  $(\lambda_g, \theta_g)$ . Mais à frente discutiremos um pouco sobre o número de iterações necessárias no cálculo do ponto médio das trajetórias.

Conhecida a posição  $r(t + \frac{\Delta t}{2})$  calculamos

$$r(t) = 2[r(t + \frac{\Delta t}{2}) \cdot g/a^2]r(t + \frac{\Delta t}{2}) - g,$$

em coordenadas  $(X, Y, Z)$  e obtemos a posição de partida da partícula.

Precisamos calcular separadamente os pontos de partida referente aos polos, uma vez que lá não estão definidas as componentes do vetor velocidade. Adotamos a solução de McDonald e Bates (1989), na qual usa-se o valor do vento nas vizinhanças do polo.

Tomemos o polo norte e suponhamos que o valor do vetor vento seja constante em sua vizinhança. Suas componentes podem ser escritas como:

$$u_p = V_p \sin(\lambda - \lambda_0) \quad (2.15)$$

$$v_p = V_p \cos(\lambda - \lambda_0) \quad (2.16)$$

onde  $V_p$  é a magnitude do vento e  $\lambda_0$  determina sua direção.

Identificamos (2.16) com a primeira componente de Fourier de  $v(\lambda_i, (\pi/2) - (\Delta\theta/2))$ , ou seja, a média de  $v$  na latitude mais próxima ao polo. Portanto temos

$$V_p = (a^2 + b^2)^{1/2},$$

$$\lambda_0 = \tan^{-1}(b/a)$$

onde

$$a = \frac{2}{N} \sum_{i=1}^N v^{n+1/2}(\lambda_i, \frac{\pi}{2} - \frac{\Delta\theta}{2}) \cos(\lambda_i)$$

$$b = \frac{2}{N} \sum_{i=1}^N v^{n+1/2}(\lambda_i, \frac{\pi}{2} - \frac{\Delta\theta}{2}) \sin(\lambda_i)$$

e  $N$  é o número de pontos no círculo de latitude.

Então o ponto de partida para o polo norte correspondente ao intervalo de tempo  $\Delta t$  é dado por

$$\lambda_* = \tan^{-1}(b/a)$$

$$\theta_* = \frac{\pi}{2} - \frac{V_p \Delta t}{a},$$

onde  $a$  é o raio da terra. Analogamente obteríamos para o polo sul

$$\lambda_* = \pi + \tan^{-1}(b/a)$$

$$\theta_* = -\frac{\pi}{2} + \frac{V_p \Delta t}{a}.$$

Observemos que no cálculo das trajetórias requer-se os valores da velocidade fora dos pontos de malha. Estes valores são obtidos por interpolação linear dos valores na malha. Estes por sua vez, são valores no instante de tempo intermediário  $t + \frac{\Delta t}{2}$ , obtidos por extrapolação linear dos valores na malha nos instantes  $t$  e  $t - \Delta t$ .

Para inicializar o método, dispomos apenas da velocidade inicial no instante 0. Para obtermos  $v(\frac{1}{2}\Delta t)$  procedemos iterativamente, tomando

$$v^{(k)}(\frac{1}{2}\Delta t) = \frac{1}{2}(v(0) + v^{(k-1)}(\Delta t)),$$

inicializando  $v^{(0)}(\Delta t) = v(0)$ . Cerca de 3 iterações já produzem uma boa aproximação ( veja Temperton e Staniforth, 1987).

## 2.6 Resultados

Aqui vamos discutir os resultados obtidos e mais alguns detalhes de implementações. Iniciaremos com os resultados do multigrid para a resolução da equação de Poisson, tanto na malha uniforme como na refinada. A seguir apresentamos resultados com o modelo de vorticidade barotrópica.

### 2.6.1 Multigrid

Inicialmente tomamos a equação de Helmholtz

$$-\Delta u + cu = f \quad (2.17)$$

onde  $c \geq 0$ , prescrevendo a solução para avaliarmos os erros de discretização. Tomando  $c > 0$  (escolhemos  $c = 0.5$  nos exemplos a seguir) garantimos a unicidade de solução.

Os resultados da tabela 2.1 mostram o máximo erro obtido em diversas malhas, após um ciclo de Full-Multigrid  $FMG(8, 1, 1, 1)$  (ou seja, usando um  $V(1,1)$  ciclo), para solução da equação de Helmholtz em um caso em que conhecemos a solução exata. Os resultados evidenciam a segunda ordem da discretização utilizada.

Malhas	Erro
16x9	4.6057898684988D-02
32x17	1.0968329948209D-02
64x33	2.8496029430405D-03
128x65	7.2522523815742D-04
256x129	1.8252035617241D-04

Tabela 2.1: Erros máximos em diferentes malhas uniformes (solução conhecida).

Na tabela 2.2 apresentamos o fator de convergência com um  $V(1, 1)$ -ciclo em diversas malhas. Aqui observa-se a 'independência de  $h$ ' dos fatores de convergência (fator assintótico de redução de erro por ciclo) que são limitados por valores muito menores que 1.

Malhas localmente refinadas são empregadas para reduzir o erro local. Suponha que temos uma região onde o erro é maior. Podemos refinar a malha para obter uma resolução melhor nesta região e assim diminuir o erro geral. Este foi o critério que usamos neste caso, refinando

Ciclos	fator de convergência
16 × 9	3.7321297999919D-02
32 × 17	4.1458496259644D-02
64 × 33	5.3080272602437D-02
128 × 65	5.9822717679057D-02
256 × 129	6.3554371747164D-02
512 × 257	7.0980011543551D-02

Tabela 2.2: Fator de convergência para diferentes malhas.

por um fator dois a região da malha onde o erro era maior. Na tabela 2.3 podemos comparar o erro em uma malha base de  $64 \times 33$  pontos com o erro na região refinada. Também observamos como o resíduo cai no refinamento durante ciclos de multigrid.

Ciclos	Erro malha base	Erro refinamento	Resíduo refinamento
1	2.8496029430405D-03	9.7048263104841D-04	1.5428768829396D-03
2	2.8126073810204D-03	5.2180850256320D-04	3.1669067734764D-04
3	2.8132302714346D-03	6.2271858306136D-04	5.7925075414552D-05
4	2.8131243625625D-03	6.3525761300647D-04	2.6844481346741D-05
5	2.8130989212079D-03	6.3243817921785D-04	2.7696890667159D-06

Tabela 2.3: Erros máximos na malha base de  $64 \times 33$  pontos. Também apresentamos o erro e resíduo na região refinada.

Na execução do programa para a equação

$$-\Delta u = f \quad (2.18)$$

ou seja, a equação (2.17) com  $c = 0$ , que aplicamos na equação da vorticidade barotrópica, não temos convergência da relaxação, (no sentido do resíduo ir a zero), pois o operador neste caso é singular.

Só podemos garantir a existência de solução de  $L_h u_h = f_h$  se estiver satisfeita a condição de compatibilidade de que  $f_h$  seja ortogonal ao núcleo do operador adjunto de  $L_h$ .

Esta condição é a versão discreta (a menos de um erro de ordem  $O(h^2)$ , veja Barros 1991) da condição de que

$$\int_{\Omega} f = 0.$$

Perturbando  $f$  ligeiramente pode-se garantir a verificação da condição de compatibilidade discreta e portanto a existência de solução que como no caso contínuo será única a menos de

uma constante.

Após feito o ajuste no lado direito temos essencialmente o mesmo comportamento do método multigrid também no caso  $c = 0$ .

Na tabela 2.4 a 2.6 apresentamos para o caso  $c = 0$  os resultados análogos aos das tabela 2.2 e 2.3 (para a mesma solução da equação). Obtemos ainda ótima convergência e erros de discretização de segundo ordem.

Malhas	Erro
16x9	4.8593272488308D-02
32x17	1.1596127189235D-02
64x33	3.0310870147363D-03
128x65	7.7330530851422D-04
256x129	1.9493199673834D-04

Tabela 2.4: Erros máximos em diferentes malhas ( $c=0$ ).

Ciclos	fator de convergência
16 × 9	4.6014782681578D-02
32 × 17	4.3723098120074D-02
64 × 33	5.3661985293763D-02
128 × 65	5.9884032804054D-02
256 × 129	6.3466572186454D-02
512 × 257	7.1049670422945D-02

Tabela 2.5: Fator de convergência para diferentes malhas ( $c=0$ ).

Ciclos	Erro malha base	Erro refinamento	Resíduo refinamento
1	3.0310870147363D-03	6.6607726944534D-04	1.1498994844704D-03
2	3.0193178983498D-03	3.7124429758631D-04	1.2579785018838D-04
3	3.0238946099224D-03	3.7233036611301D-04	7.5456346730601D-06
4	3.0240268108957D-03	3.7254553441499D-04	1.3783231250994D-06
5	3.0240237158246D-03	3.7252967464649D-04	6.8420681696713D-08

Tabela 2.6: Erros máximos na malha base de  $64 \times 33$  pontos. Também apresentamos o erro e resíduo na região refinada ( $c=0$ ).

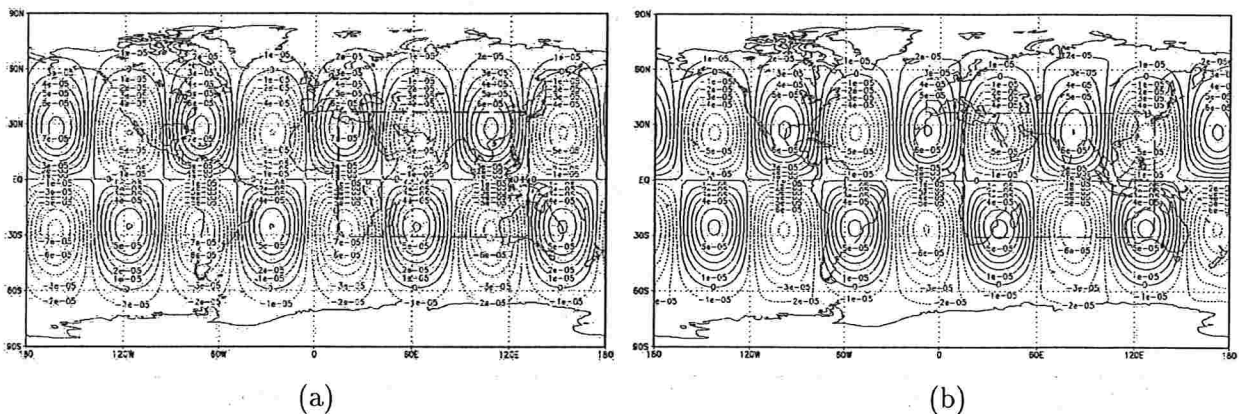
### 2.6.2 Vorticidade barotrópica semi-Lagrangeana

Em todas as experiências realizadas para este modelo estamos usando uma malha base de  $128 \times 65$  pontos.

Nos exemplos seguintes usamos  $\Delta t = 1h$  e empregamos um ciclo de Full-Multigrid  $FMG(8, 1, 1, 1)$  (ou seja, usando um  $V(1,1)$  ciclo) para resolver a equação de Poisson a cada passo no tempo. Cabe lembrar que por estarmos usando um método semi-Lagrangeano não temos restrições no passo no tempo devidas a condições de CFL para estabilidade (um modelo euleriano numa malha  $128 \times 65$  pontos iria requerer  $\Delta t$  da ordem de 30 minutos, e com um nível de refinamento um  $\Delta t$  de cerca de 15 minutos).

Para o primeiro teste escolhemos como dado inicial uma Onda de Haurwitz (Phillips, 1959). Trata-se de um teste simples, mas adequado para se testar a estabilidade. Pelo padrão definido da solução podemos observar possíveis deformações da solução aproximada causadas pela presença do refinamento local. Sobre uma malha de  $128 \times 65$  pontos colocamos um refinamento local e rodamos o algoritmo durante 10 dias.

Na figura 2.7 observamos no gráfico (a) a vorticidade aproximada depois de 5 dias de integração numa malha com um nível de refinamento local (um refinamento de fator dois sobre uma malha base de  $128 \times 65$  pontos). A região do refinamento também está indicada no gráfico. No gráfico (b), temos a vorticidade aproximada depois de 10 dias.



Adotamos como vorticidade inicial (ver figura 2.8), uma função suave que foi perturbada, através da inclusão de uma gaussiana, centrada em  $\lambda = 120^\circ$  e  $\theta = 50^\circ$  (Hortal et al. 1991). A função suave vai rodar ao redor do polo levando consigo a perturbação. Depois de 18h de integração a perturbação passa por cima do polo norte. Nas figura 2.9 na coluna (b), podemos observar uma secção através do polo da vorticidade em determinados tempos, todos numa malha uniforme bem fina, de  $1024 \times 513$  pontos, que nos servirá de referência e será usada como solução 'exata' nas comparações de resultados. Na figura 2.9 vemos na primeira linha o resultado obtido depois de 18h de integração e na segunda linha e na terceira depois de 36h e 54h.

Agora na malha base de  $128 \times 65$  pontos testamos o programa com diferente número de iterações no cálculo do ponto médio da trajetória. Usamos o resultado depois de 54h e fizemos um corte na função 'exata' na altura do máximo valor, na latitude= $50^\circ$  o qual mostramos na figura 2.10. Nos gráficos seguintes veremos os resultados depois de 54h de integração e comparamos as diferentes soluções com a função 'exata'. Na primeira linha vemos a vorticidade aproximada na coluna (a) e na coluna (b) um corte da vorticidade 'exata' (latitude= $50^\circ$ ) e da vorticidade aproximada com uma iteração para o cálculo do ponto médio da trajetória. Na segunda linha temos os resultados correspondentes utilizando duas iterações no cálculo da trajetória e na terceira linha os mesmos resultados obtidos com dez iterações para o cálculo da trajetória.

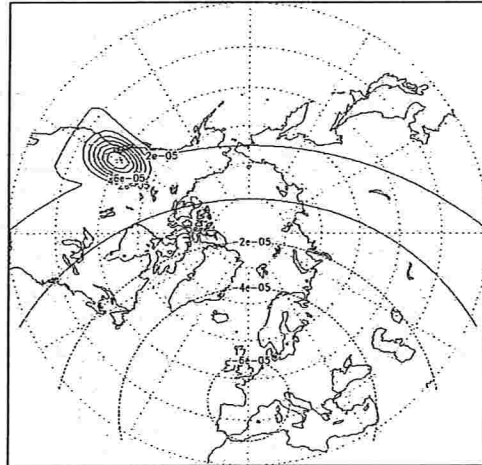


Figura 2.8: Projeção estereográfica no polo norte do dado inicial.

Não temos diferenças significativas no exemplo usando duas ou dez iterações. No caso de usar só uma, podemos observar uma perda na precisão. Portanto consideramos que duas iterações no cálculo do ponto médio da trajetória são suficientes e é o que usaremos nos exemplos a seguir.



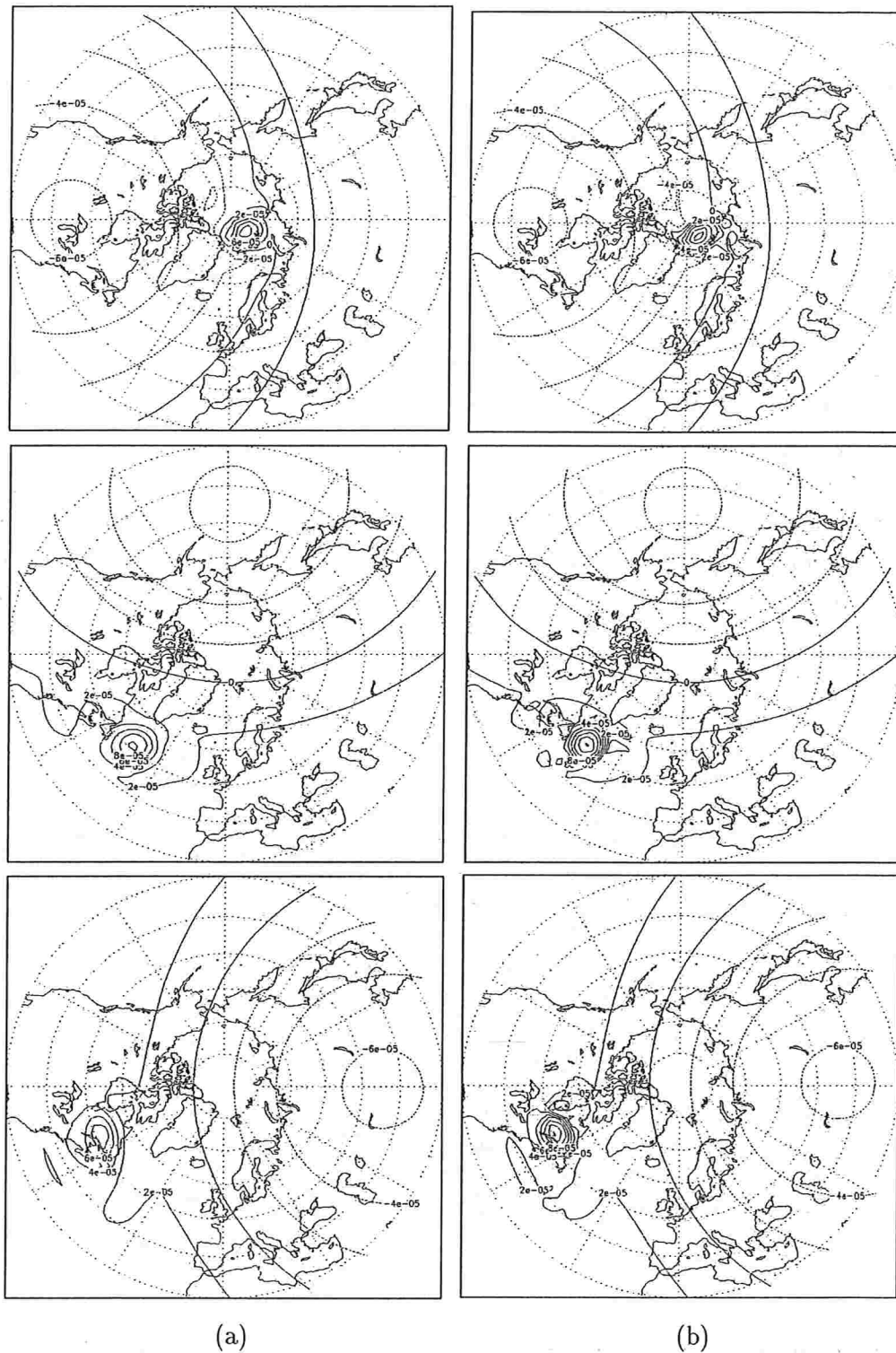


Figura 2.9: Na coluna (a) temos os resultados obtidos usando a malha base de 128x65 pontos para 18h, 36h e 54h. Na coluna (b) temos os resultados obtidos nos mesmos tempos na malha mais fina de  $1024 \times 513$  pontos.

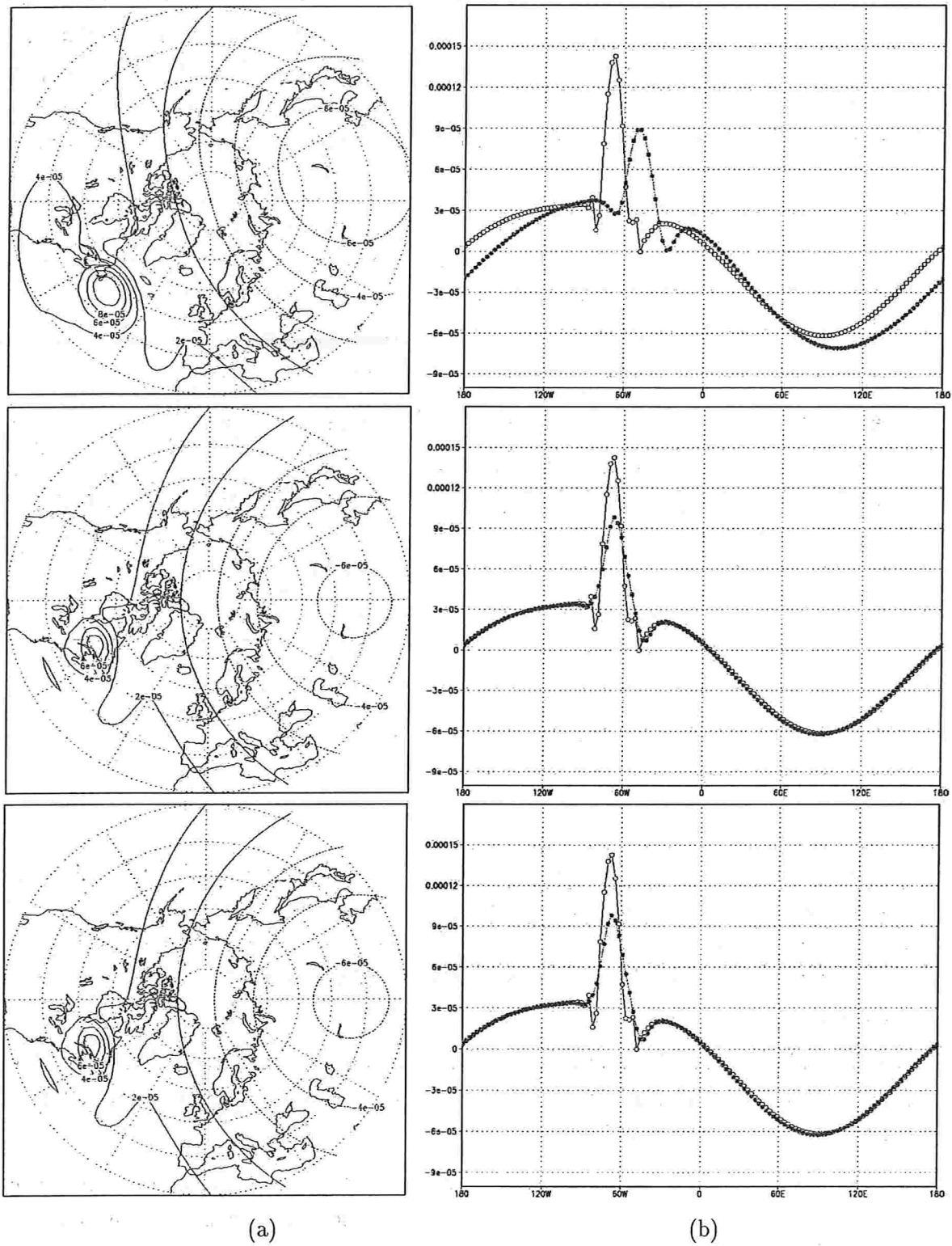


Figura 2.10: Na coluna (a) temos a vorticidade aproximada usando 1,2 e 10 iterações para o cálculo da trajetória, depois de 54h. Na coluna (b) temos o corte transversal da função 'exata' e as funções aproximadas calculadas usando diferente numero de iterações.

O passo seguinte é comparar os resultados com a malha uniforme e a malha refinada. Usamos de novo uma malha base de 128x65 pontos, na qual testamos dois exemplos de refinamentos. Primeiramente refinamos uma pequena região centrada no máximo da perturbação. A perturbação depois de poucas iterações sai do refinamento e não volta. No segundo exemplo temos um refinamento um pouco maior de onde a perturbação sai, voltando a entrar na região refinada posteriormente.

Na figura 2.11 podemos observar o dado inicial e os dois refinamentos usados nos testes. Depois de 54h. de integração, temos na figura 2.12, na coluna (a) a vorticidade aproximada e na coluna (b) o corte transversal da vorticidade 'exata' e a aproximada na latitude = 50° a latitude onde a vorticidade 'exata' toma o valor máximo.

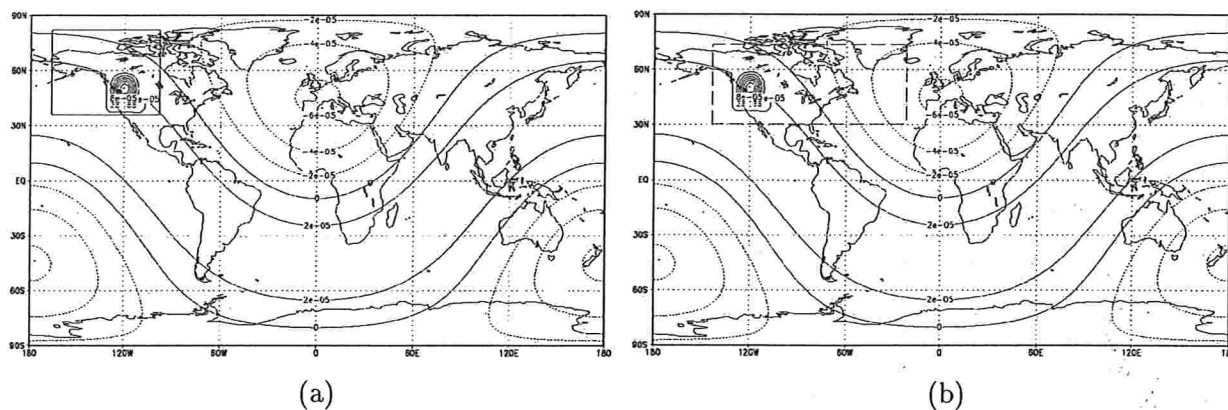


Figura 2.11: Dado inicial em todo o globo, e os dois refinamento usados.

Podemos observar na primeira linha os resultados obtidos usando uma malha sem refinamento local, ou seja usando só a malha base. Na segunda linha podemos observar os resultados obtidos com uma malha como mostrada na figura 2.11(a). Neste caso a perturbação que inicialmente se encontra dentro da malha refinada, depois de poucas iterações sai do refinamento, não entrando nele novamente.

Na terceira linha da figura 2.12 temos os resultados com a malha mostrada em 2.11(b). Neste caso a perturbação que inicialmente se encontra dentro do refinamento depois de poucas iterações sai para entrar de novo no final da integração.

Podemos observar que no caso do refinamento menor, onde a perturbação permanece poucas iterações dentro do refinamento, a presença do refinamento local não melhora muito a solução final, comparada com o caso em que só usamos a malha base. Afora isso podemos observar que a saída do refinamento é suave, não distorcendo a solução em relação à solução na malha uniforme.

Na segunda malha refinada, a perturbação sai fora do refinamento local após poucas iterações, retornando depois. Podemos observar o efeito positivo na solução aproximada depois de 54h de integração, causado pela maior permanência dentro do refinamento local, como esperado.

No capítulo 4 apresentaremos mais resultados com o uso de refinamento local neste modelo e introduziremos a adaptatividade, com a possibilidade do refinamento acompanhar a trajetória da Gaussiana.

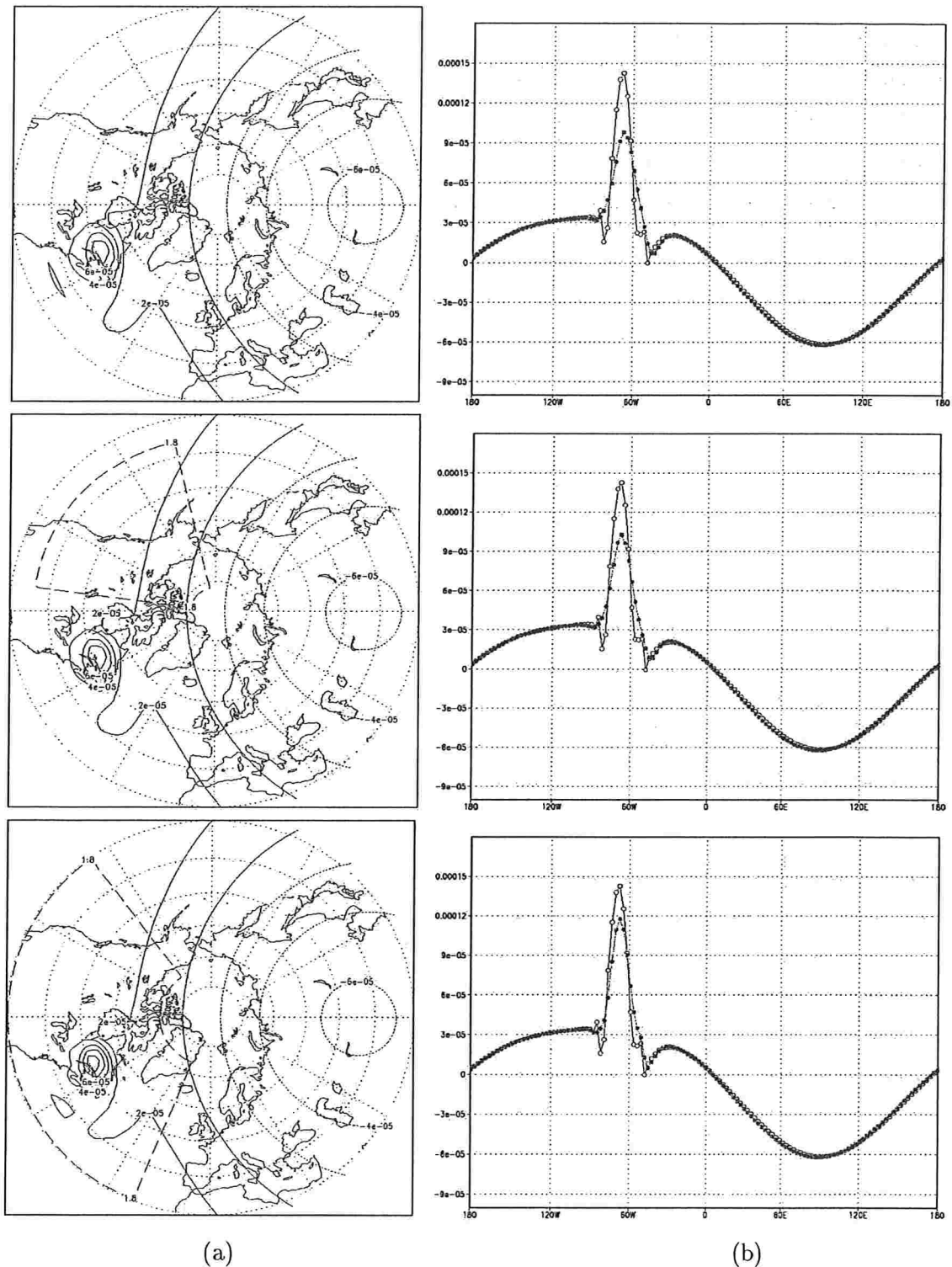


Figura 2.12: Resultados obtidos nas diferentes malhas depois de 54h. Na coluna (a) temos a projeção estereográfica da vorticidade aproximada e na coluna (b) o corte transversal desta e a vorticidade 'exata' na latitude  $50^\circ$ .



---

# Equações de Água-Rasa

Este capítulo, parte central desta tese, apresenta dois aspectos fundamentais. Primeiramente há o desenvolvimento de um modelo global de diferenças finitas, semi-implícito, semi-Lagrangeano em dois níveis no tempo para as equações de água-rasa na esfera.

Este modelo, que desenvolvemos, é uma variante do modelo proposto por Bates et al. (1990). A principal diferença é que adotamos um tratamento totalmente implícito do termo de divergência de massa na equação da continuidade, o qual escrevemos em função do logaritmo do geopotencial. Desta maneira, podemos eliminar o termo de 'divergence damping' utilizado por eles, o qual era necessário para a estabilização do modelo em algumas situações. Com essa formulação obtemos um método estável sem precisar da adição de termos de filtragem seletiva. (Formulação semelhante para a equação de massa foi empregada por Temperton e Staniforth 1987 em um modelo regional). O preço a pagar por esta formulação é a necessidade de resolver uma equação não linear, para a qual desenvolvemos um método multigrid com eficiência semelhante a de um '*fast Poisson solver*'. Além disso, adotamos o tratamento semi-Lagrangeano de Ritchie (1988), que consideramos mais simples que o usado por Bates et. al. (1990).

O segundo aspecto, que é a principal proposta do trabalho, é o uso de técnicas de refinamento local neste modelo. Os vários aspectos algorítmicos e computacionais do emprego de refinamento local, foram gradativamente estudados e desenvolvidos através da hierarquia de modelos que selecionamos e apresentamos nos capítulos anteriores. Apresentamos aqui o modelo global de água-rasa com técnicas de refinamento local, mostrando que essa é uma alternativa interessante para a obtenção de alta resolução em uma região de interesse, a baixo custo computacional e sem envolver o problema do tratamento de condições de fronteira artificiais, uma vez que mantemos um modelo global.

### 3.1 Aspectos gerais do modelo

Podemos escrever as equações de água-rasa na esfera, em sua forma vetorial, como (veja Haltiner e Willians 1980):

$$\frac{dV}{dt} + fk \times V + \nabla_H \phi = 0$$

$$\frac{d \log(\phi)}{dt} + \nabla_H \cdot V = 0$$

onde  $V = (u, v)$  é o vetor de velocidades,  $\phi$  o geopotencial ( $\phi = gz$ ),  $f = 2\Omega \sin \theta$  o fator de Coriolis e  $\frac{d}{dt} = \frac{\partial}{\partial t} + V \cdot \nabla_H$  é a derivada total. Este se constitui em um problema a valores iniciais para  $V$  e  $\phi$ .

A forma geral da discretização temporal semi-implícita, semi-Lagrangeana de dois níveis no tempo para as equações de água-rasa que propomos é a seguinte (na forma vetorial). A partir dos valores de  $V$  e  $\phi$  no instante inicial, calcula-se a evolução temporal com passos no tempo de  $\Delta t$  através das equações:

de momento:

$$\left( V + \frac{\Delta t}{2} fk \times V + \frac{\Delta t}{2} \nabla \phi \right)^{n+1} = \left( V - \frac{\Delta t}{2} fk \times V - \frac{\Delta t}{2} \nabla \phi \right)_*^n \quad (3.1)$$

da continuidade:

$$\left( \log(\phi) + \frac{\Delta t}{2} D \right)^{n+1} = \left( \log(\phi) - \frac{\Delta t}{2} D \right)_*^n \quad (3.2)$$

onde  $V$  é o campo de velocidades,  $\phi$  o geopotencial,  $D$  o divergente ( $D = \nabla \cdot V$ ),  $f$  o fator de Coriolis e o símbolo \* se refere à posição de partida de uma partícula no instante  $n$  que chega num ponto de grade no instante  $n + 1$ , do método semi-Lagrangeano (ver capítulos 1 e 2).

Para a determinação das incógnitas no instante  $n + 1$  temos que resolver o sistema resultante. Este processo envolve as seguintes etapas:

- i) Cálculo das trajetórias das partículas e determinação dos pontos de partida \* no instante  $n$ .
- ii) Cálculo do lado direito das equações (3.1) e (3.2) nos pontos de malha e interpolação para seus respectivos pontos de partida.
- iii) Redução do sistema a uma equação escalar (não linear) para o geopotencial e sua solução.
- iv) Atualização do vetor velocidade usando as equações de momento.

A seguir apresentamos os detalhes das partes do algoritmo, iniciando por uma flexibilização na discretização.



### 3.2 Discretização temporal

A discretização temporal semi-implícita, semi-Lagrangeana (na forma vetorial) de dois níveis no tempo para as equações de água-rasa que adotamos engloba a possibilidade de pequena descentralização temporal. Temos então as

Equações de momento:

$$(V + (1 - \epsilon)\Delta t f k \times V + (1 - \epsilon)\Delta t \nabla \phi)^{n+1} = (V - \epsilon \Delta t f k \times V - \epsilon \Delta t \nabla \phi)_*^n$$

Equação da continuidade:

$$(\log(\phi) + (1 - \epsilon)\Delta t D)^{n+1} = (\log(\phi) - \epsilon \Delta t D)_*^n$$

onde o símbolo \* se refere à posição de partida de uma partícula no instante  $n$  que chega num ponto de grade no instante  $n + 1$  e  $\epsilon$  satisfaz  $1/2 - \delta \leq \epsilon \leq 1/2$ , (se  $\epsilon = 1/2$ , nós temos as equações completamente centradas tipo 'Crank-Nicolson'). Uma pequena descentralização da discretização é em certos casos adequada para supressão de ruído, tornando o esquema ligeiramente dissipativo.

A derivada total é aproximada pela diferença ao longo da trajetória e o termo de Coriolis e o gradiente do geopotencial são aproximados pela média ao longo da trajetória.

Para simplificar a notação, definimos

$$F_n = (1 - \epsilon)\Delta t f \quad e \quad F_v = \epsilon \Delta t f$$

e obtemos

$$(V + F_n k \times V + (1 - \epsilon)\Delta t \nabla \phi)^{n+1} = (V - F_v k \times V - \epsilon \Delta t \nabla \phi)_*^n, \quad (3.3)$$

e

$$(\log(\phi) + (1 - \epsilon)\Delta t D)^{n+1} = (\log(\phi) - \epsilon \Delta t D)_*^n. \quad (3.4)$$

A formulação das equações de momento em coordenadas esféricas para as componentes da velocidade leva à aparição dos chamados termos métricos, devido à geometria da esfera. Estes termos tomam valores cada vez maiores próximo aos polos (ver McDonald e Bates 1987), sendo fonte de instabilidade em modelos semi-Lagrangeanos que usam as equações nessa forma. A forma correta para o tratamento semi-Lagrangeano das equações é conservar a formulação vetorial das equações (Ritchie 1988, Côte e Staniforth 1988, Bates et. al 1990).

Adotamos aqui a formulação proposta por Ritchie, por sua elegância e simplicidade. Esta não requer um tratamento especial na vizinhança dos polos, como em Bates et. al (1990) e é mais simples que o esquema de Côte e Staniforth (1988). Todas elas produzem resultados similares (veja Staniforth e Côte 1991).

Nas equações de momento temos que igualar dois vetores que pertencem a diferentes sistemas de referência, um no ponto de partida e outro no ponto de chegada da trajetória. Estes dois vetores são tangentes à esfera. Para compará-los faremos a projeção dos dois vetores no plano tangente ao ponto médio da trajetória e assim com os dois vetores no mesmo plano poderemos resolver as equações.

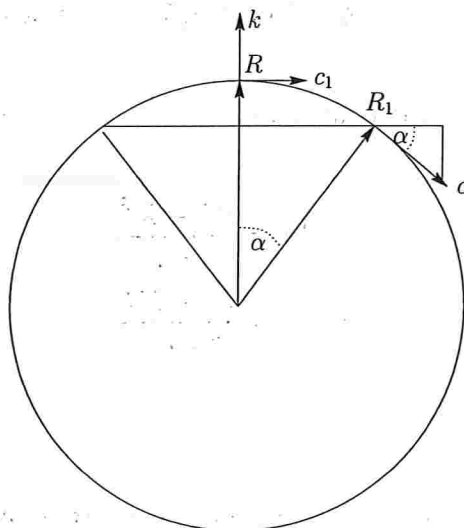


Figura 3.1: Diagrama dos vetores ao longo da trajetória.

Apresentamos aqui a descrição geral desta transformação, deixando uma descrição mais detalhada para o apêndice B. Definimos  $\alpha$  como o deslocamento angular ao longo do grande círculo de trajetória entre dois pontos  $R$  e  $R_1$ , onde  $R$  é o ponto médio da trajetória e  $R_1$  o ponto de partida ou o ponto de chegada da mesma (os cálculos são análogos nos dois casos). Sendo  $h_1$  o vetor horizontal no plano tangente gerado por  $(\hat{\lambda}_1, \hat{\theta}_1)$  em  $R_1$ , queremos transformá-lo para o plano tangente gerado por  $(\hat{\lambda}, \hat{\theta})$  no ponto  $R$  (ver figura 3.1)

Primeiro definimos o vetor  $c$ , como um vetor unitário na direção do vento, no ponto médio da trajetória  $R$ , e  $n$  o vetor unitário ortogonal à esquerda de  $c$  (portanto  $c$  e  $n$  formam outra base do plano gerado por  $(\hat{\lambda}, \hat{\theta})$ ).

No plano tangente em  $R_1$  podemos escrever o vetor horizontal  $h_1$  como

$$h_1 = h_\lambda \hat{\lambda}_1 + h_\theta \hat{\theta}_1$$

na posição  $R_1$ . Queremos transformar este vetor para o plano Cartesiano tangencial  $R$ , o que equivale a achar um vetor neste novo plano, com a mesma grandeza de  $h_1$  e tal que a direção seja a direção projetada de  $h_1$  neste plano.

Isto é feito primeiro expressando  $\hat{\lambda}_1$  e  $\hat{\theta}_1$  em função de  $\mathbf{c}_1$  e  $\mathbf{n}_1$ , onde  $\mathbf{n}_1 = \mathbf{n}$  (pois considera-se a trajetória circular) e  $\mathbf{c}_1$  é o vetor unitário na direção do vento no ponto  $R_1$ . A seguir faz-se a rotação de  $\mathbf{c}_1$  na direção de  $\mathbf{c}$ , levando a dois novos vetores unitários  $(\hat{\lambda}_1)_R$  e  $(\hat{\theta}_1)_R$ , ambos no plano tangente gerado por  $(\hat{\lambda}, \hat{\theta})$ , pois são combinação linear de  $\mathbf{c}$  e  $\mathbf{n}$ . Logo, o vetor

$$h_\lambda(\hat{\lambda}_1)_R + h_\theta(\hat{\theta}_1)_R$$

tem o mesmo módulo que  $h_1$ , e sua direção é a projeção da direção de  $h_1$  no plano  $(\hat{\lambda}, \hat{\theta})$ , conforme queríamos.

Finalmente escrevemos  $\mathbf{c}$  e  $\mathbf{n}$  em função de  $\hat{\lambda}$  e  $\hat{\theta}$  para encontrar as coordenadas do novo vetor neste plano

$$\begin{aligned} (\hat{\lambda}_1)_R &= \frac{Y\hat{\lambda} + X\hat{\theta}}{\cos\theta_{R_1}} \\ (\hat{\theta}_1)_R &= \frac{Y\hat{\theta} - X\hat{\lambda}}{\cos\theta_{R_1}} \end{aligned}$$

onde

$$\begin{aligned} X &= \sin\alpha \cos\gamma \sin\theta + (1 - \cos\alpha) \sin\gamma \cos\gamma \cos\theta, \\ Y &= \cos\theta - \sin\alpha \sin\gamma \sin\theta - (1 - \cos\alpha) \sin^2\gamma \cos\theta, \end{aligned}$$

$\gamma$  é o ângulo entre  $\mathbf{c}$  e  $\hat{\lambda}$ ,  $\alpha$  é o ângulo entre  $R$  e  $R_1$  (veja os detalhes no apêndice B).

Aplicando esta transformação às equações de momento (3.3) obtemos:

$$(X_u)^{n+1} \hat{\lambda}^{n+1} + (X_v)^{n+1} \hat{\theta}^{n+1} = (Y_u)_*^n \hat{\lambda}_*^n + (Y_v)_*^n \hat{\theta}_*^n$$

onde

$$\begin{aligned} X_u &= u - F_n v + \Delta t(1 - \epsilon) \frac{1}{a \cos\theta} \frac{\partial\phi}{\partial\lambda} \\ X_v &= v + F_n u + \Delta t(1 - \epsilon) \frac{1}{a} \frac{\partial\phi}{\partial\theta} \\ Y_u &= u + F_v v - \Delta t\epsilon \frac{1}{a \cos\theta} \frac{\partial\phi}{\partial\lambda} \\ Y_v &= v - F_v u - \Delta t\epsilon \frac{1}{a} \frac{\partial\phi}{\partial\theta}. \end{aligned}$$

Escrevemos  $\hat{\lambda}^{n+1}$ ,  $\hat{\theta}^{n+1}$ ,  $\hat{\lambda}_*^n$  e  $\hat{\theta}_*^n$  em função de  $\hat{\lambda}$  e  $\hat{\theta}$  (coordenadas do plano tangente no ponto médio da trajetória) e obtemos:

$$X_u^{n+1} = \frac{1}{\cos\theta^{n+1}} (Y^{n+1} B - X^{n+1} A) \quad (3.5)$$

$$X_v^{n+1} = \frac{1}{\cos \theta^{n+1}} (X^{n+1} A - Y^{n+1} B), \quad (3.6)$$

onde

$$B = \left( \frac{Y Y_u - X Y_v}{\sqrt{X^2 + Y^2}} \right)_*$$

e

$$A = \left( \frac{X Y_u + Y Y_v}{\sqrt{X^2 + Y^2}} \right)_*.$$

Portanto agora podemos isolar  $u$ ,  $v$  das equações de momento para calcular o divergente no instante  $n + 1$  e assim derivar a equação não linear para o geopotencial.

Definindo

$$\frac{1}{\cos \theta^{n+1}} (Y^{n+1} B - X^{n+1} A) = A_u,$$

$$\frac{1}{\cos \theta^{n+1}} (X^{n+1} A - Y^{n+1} B) = A_v,$$

podemos escrever a equação de momento resultante como:

$$\begin{pmatrix} 1 & -F_n \\ F_n & 1 \end{pmatrix} \begin{pmatrix} u^{n+1} \\ v^{n+1} \end{pmatrix} + (1 - \epsilon) \Delta t \nabla \phi^{n+1} = \begin{pmatrix} A_u \\ A_v \end{pmatrix}. \quad (3.7)$$

Isolando as componentes do vento na equação de momento obtemos:

$$\begin{pmatrix} u^{n+1} \\ v^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & -F_n \\ F_n & 1 \end{pmatrix}^{-1} \begin{pmatrix} A_u \\ A_v \end{pmatrix} - \begin{pmatrix} 1 & -F_n \\ F_n & 1 \end{pmatrix}^{-1} (1 - \epsilon) \Delta t \nabla \phi^{n+1}.$$

Definindo

$$A_\phi = (\log(\phi) - \epsilon \Delta t D)^n,$$

a equação da continuidade fica:

$$(\log(\phi) + (1 - \epsilon) \Delta t D)^{n+1} = A_\phi.$$

E como

$$\begin{pmatrix} 1 & -F_n \\ F_n & 1 \end{pmatrix}^{-1} = G \begin{pmatrix} 1 & F_n \\ -F_n & 1 \end{pmatrix},$$

onde  $G = 1/(1 + F_n^2)$  (e  $D = \nabla_H \cdot V$ ), podemos re-escrever a equação da continuidade como uma equação para o geopotencial

$$\log(\phi^{n+1}) + (1 - \epsilon) \Delta t \nabla_H \cdot \left( - \begin{pmatrix} 1 & -F_n \\ F_n & 1 \end{pmatrix}^{-1} (1 - \epsilon) \Delta t \nabla \phi^{n+1} \right) =$$

$$A_\phi - (1 - \epsilon) \Delta t \nabla_H \cdot \begin{pmatrix} 1 & -F_n \\ F_n & 1 \end{pmatrix}^{-1} \begin{pmatrix} A_u \\ A_v \end{pmatrix}.$$

Introduzindo as notações:

$$\begin{pmatrix} d_u \\ d_v \end{pmatrix} = \begin{pmatrix} 1 & F_n \\ -F_n & 1 \end{pmatrix}^{-1} \begin{pmatrix} A_u \\ A_v \end{pmatrix},$$

$$H^n = \nabla_H \cdot \begin{pmatrix} d_u \\ d_v \end{pmatrix},$$

e

$$\widetilde{d}_\phi = A_\phi - (1 - \epsilon)\Delta t H^n,$$

obtemos a equação não linear para o geopotencial dada por:

$$\log(\phi^{n+1}) - (1 - \epsilon)^2 \Delta t^2 \nabla_H \cdot G \begin{pmatrix} 1 & F_n \\ -F_n & 1 \end{pmatrix} \nabla \phi^{n+1} = \widetilde{d}_\phi. \quad (3.8)$$

Em coordenadas esféricas, temos

$$\log(\phi^{n+1}) - (1 - \epsilon)^2 \Delta t^2 \nabla_H \cdot G \begin{pmatrix} 1 & F_n \\ -F_n & 1 \end{pmatrix} \frac{1}{a} \begin{pmatrix} \frac{1}{\cos \theta} \frac{\partial \phi^{n+1}}{\partial \lambda} \\ \frac{\partial \phi^{n+1}}{\partial \theta} \end{pmatrix} = \widetilde{d}_\phi,$$

resultando em

$$\log(\phi^{n+1}) - \frac{(1 - \epsilon)^2 \Delta t^2}{a} \nabla_H \cdot G \begin{pmatrix} \frac{1}{\cos \theta} \frac{\partial \phi^{n+1}}{\partial \lambda} + F_n \frac{\partial \phi^{n+1}}{\partial \theta} \\ -\frac{F_n}{\cos \theta} \frac{\partial \phi^{n+1}}{\partial \lambda} + \frac{\partial \phi^{n+1}}{\partial \theta} \end{pmatrix} = \widetilde{d}_\phi.$$

A expressão do divergente é dada por:

$$\nabla_H \cdot \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \frac{1}{a \cos \theta} \left( \frac{\partial f_1}{\partial \lambda} + \frac{\partial \cos \theta f_2}{\partial \theta} \right),$$

levando a

$$\log(\phi^{n+1}) - \frac{(1 - \epsilon)^2 \Delta t^2}{a^2} \frac{1}{\cos \theta} \left( \frac{\partial}{\partial \lambda} \left( \frac{G}{\cos \theta} \frac{\partial \phi^{n+1}}{\partial \lambda} + F_n G \frac{\partial \phi^{n+1}}{\partial \theta} \right) + \frac{\partial}{\partial \theta} \left( -F_n G \frac{\partial \phi^{n+1}}{\partial \lambda} + \cos \theta G \frac{\partial \phi^{n+1}}{\partial \theta} \right) \right) = \widetilde{d}_\phi.$$

Lembrando que  $F_n$  e  $G$  só dependem de  $\theta$ , temos:

$$\log(\phi^{n+1}) - \frac{(1 - \epsilon)^2 \Delta t^2}{a^2} \left( \frac{G}{\cos^2 \theta} \frac{\partial^2 \phi^{n+1}}{\partial \lambda^2} + \frac{F_n G}{\cos \theta} \frac{\partial^2 \phi^{n+1}}{\partial \lambda \partial \theta} - \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} \left( F_n G \frac{\partial \phi^{n+1}}{\partial \lambda} \right) + \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} \left( G \cos \theta \frac{\partial \phi^{n+1}}{\partial \theta} \right) \right) = \widetilde{d}_\phi.$$

Com

$$\tilde{\mu} = \frac{a^2}{(1-\epsilon)^2 \Delta t^2} \quad e \quad d_\phi = \tilde{\mu} \tilde{d}_\phi,$$

temos

$$\begin{aligned} \tilde{\mu} \log(\phi^{n+1}) - \frac{G}{\cos^2 \theta} \frac{\partial^2 \phi^{n+1}}{\partial \lambda^2} - \frac{F_n G}{\cos \theta} \frac{\partial^2 \phi^{n+1}}{\partial \lambda \partial \theta} + \frac{F_n G}{\cos \theta} \frac{\partial^2 \phi^{n+1}}{\partial \lambda \partial \theta} \\ + \frac{1}{\cos \theta} \frac{\partial F_n G}{\partial \theta} \frac{\partial \phi^{n+1}}{\partial \lambda} - \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} \left( G \cos \theta \frac{\partial \phi^{n+1}}{\partial \theta} \right) = d_\phi. \end{aligned}$$

Portanto

$$\tilde{\mu} \log(\phi^{n+1}) - \frac{G}{\cos^2 \theta} \frac{\partial^2 \phi^{n+1}}{\partial \lambda^2} + \frac{1}{\cos \theta} \frac{\partial F_n G}{\partial \theta} \frac{\partial \phi^{n+1}}{\partial \lambda} - \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} \left( G \cos \theta \frac{\partial \phi^{n+1}}{\partial \theta} \right) = d_\phi. \quad (3.9)$$

Esta é a equação escalar para o geopotencial a que reduzimos o sistema. Esta tem que ser resolvida para que então obtenhamos  $V^{n+1}$  de (3.7).

### 3.3 Malha alternada

Empregamos uma malha alternada (staggered grid) de tipo C (ver Arakawa 1966 e a figura 3.4). Os valores do geopotencial e do vento, estão distribuídos como mostra a figura 3.2.

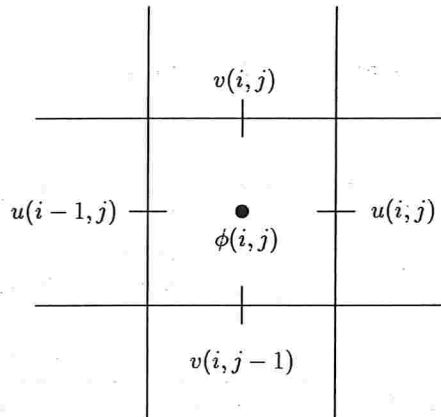


Figura 3.2: Desenho da malha alternada.

Com esta distribuição das variáveis, calculamos  $A_u$ ,  $A_v$  e  $A_\phi$  nas posições de  $u$ ,  $v$  e  $\phi$  respectivamente.

A maior parte das variáveis se encontra convenientemente localizada. Por exemplo

$$\left( \frac{\partial \phi}{\partial \lambda} \right)_{(i,j)_u} \cong \frac{\phi_{ij+1} - \phi_{ij}}{h},$$

é a aproximação centrada da derivada total do geopotencial no ponto  $(i, j)$  da malha para  $u$ .

Em alguns termos, como os de Coriolis, precisamos tomar médias de termos vizinhos, usando para tal as 4 (ou 16) posições mais próximas

Outra consequência importante do uso da malha alternada, é que agora teremos 3 trajetórias diferentes a serem calculadas, uma para os pontos de  $u$ , outra para  $v$  e também para  $\phi$ .

### 3.4 Discretização espacial da equação não linear para o geopotencial

A discretização espacial de (3.9) é feita através de diferenças finitas centradas, de segunda ordem, levando ao operador:

$$\begin{aligned} \tilde{\mu} \log(\phi_{ij}) - \frac{G_j}{\cos^2 \theta_j} \left( \frac{\phi_{i+1j} - 2\phi_{ij} + \phi_{i-1j}}{h^2} \right) + \left( \frac{1}{\cos \theta} \frac{\partial F_n G}{\partial \theta} \right)_j \left( \frac{\phi_{i+1j} - \phi_{i-1j}}{2h} \right) \\ - \frac{1}{\cos \theta_j h} \left[ \left( (G \cos)(\theta_{j+\frac{1}{2}}) \frac{\phi_{ij+1} - \phi_{ij}}{h} \right) - \left( (G \cos)(\theta_{j-\frac{1}{2}}) \frac{\phi_{ij} - \phi_{ij-1}}{h} \right) \right] = (d_\phi)_{ij}, \end{aligned}$$

tomando  $h = \Delta\lambda = \Delta\theta$ . Multiplicando por  $h^2$ , temos

$$\begin{aligned} \tilde{\mu} h^2 \log(\phi_{ij}) + \left( -\frac{G_j}{\cos^2 \theta_j} + \left( \frac{h}{2 \cos \theta} \frac{\partial F_n G}{\partial \theta} \right)_j \right) \phi_{i+1j} \\ + \left( -\frac{G_j}{\cos^2 \theta_j} - \left( \frac{h}{2 \cos \theta} \frac{\partial F_n G}{\partial \theta} \right)_j \right) \phi_{i-1j} \\ + \left( 2 \frac{G_j}{\cos^2 \theta_j} + \frac{(G \cos)(\theta_{j-\frac{1}{2}})}{\cos \theta_j} + \frac{(G \cos)(\theta_{j+\frac{1}{2}})}{\cos \theta_j} \right) \phi_{ij} \\ - \frac{(G \cos)(\theta_{j+\frac{1}{2}})}{\cos \theta_j} \phi_{ij+1} - \frac{(G \cos)(\theta_{j-\frac{1}{2}})}{\cos \theta_j} \phi_{ij-1} = (d_\phi)_{ij}. \end{aligned}$$

A parte linear do operador é um operador de cinco pontos

$$\begin{bmatrix} & dn & & & \\ dw & dc & de & & \\ & ds & & & \end{bmatrix}$$

onde

$$\begin{aligned} dn &= -\frac{(G \cos)(\theta_{j+\frac{1}{2}})}{\cos \theta_j} \\ ds &= -\frac{(G \cos)(\theta_{j-\frac{1}{2}})}{\cos \theta_j} \\ dw &= \left( -\frac{G_j}{\cos^2 \theta_j} - \left( \frac{h}{2 \cos \theta} \frac{\partial F_n G}{\partial \theta} \right)_j \right) \\ de &= \left( -\frac{G_j}{\cos^2 \theta_j} + \left( \frac{h}{2 \cos \theta} \frac{\partial F_n G}{\partial \theta} \right)_j \right) \\ dc &= -(dn + ds + dw + de) \end{aligned}$$

Este operador não está formalmente definido no polo (onde os termos são singulares). A seguir derivamos a discretização neste caso, passando para uma formulação integral como no capítulo anterior.

### 3.4.1 Discretização nos polos

A discretização nos polos é obtida a partir da forma integral de (3.9). Integrando esta equação numa calota polar de raio  $\rho = \frac{h}{2}$ , obtemos

$$\begin{aligned} \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \tilde{\mu} \log(\phi^{n+1}) \cos \theta d\theta d\lambda - \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \frac{1}{\cos \theta} \frac{\partial}{\partial \theta} \left( G \cos \theta \frac{\partial \phi^{n+1}}{\partial \theta} \right) \cos \theta d\theta d\lambda = \\ \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} d_\phi \cos \theta d\theta d\lambda, \end{aligned}$$

onde  $\tilde{\mu} = \frac{a^2}{(1-\epsilon)^2 \Delta t^2}$  e  $G = 1/(1 + F_n^2)$  com  $F_n = (1 - \epsilon)\Delta t f$  (os outros termos se anulam devido à periodicidade em  $\lambda$ ). Portanto

$$\int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} (\tilde{\mu} \log(\phi^{n+1}) - d_\phi) \cos \theta d\theta d\lambda - \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \frac{\partial}{\partial \theta} \left( G \cos \theta \frac{\partial \phi^{n+1}}{\partial \theta} \right) d\theta d\lambda = 0.$$

Agora, usando valor de  $d_\phi$  e integrando o segundo termo com respeito a  $\theta$ , temos:

$$\int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} (\tilde{\mu} \log(\phi^{n+1}) - \tilde{\mu} (A_\phi - \Delta t(1 - \epsilon)H^n)) \cos \theta d\theta d\lambda + (G \cos)\left(\frac{\pi}{2} - \rho\right) \int_{-\pi}^{\pi} \frac{\partial \phi^{n+1}}{\partial \theta} d\lambda = 0.$$

Substituindo o valor de  $H^n$  e usando a regra de n-trapézios para o último termo:

$$\int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \tilde{\mu} (\log(\phi^{n+1}) - A_\phi) \cos \theta d\theta d\lambda + \int_{-\pi}^{\pi} \int_{\frac{\pi}{2}-\rho}^{\frac{\pi}{2}} \tilde{\mu} \Delta t(1-\epsilon) \frac{1}{a \cos \theta} \left( \frac{\partial du}{\partial \lambda} + \frac{\partial \cos \theta dv}{\partial \theta} \right)^n \cos \theta d\theta d\lambda$$



$$+(G \cos)\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} \frac{\phi^{n+1}(P_N) - \phi^{n+1}(\lambda_i, \frac{\pi}{2} - \rho)}{h} h = 0,$$

onde  $P_N$  é o polo norte.

Com a regra do ponto médio para o primeiro termo e a periodicidade na direção de  $\lambda$  temos

$$\begin{aligned} & \tilde{\mu} (\log(\phi^{n+1}(P_N)) - A_\phi(P_N)) \cos\left(\frac{\pi}{2} - \rho\right) h \frac{\pi}{2} + \frac{\tilde{\mu} \Delta t}{a} (1 - \epsilon) \int_{-\pi}^{\pi} \int_{\frac{\pi}{2} - \rho}^{\frac{\pi}{2}} \frac{\partial \cos \theta dv^n}{\partial \theta} d\theta d\lambda \\ & + (G \cos)\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} \frac{\phi^{n+1}(P_N) - \phi^{n+1}(\lambda_i, \frac{\pi}{2} - \rho)}{h} h = 0, \end{aligned}$$

e integrando com respeito a  $\theta$  o segundo termo e aplicando a regra do trapézio chegamos a

$$\begin{aligned} & \tilde{\mu} (\log(\phi^{n+1}(P_N)) - A_\phi(P_N)) \cos\left(\frac{\pi}{2} - \rho\right) h \frac{\pi}{2} + \frac{\tilde{\mu} \Delta t (1 - \epsilon)}{a} \cos\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} dv^n(\lambda_i, \frac{\pi}{2} - \rho) h \\ & + (G \cos)\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} \phi^{n+1}(P_N) - \phi^{n+1}(\lambda_i, \frac{\pi}{2} - \rho) = 0. \end{aligned}$$

Multiplicando por  $h/\cos(\frac{\pi}{2} - \epsilon)\frac{\pi}{2}$ , obtemos a equação para  $\phi$  no polo norte

$$\begin{aligned} & \tilde{\mu} h^2 (\log(\phi^{n+1}(P_N)) - A_\phi(P_N)) + \frac{\tilde{\mu} \Delta t}{a} \frac{4(1 - \epsilon)h}{N_\lambda} \sum_{i=1}^{N_\lambda} dv^n(\lambda_i, \frac{\pi}{2} - \rho) \\ & + 4G\left(\frac{\pi}{2} - \rho\right) \left( \phi^{n+1}(P_N) - \frac{1}{N_\lambda} \sum_{i=1}^{N_\lambda} \phi^{n+1}(\lambda_i, \frac{\pi}{2} - \rho) \right) = 0. \end{aligned}$$

No polo sul respectivamente teremos

$$\begin{aligned} & \tilde{\mu} h^2 (\log(\phi^{n+1}(P_S)) - A_\phi(P_S)) - \frac{\tilde{\mu} \Delta t}{a} \frac{4(1 - \epsilon)h}{N_\lambda} \sum_{i=1}^{N_\lambda} dv^n(\lambda_i, -\frac{\pi}{2} + \rho) \\ & + 4G\left(-\frac{\pi}{2} + \rho\right) \left( \phi^{n+1}(P_S) - \frac{1}{N_\lambda} \sum_{i=1}^{N_\lambda} \phi^{n+1}(\lambda_i, -\frac{\pi}{2} + \rho) \right) = 0. \end{aligned}$$

### Cálculo do divergente no polo

Para o cálculo do operador não linear para o geopotencial no polo, precisaremos do valor de  $A_\phi$ , dado por

$$A_\phi = (\log(\phi) - \epsilon \Delta t D)_*^n,$$

que envolve o valor do divergente no polo. Para isso integraremos o divergente em uma calota polar de raio  $\rho = \frac{h}{2}$ , a qual chamaremos de  $\Omega$ .

Temos por exemplo no polo norte, usando a regra do ponto médio:

$$\int_{\Omega} D^n dA = a^2 D(P_N) h \cos\left(\frac{\pi}{2} - \rho\right) \frac{\pi}{2} \quad (3.10)$$

Por outro lado, pelo teorema do divergente,

$$\int_{\Omega} D^n dA = \int_{\Omega} \nabla \cdot V^n dA = \int_{\partial\Omega} V^n \cdot \eta \, dC = - \int_{\partial\Omega} v^n dC.$$

Usando a regra do trapézio temos

$$\int_{\partial\Omega} v^n dC = -ah \cos\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} v^n(\lambda_i, \frac{\pi}{2} - \rho),$$

ou seja

$$\int_{\Omega} D^n dA = -ah \cos\left(\frac{\pi}{2} - \rho\right) \sum_{i=1}^{N_\lambda} v^n(\lambda_i, \frac{\pi}{2} - \rho). \quad (3.11)$$

Igualando (3.10) e (3.11) obtemos que

$$D(P_N) = -\frac{1}{a\frac{\pi}{2}} \sum_{i=1}^{N_\lambda} v^n(\lambda_i, \frac{\pi}{2} - \rho)$$

Para o polo sul temos:

$$D(P_S) = \frac{1}{a\frac{\pi}{2}} \sum_{i=1}^{N_\lambda} v^n(\lambda_i, \frac{\pi}{2} - \rho).$$

Conhecido o valor do divergente no polo, podemos obter o valor de  $(\log(\phi) - \frac{\Delta t}{2} D)^n$  e assim interpolar cubicamente nos pontos de partida para obter o valor de  $A_\phi$ .

### 3.5 Cálculo das trajetórias

Pelo fato de estarmos usando a malha alternada (staggered do tipo C), temos que considerar três trajetórias diferentes. Na prática calculamos a trajetória para a malha correspondente ao geopotencial, e a partir desta vamos interpolar para assim obter os pontos de partida correspondentes as malhas para  $u$  e  $v$ .

No cálculo das trajetórias para o geopotencial, usamos o mesmo algoritmo utilizado na equação da vorticidade barotrópica (capítulo 2), baseado no trabalho de Ritchie (1987).

Para a obtenção dos pontos de partida das trajetórias das malhas para  $u$  e para  $v$  empregamos interpolação linear dos pontos de partida dos pontos da malha de  $\phi$  mais próximos. Por exemplo

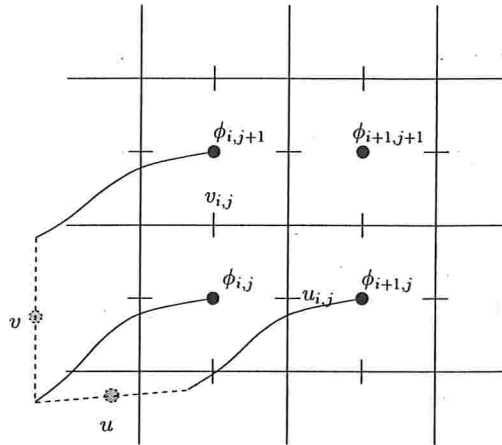


Figura 3.3: Cálculo das trajetórias para  $u$  e para  $v$ .

no cálculo do ponto de partida na malha para  $u$ , de  $u_{i,j}$ , interpolamos a posição dos pontos de partida de  $\phi_{i,j}$  e  $\phi_{i+1,j}$ . Quando queremos calcular o ponto de partida de  $v_{i,j}$ , interpolamos os pontos de partida de  $\phi_{i,j}$  e  $\phi_{i,j+1}$  (ver figura 3.3).

Temos que tomar cuidado especial nos cálculos dos pontos de partida da malha para  $v$  que ficam perto do polo.

Como mostra a figura 3.4, temos que calcular o valor da posição de partida para os pontos da malha  $v$ , que ficam ao lado dos polos, ou seja os pontos  $v(\lambda_i, \theta_{n_\theta-1})$  com  $i = 1, \dots, N_\lambda$ .

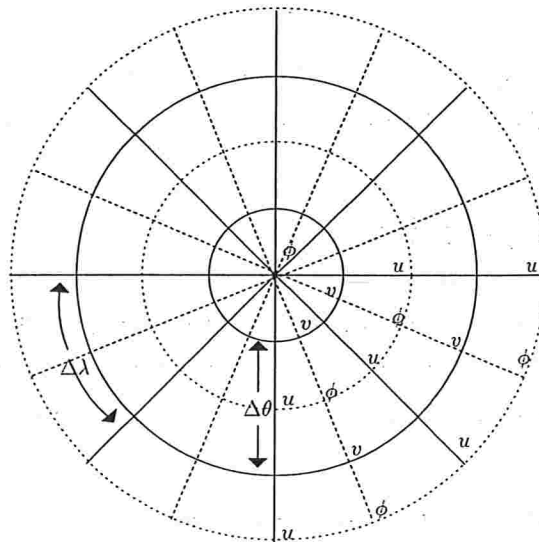


Figura 3.4: Esquema da malha  $C$  nas proximidades do polo.

Para tal transferimos o ponto de partida de  $\phi(P)$  e  $\phi((\lambda_i, \theta_{n_\theta-1}))$ , os dois pontos mais próximos a  $v(\lambda_i, \theta_{n_\theta-1})$ , para o plano tangente no polo. A interpolação é feita no plano tangente e projetada na esfera para obter-se a posição de partida de  $v(\lambda_i, \theta_{n_\theta-1})$ .

### 3.6 Resolução da equação não linear via Multigrid

No capítulo anterior desenvolvemos um algoritmo FAS Multigrid, para a resolução da equação de Poisson na esfera, em malhas com refinamento local, o qual utilizamos no modelo para a equação da vorticidade barotrópica.

A equação que precisamos resolver aqui é ainda uma equação elíptica, contendo termos de primeira ordem além do Laplaciano. Há ainda um termo não linear, sendo a equação da forma

$$\mu \log(\phi) + A\phi = f$$

onde  $A$  é um operador elíptico, a ser discretizado como mostrado na seção 3.4.

A vantagem do uso de método multigrid é sua aplicabilidade relativamente geral, com sua performance não dependendo de separabilidade do operador, coeficientes constantes, etc.

Adotamos aqui o mesmo multigrid desenvolvido para a equação de Poisson, sendo apenas necessário adaptar a relaxação devido à não linearidade da equação. (Note que não empregamos nenhuma linearização global e sim apenas localmente na relaxação). Adotamos a relaxação tipo zebra com Newton/Gauss-Seidel por linhas.

Na relaxação por linhas, todas as incógnitas pertencentes a um mesmo círculo de latitude são atualizadas simultaneamente. Enquanto que na equação de Poisson tínhamos apenas que resolver um sistema tridiagonal (periódico), aqui ainda há a não linearidade na diagonal. Usamos então um passo do método de Newton para resolver o sistema não linear associado a cada linha, supondo as demais incógnitas na malha conhecidas. Note que o sistema para cada linha não chega a ser resolvido exatamente (pois usamos apenas uma iteração do método de Newton), mas isto é suficiente para produzir a suavização do erro necessária à boa convergência do método. Nosso método multigrid neste caso tem comportamento muito similar ao método usado para a equação de Poisson.

### 3.7 Resultados

Começamos esta seção com investigações numéricas que comprovam o bom desempenho do método multigrid para a equação não linear. A seguir apresentaremos testes com nosso modelo de água-rasa em malhas uniformes, verificando também a estabilidade do esquema. A seguir

passaremos ao estudo numérico do modelo em malhas com refinamento local, comprovando a viabilidade computacional do uso de tal técnica para previsões regionais.

### 3.7.1 Multigrid não linear

Na tabela 3.1, em um exemplo com solução conhecida, podemos observar como varia o erro nas diferentes malhas com um passo de full-multigrid, evidenciando que a discretização é de segunda ordem.

Malhas	Erro
16x9	1.8563777899466D-02
32x17	4.5424480017546D-03
64x33	1.1390306627703D-03
128x65	2.9179802938195D-04

Tabela 3.1: Erros em diferentes malhas (solução exata conhecida).

Ciclos	Erro	Resíduo
1	4.5424480017546D-03	2.5027813606469D-04
2	4.7156030756934D-03	5.8369514471107D-06
3	4.7189999633437D-03	1.6665922365766D-07
4	4.7190971537683D-03	5.1466695072389D-09
5	4.7191161438285D-03	8.8874443739161D-10

Tabela 3.2: Malha de  $32 \times 17$ .

Ciclos	Erro	Resíduo
1	1.1390306627703D-03	3.2897329249865D-05
2	1.1810944545116D-03	5.5345182781857D-07
3	1.1819136941860D-03	1.8341904651359D-08
4	1.1819320006614D-03	8.0537383533030D-10
5	1.1819363044010D-03	7.0703056952625D-11

Tabela 3.3: Malha de  $64 \times 33$ .

A seguir apresentamos 3 tabelas nas quais poderemos ver o comportamento do erro e do resíduo, para vários ciclos de full-multigrid em diferentes malhas. Na tabela 3.2 temos uma malha de  $32 \times 17$  pontos, na tabela 3.3, uma malha de  $64 \times 33$  pontos e por último na tabela 3.4 uma malha de  $128 \times 65$  pontos.

Ciclos	Erro	Resíduo
1	2.9179802938195D-04	4.3623360836731D-06
2	3.0219435248746D-04	5.4828001236351D-08
3	3.0239241716412D-04	2.2547331545233D-09
4	3.0239613279992D-04	1.1273893450736D-10
5	3.0239689351319D-04	5.8626367649417D-12

Tabela 3.4: Malha de  $128 \times 65$ 

Os resultados comprovam a eficiência do método e só precisaremos de um ciclo de full-multigrid no modelo de água-rasa, para a obtenção do novo geopotencial. Este algoritmo, foi adaptado a malhas com refinamento local, como já descrito para a equação de vorticidade barotrópica.

### 3.7.2 Modelo de água-rasa

#### Malha uniforme

Apresentamos aqui alguns resultados obtidos com o modelo de água-rasa. Primeiro testamos a estabilidade do modelo através de 10 dias de integração em uma malha de alta resolução,  $1536 \times 769$ , com  $\Delta t = 1h$  (em modelos semi-implícitos Eulerianos a condição de CFL permitiria um  $\Delta t$  máximo de cerca de 2.5 minutos).

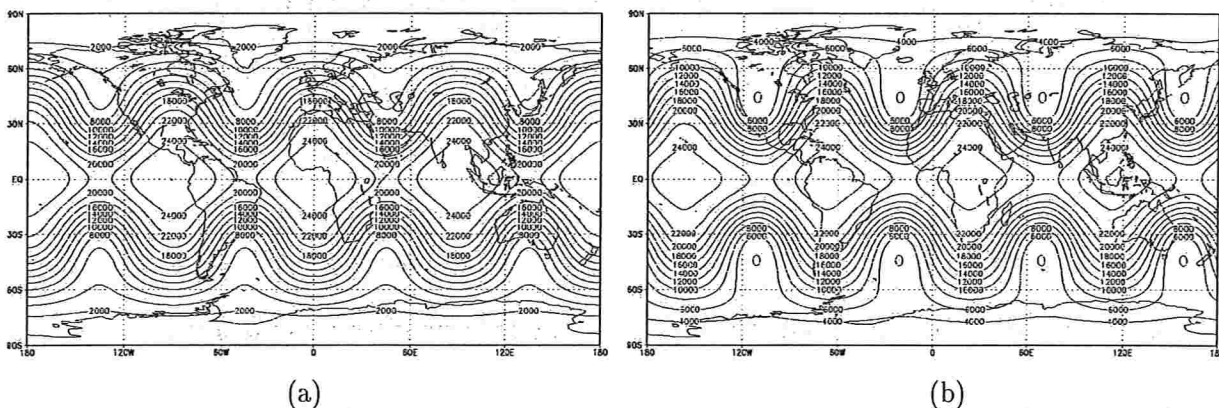


Figura 3.5: Malha  $1536 \times 769$ , (a) é o geopotencial inicial e (b) aos 10 dias com  $\Delta t = 1h$ .

Para o teste usamos uma Onda de Haurwitz, na figura 3.5 vemos o geopotencial inicial e depois de 10 dias de integração. Como podemos observar, a estabilidade do esquema é bastante robusta.

Consideraremos agora dados iniciais reais (obtidos de campos de vento e geopotencial a 300mb, de 9 de Janeiro de 1979). Na figura 3.8 apresentamos as isolinhas do campo geopotencial

inicial. A partir destas condições iniciais integramos o modelo utilizando várias resoluções com malhas uniformes. Nestes experimentos usamos  $\Delta t = 1h$  e empregamos um ciclo de Full-Multigrid  $FMG(8, 1, 1, 1)$  para resolver a equação não linear para o geopotencial, a cada passo no tempo.

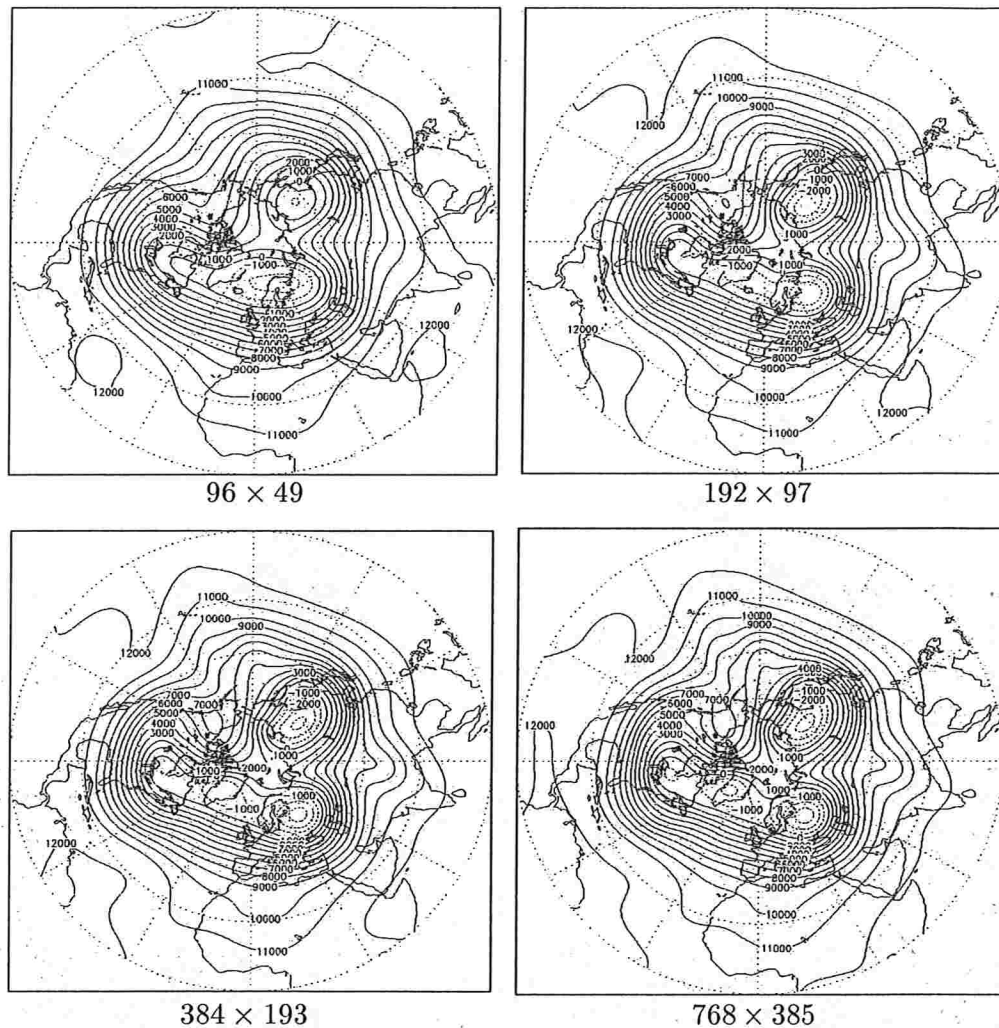


Figura 3.6: Resultados depois de três dias de integração em diferentes malhas uniformes.

Na figura 3.6, mostramos os resultados obtidos (projeção no hemisfério norte) depois de três dias de integração para 4 malhas,  $96 \times 49$  pontos,  $192 \times 97$  pontos,  $384 \times 193$  e  $768 \times 385$ . Usaremos posteriormente esta integração na malha de alta resolução como uma referência para comparações.

Vamos, no entanto, fazer primeiramente um teste de sensibilidade em relação à escolha de  $\Delta t$ , tomando como exemplo a malha de  $192 \times 97$  pontos. Num modelo Euleriano requer-se  $\Delta t$  de cerca de 20 minutos para estabilidade neste caso.

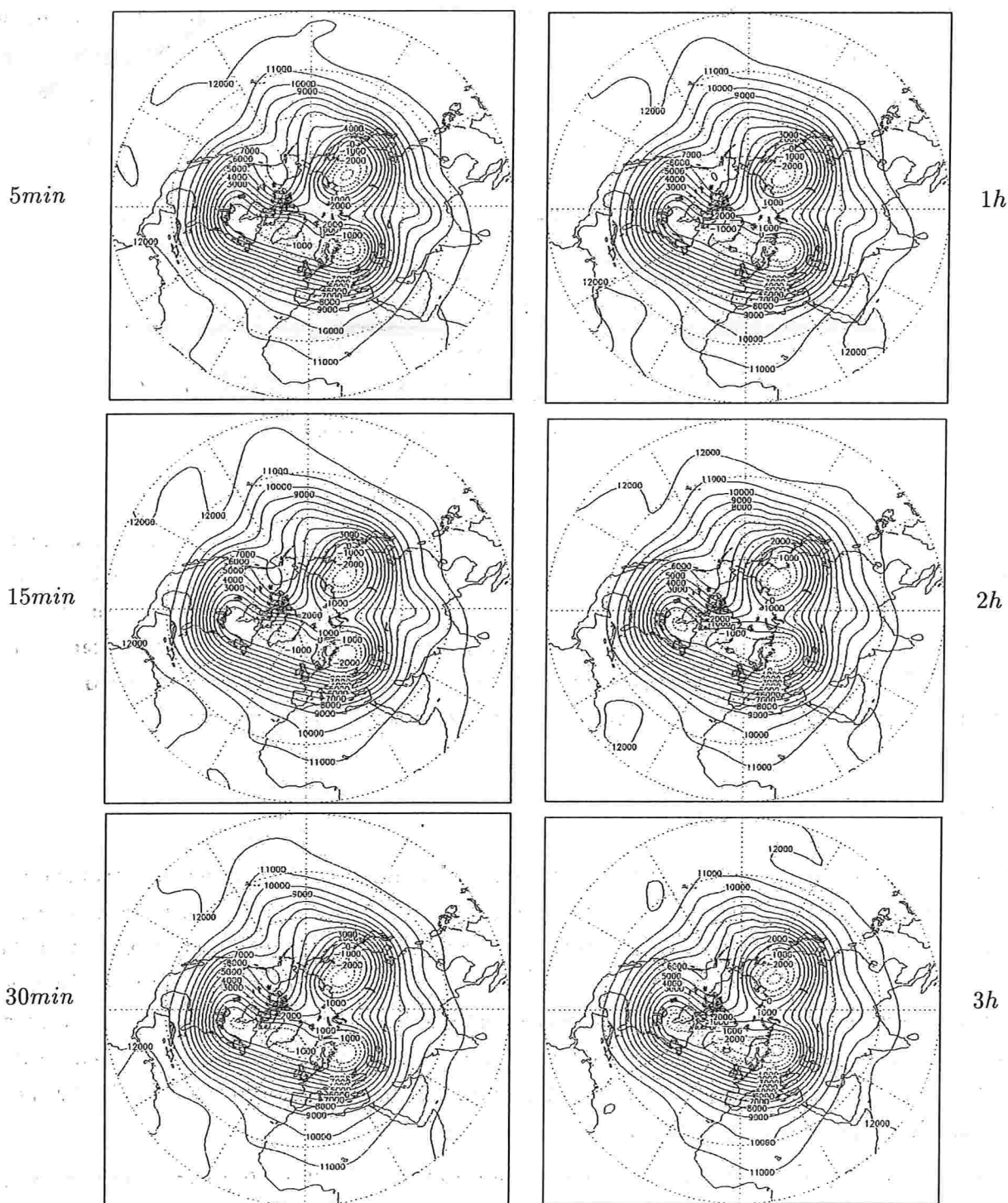


Figura 3.7: Resultados depois de três dias de integração em uma malha uniforme de  $192 \times 97$  para diferentes valores de  $\Delta t$  (indicados ao lado de cada figura).



Integramos o modelo com passos no tempo de  $5min$ ,  $15min$ ,  $30min$ ,  $1h$ ,  $2h$  e  $3h$  para comparação. O geopotencial obtido após 3 dias de integração com os vários passos no tempo pode ser observado na figura 3.7.

A tabela 3.5, traz a norma quadrática do erro (RMS- 'root mean square') entre os resultados obtidos com os diversos passos no tempo comparados com o resultado obtido com  $\Delta t = 5min$ .

$\Delta t$	1 dia	2 dias	3 dias	4 dias	5 dias
15 min	12.1109	12.7829	12.7815	13.5952	13.7632
30 min	16.7280	18.1730	17.3848	17.8822	20.6592
1 h	18.8025	22.4041	22.4923	28.5179	29.9913
2 h	24.0767	28.3503	33.9920	40.0684	43.4756
3 h	27.9935	36.8705	43.6968	48.6463	52.9649

Tabela 3.5: Erros quadráticos da altura

Observemos que o erro é relativamente grande já após um dia de integração com crescimento lento (para  $\Delta t$  até  $1h$ ) do primeiro ao quinto dia. Para passos no tempo de 2 ou 3 horas o crescimento já é bastante grande. Este erro inicial relativamente alto é provavelmente fruto da presença de ondas de gravidade nos dados iniciais, uma vez que não fizemos inicialização dos mesmos. Como as ondas de gravidade não são bem resolvidas ao empregarmos  $\Delta t$  maior, temos este erro logo no primeiro dia. Esta parte no erro não se amplifica no entanto.

### Malha com refinamento local

Passamos agora para o teste do uso de refinamento local no modelo, tomando novamente o caso com dados reais. Queremos verificar até que ponto é viável obter uma integração local de alta precisão em uma dada região coberta por uma malha localmente refinada mesmo tendo baixa resolução na maior parte do globo. É claro que mais cedo ou mais tarde fenômenos mal resolvidos em área não coberta pelo refinamento podem prejudicar a qualidade da solução refinada. Escolhemos como região de interesse uma área sobre a Europa onde a malha será gradativamente refinada. Na figura 3.8 apresentamos o campo geopotencial inicial e destacamos a região de interesse sobre a Europa.

Nos experimentos escolhemos uma malha base uniforme de  $96 \times 49$  pontos. Adicionamos gradativamente diferentes níveis de refinamento local (1, 2 e 3). Como referência tomamos uma malha uniforme de alta resolução de  $768 \times 385$  pontos. Os resultados nesta malha fina servem como parâmetro para os resultados obtidos nas malhas refinadas.

A resolução das malhas refinadas sobre a região de interesse correspondem respectivamente

à resolução de malhas uniformes com  $192 \times 97$ ,  $384 \times 193$  e  $768 \times 385$  pontos, respectivamente para 1, 2 e 3 níveis de refinamento. Em todos os casos empregamos  $\Delta t = 1h$  e um ciclo de Full-Multigrid  $FMG(8, 1, 1, 1)$  para resolver a equação não linear para o geopotencial.

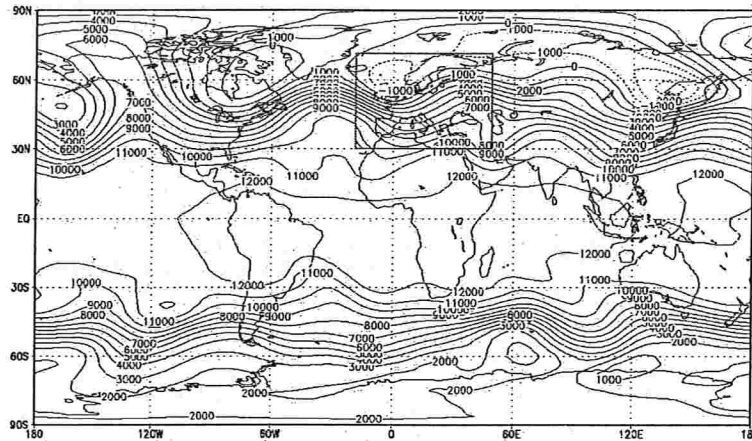
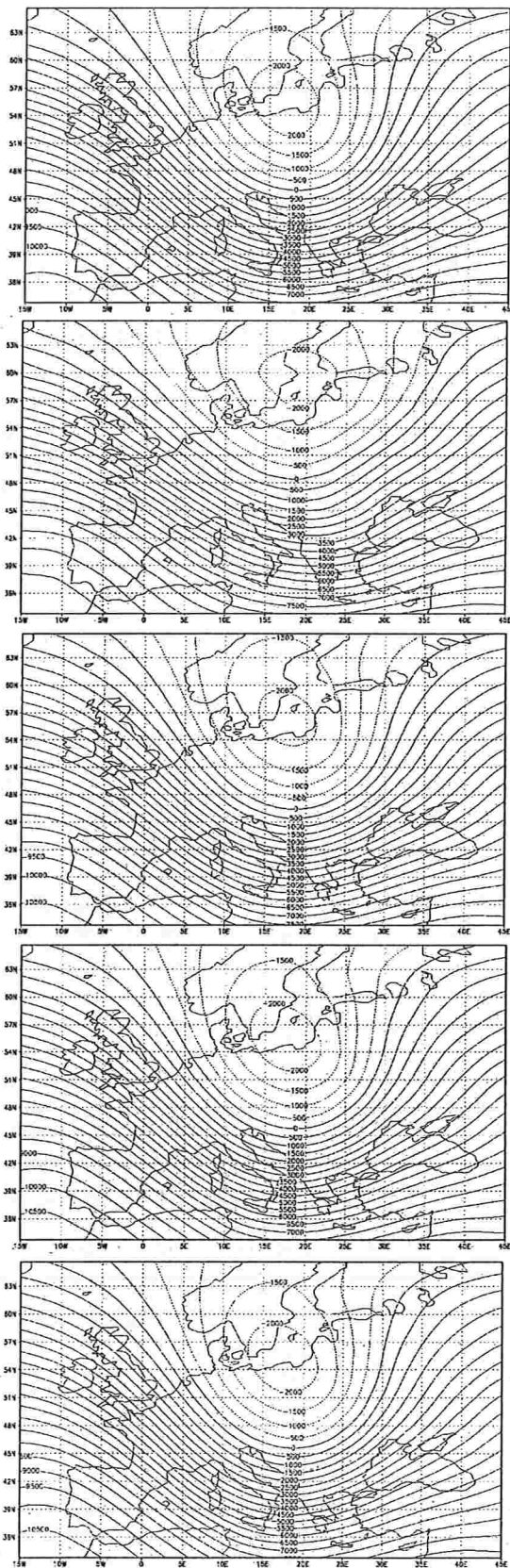


Figura 3.8: Posição da região de interesse.

Nas figuras 3.9, 3.10 e 3.11 podemos observar os resultados obtidos para o geopotencial, na região de interesse com os diversos refinamentos, depois de 1, 2 e 3 dias de integração respectivamente. No topo observamos os resultados obtidos com a malha referência de  $768 \times 385$ . Na segunda linha temos os resultados obtidos usando só a malha base, na malha de  $96 \times 49$ . Na terceira linha empregamos um nível de refinamento na região de interesse, na quarta linha com dois níveis de refinamento e na última entrada temos resultados correspondentes ao uso de três níveis de refinamento. Neste último caso, obtemos na região de interesse a mesma resolução que a usada na malha de referência.

Vemos que com o uso de refinamento local consegue-se capturar a intensidade e a localização do centro de baixa do geopotencial, mesmo mantendo-se a baixa resolução fora da região de interesse. Para um dia de integração com o modelo, com três níveis de refinamento consegue-se uma boa resolução na região de interesse, similar à da malha uniformemente fina. Após dois dias de integração a solução na malha refinada ainda bastante similar à da malha fina, sendo que no terceiro dia a localização do centro é menos precisa, captura-se ainda o padrão geral da solução. Os erros quadráticos médios são apresentados na tabela 3.6.

Na figura 3.12 mostramos os resultados das mesmas integrações através de uma projeção estereográfica no hemisfério norte, depois de 3 dias de integração. Assim podemos observar o padrão da solução tanto na região refinada como fora do refinamento. Notemos que na malha de alta resolução obtemos um centro de alta junto à costa oeste da América do Norte, enquanto que nas outras malhas, onde a resolução nesta área é baixa o centro não foi formado. A boa resolução fica concentrada na região refinada. Além disso, podemos notar que a transição entre os níveis de refinamento é suave, sem o ruído nas interfaces.



Malha uniforme  
768 × 385

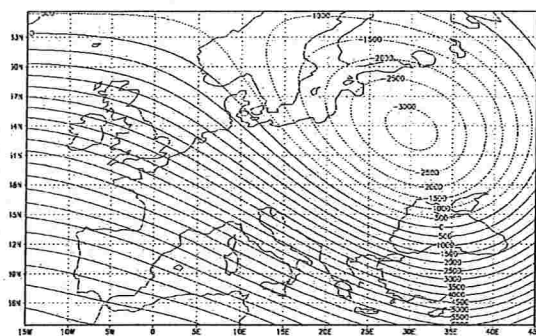
Malha uniforme  
96 × 49

1 nível de  
refinamento

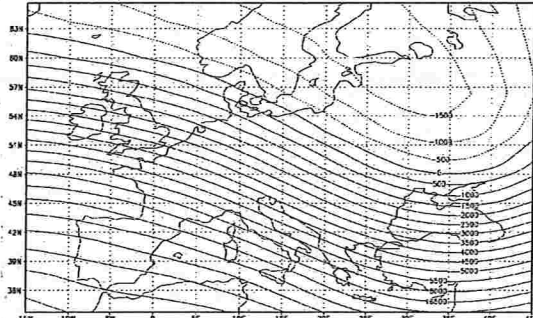
2 níveis de  
refinamento

3 níveis de  
refinamento

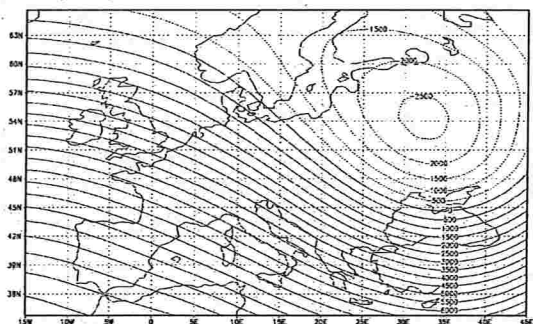
Figura 3.9: Resultados obtidos depois de 1 dia de integração.



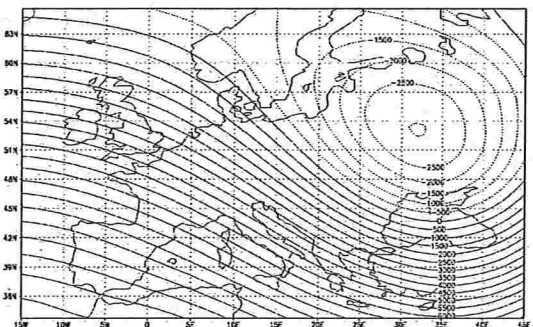
Malha uniforme  
768 × 385



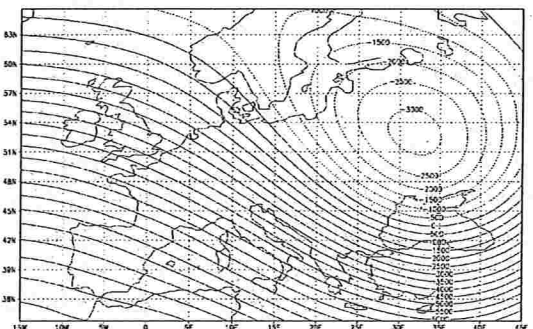
Malha uniforme  
96 × 49



1 nível de  
refinamento



2 níveis de  
refinamento



3 níveis de  
refinamento

Figura 3.10: Resultados obtidos depois de 2 dias de integração.

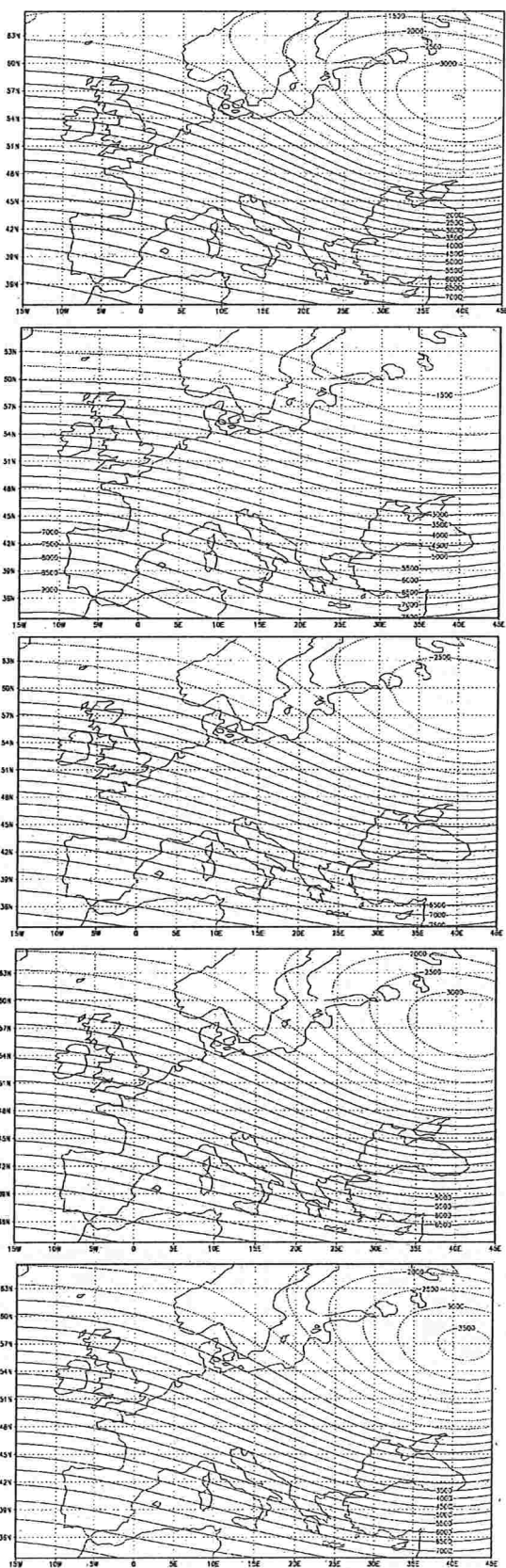
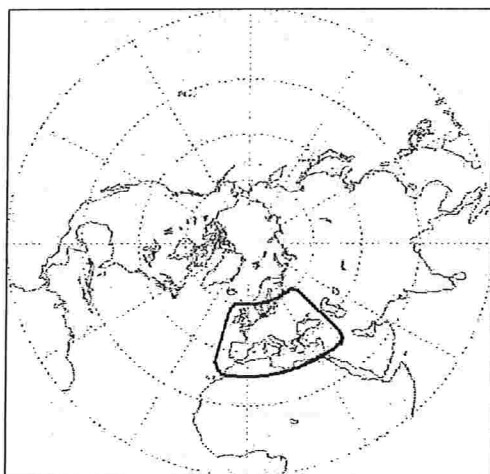
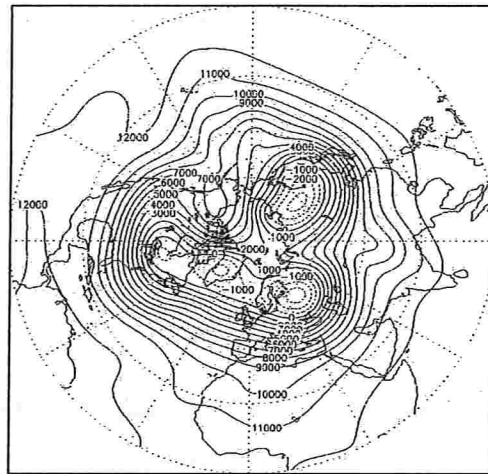
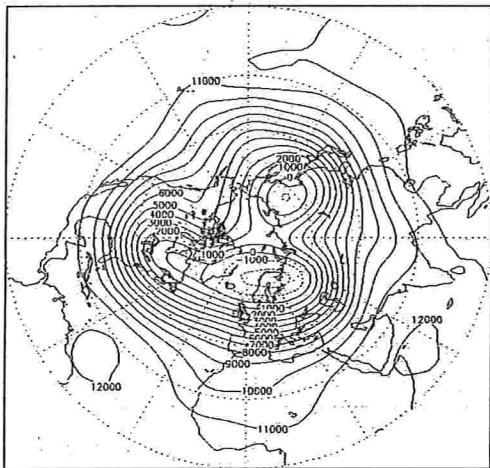
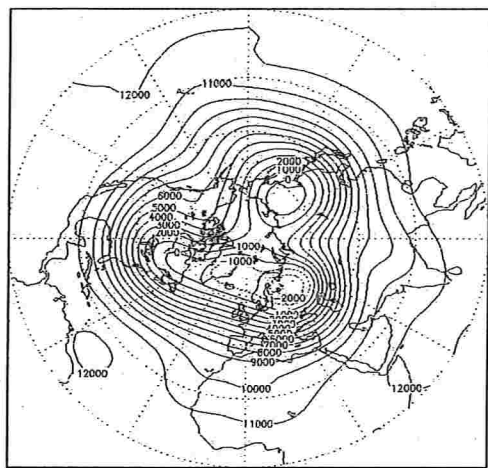


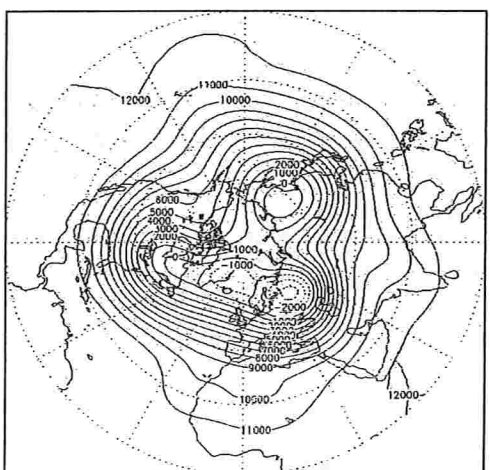
Figura 3.11: Resultados obtidos depois de 3 dias de integração.



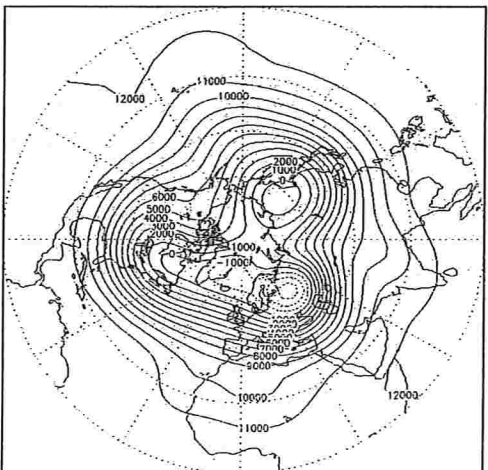
Posição da região de interesse

Malha uniforme,  $768 \times 385$ Malha uniforme,  $96 \times 49$ 

1 nível de refinamento



2 níveis de refinamento



3 níveis de refinamento

Figura 3.12: Resultados obtidos depois de 3 dias de integração.

Tempo	base ( $96 \times 49$ )	um nível	dois níveis	três níveis
1 dia	42.1106	44.9708	20.7307	13.0670
2 dias	77.3241	51.1894	31.3099	23.0003
3 dias	89.3078	56.9356	37.8852	23.2102

Tabela 3.6: Norma quadrática do erro (em metros) nas diferentes malhas mostradas nas figuras 3.9 a 3.11 em comparação com os resultados na malha de alta resolução de  $768 \times 385$  pontos. Em todos os exemplos estamos usando  $\Delta t = 1h$ .

Para julgar a aplicabilidade da técnica proposta temos ainda que levar em conta sua eficiência computacional. Na tabela 3.7, apresentamos tempos de CPU (em segundos) para um dia de integração do modelo com  $\Delta t = 1h$ , com as diversas resoluções utilizadas (estes tempos foram obtidos em uma estação DIGITAL DEC ALPHA 4100 do LCCA, Laboratório de Computação Científica Avançada da USP). Além disso, incluímos na tabela o número de pontos total de grade de cada uma das malhas utilizadas.

Malha	tempo de CPU	num. de pontos
base $96 \times 49$	2.11 seg.	4514
um nível de ref.	2.95 seg.	6146
dois níveis de ref.	4.76 seg.	9770
três níveis de ref.	9.52 seg.	23297
referência $768 \times 385$	144.72 seg.	294146

Tabela 3.7: Tempo de CPU obtidos para as diferentes malhas utilizadas, depois de um dia de integração com  $\Delta t = 1h$  e número de pontos total de grade.

Podemos observar que o custo computacional varia quase que linearmente com o número de pontos das malha. O custo para a previsão local com o refinamento local é mais de uma ordem de magnitude menor que o custo na malha fina uniforme. Além disso, o custo da integração com malha localmente refinada é essencialmente o custo de integração do modelo global na malha base mais o custo envolvido na região refinada. Isto mostra que o modelo também é competitivo computacionalmente com modelos locais (sem ter os problemas deste.)

#### Mais experimentos:

Fizemos outro teste, com o mesmo dado inicial, mudando o local da região de interesse para uma área sobre o noroeste asiático. A posição do refinamento mais fino esta marcada na figura 3.13.

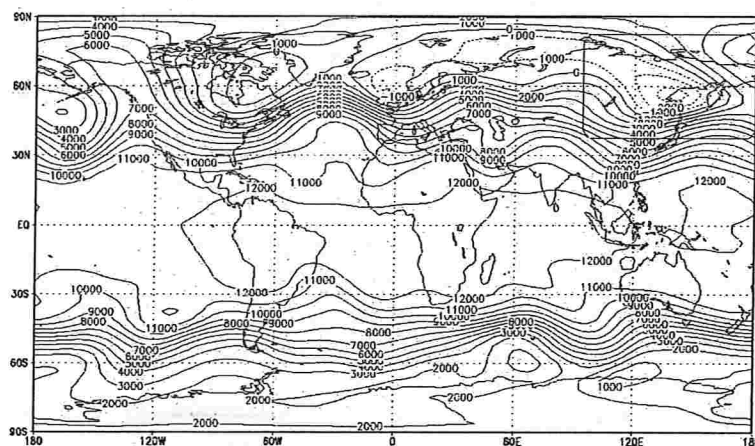
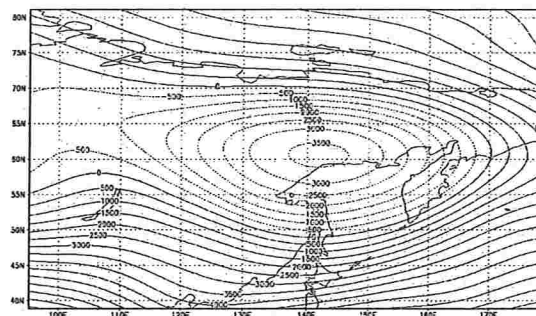


Figura 3.13: Posição do refinamento de maior resolução.

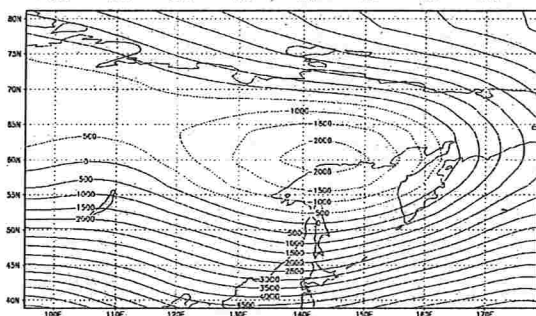
Nas figuras 3.14, 3.15 e 3.16 observamos os resultados obtidos para o geopotencial na região de interesse depois de 1, 2 e 3 dias de integração respectivamente. Usamos a mesma malha fina ( $768 \times 385$ ) para referência e uma malha base de  $96 \times 49$  pontos à qual adicionamos 1, 2 e 3 níveis de refinamentos. De novo temos primeiro o resultado obtido na malha bem fina, depois na malha base, a seguir na malha com um nível de refinamento, com dois níveis e por último com três níveis de refinamentos.

Novamente estamos acompanhando um centro de baixa do geopotencial. Como refinamento de três capturamos bem a localização e intensidade do centro depois de um dia de integração. Nas malhas mais grossas a intensidade é diminuída. Depois de dois dias ainda há boa concordância na solução, com alguma piora no terceiro dia. Qualitativamente o comportamento do algoritmo é similar ao do caso anterior.

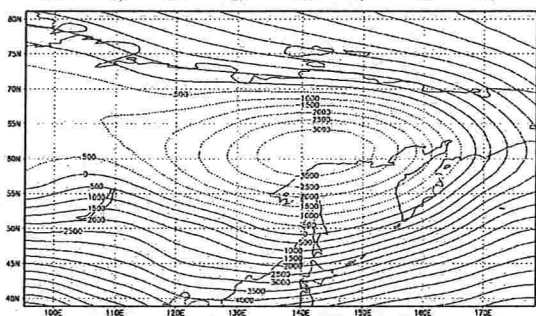




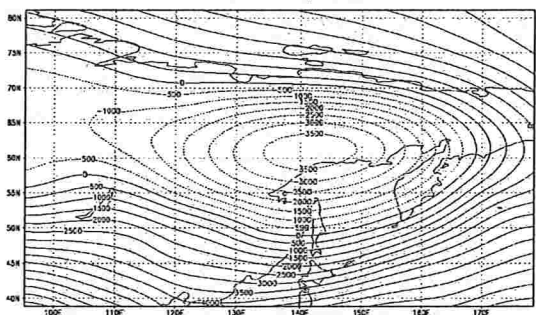
Malha uniforme  
768 × 385



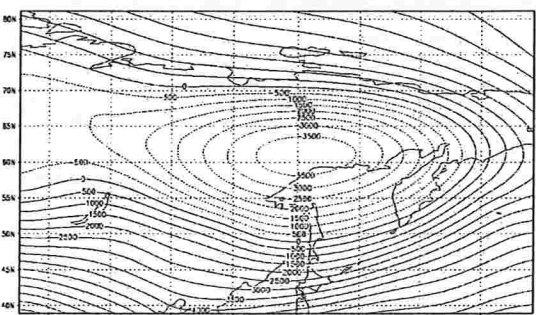
Malha uniforme  
96 × 49



1 nível de  
refinamento



2 níveis de  
refinamento



3 níveis de  
refinamento

Figura 3.14: Resultados obtidos depois de 1 dia de integração.

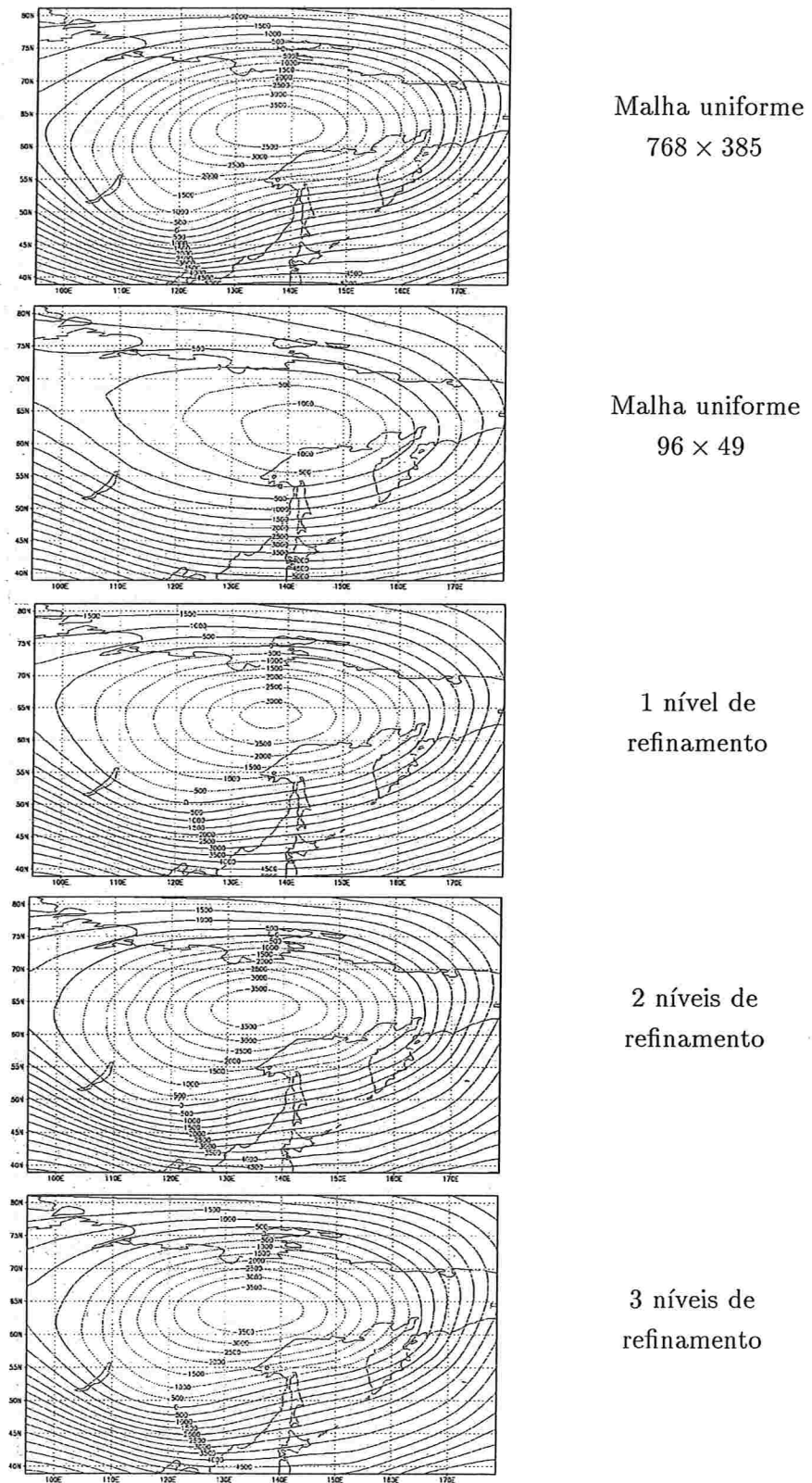
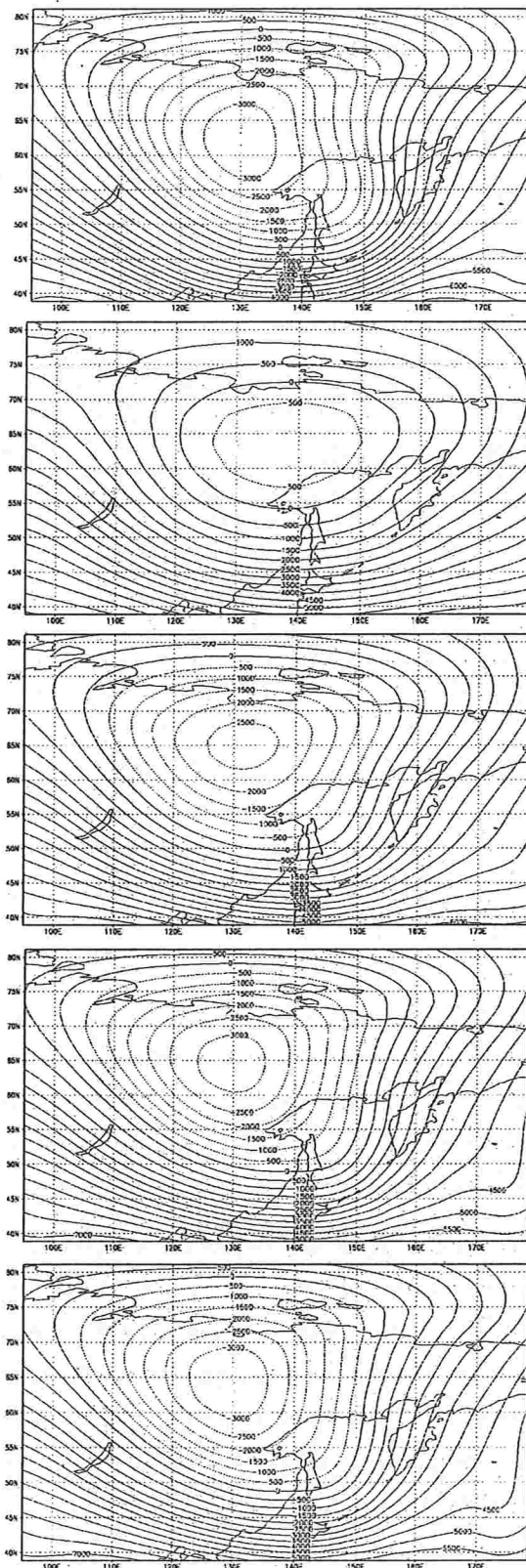


Figura 3.15: Resultados obtidos depois de 2 dias de integração.



Malha uniforme  
768 × 385

Malha uniforme  
96 × 49

1 nível de  
refinamento

2 níveis de  
refinamento

3 níveis de  
refinamento

Figura 3.16: Resultados obtidos depois de 3 dias de integração.

Consideramos agora um caso onde há a formação de um centro de alta na costa oeste da América do Norte conforme podemos observar na figura 3.12, na malha de alta resolução. Este centro não se observa no caso em que se utiliza uma resolução pior nessa região. Vamos então situar o refinamento sobre a região onde o centro se forma, tomando a mesma malha base e malha de alta resolução.

Na figura 3.17, temos a localização do refinamento de maior resolução sobreposto ao geopotencial inicial.

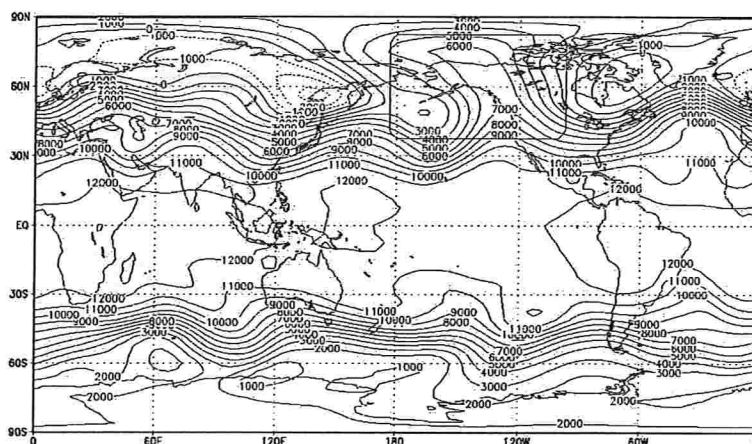


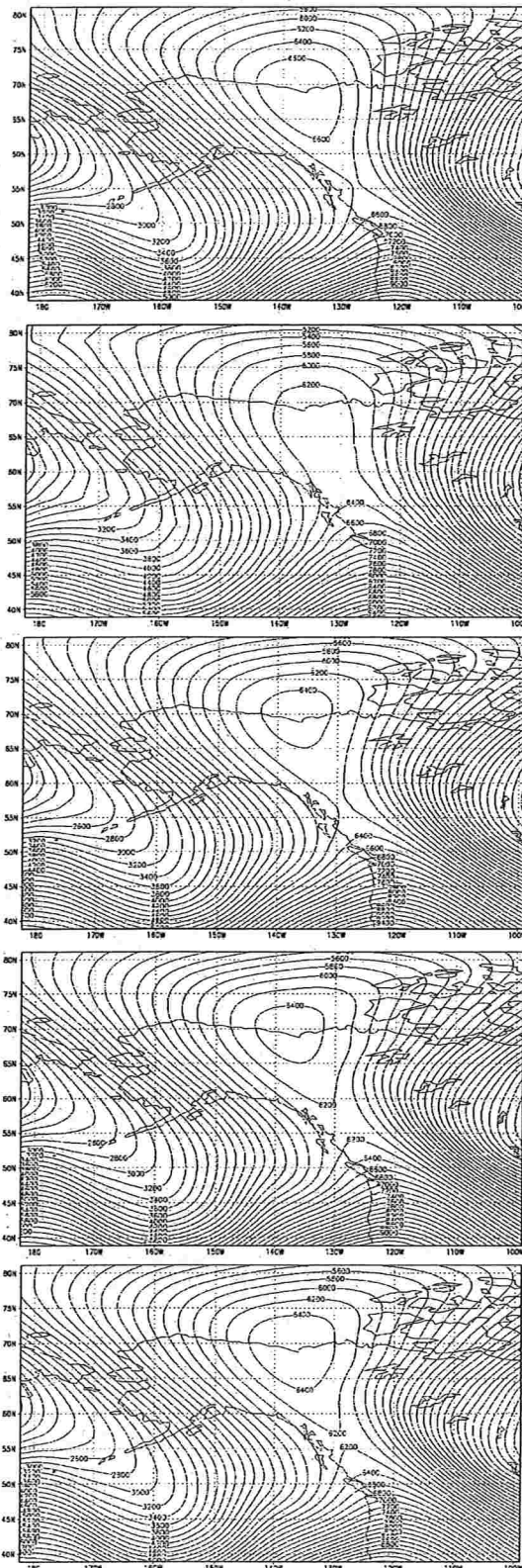
Figura 3.17: Posição do refinamento de maior resolução.

Nas figuras 3.18, 3.19 e 3.20 observamos os resultados obtidos para o geopotencial na região de alta resolução depois de 1, 2 e 3 dias de integração respectivamente.

Já depois de um dia de integração temos a formação do centro de alta, resultado que é obtido com os modelos com refinamento local. A intensidade do centro é bastante atenuada em baixa resolução. Com três níveis de refinamento os resultados são melhores, sem que no entanto consigamos atingir a mesma intensidade obtida com a malha uniforme. O padrão razoavelmente complexo da solução é bem capturado.

Depois de dois dias de integração já há um deslocamento de cerca de 4 graus para leste na posição do centro obtido com a malha refinada em relação à referência. O padrão da solução ainda é razoavelmente bem aproximado. Após 3 dias, os erros em relação à referência já são bem maiores. Quanto menos níveis de refinamento, menor a intensidade do centro, aumentando a diferença em relação a referência. Note-se que na malha base uniforme o centro simplesmente não aparece. Este é um caso em que os efeitos da baixa resolução fora da região de interesse se fazem sentir mais rapidamente no interior. De qualquer forma ainda obtemos resultados razoáveis, mesmos após três dias de integração com a malha localmente refinada. Creemos que esta possa ser bem utilizada em previsões de curto prazo como alternativa em relação a modelos locais de previsão.

Em todos os exemplos de refinamento local nesta seção, os diferentes níveis de refinamento foram colocados de forma a obtermos uma transição suave entre a malha de resolução mais baixa (malha base) e a de melhor resolução.



Malha uniforme  
 $768 \times 385$

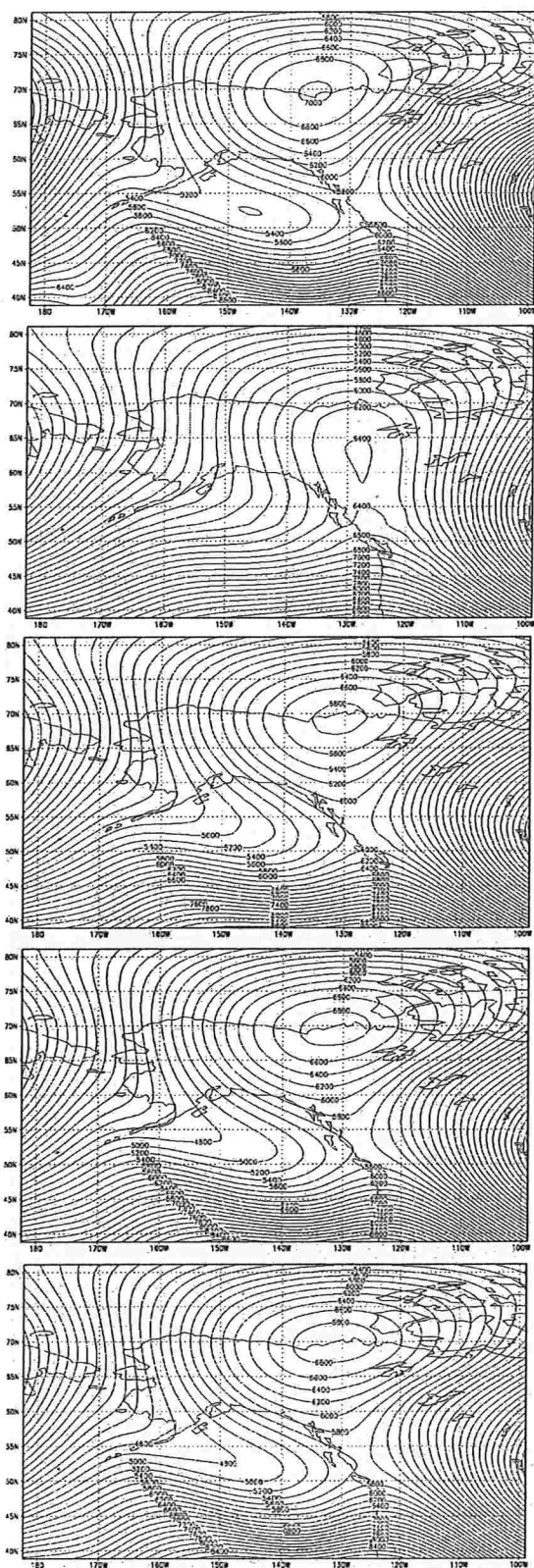
Malha uniforme  
 $96 \times 49$

1 nível de  
refinamento

2 níveis de  
refinamento

3 níveis de  
refinamento

Figura 3.18: Resultados obtidos depois de 1 dia de integração.



Malha uniforme  
768 × 385

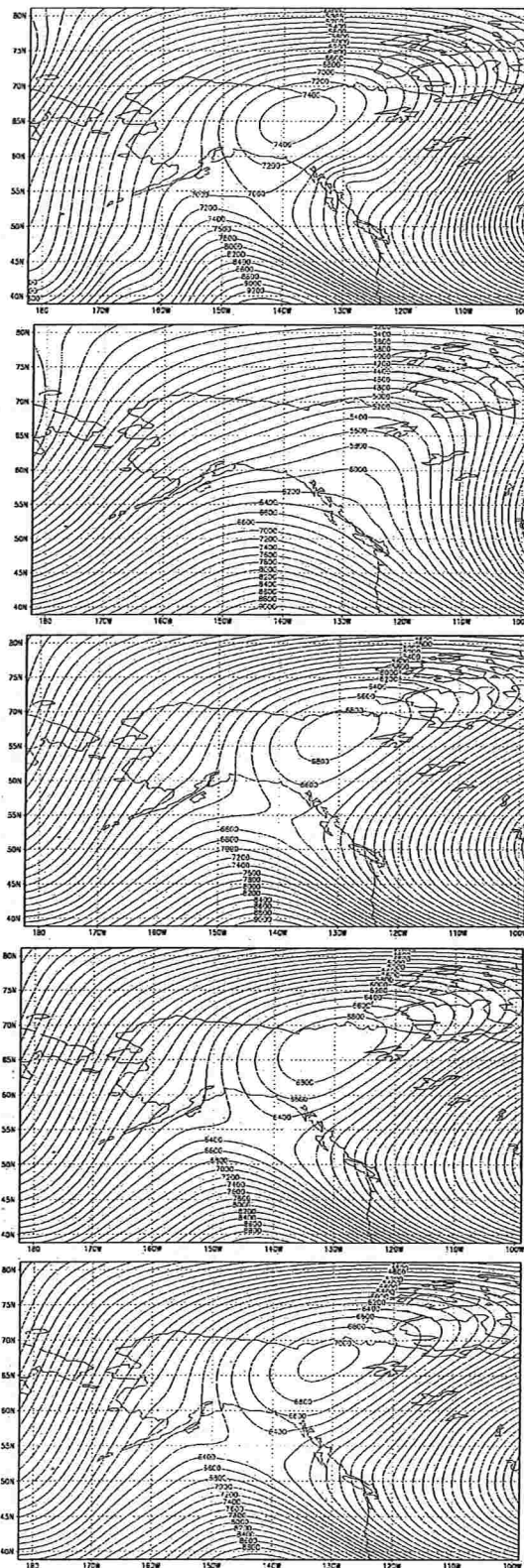
Malha uniforme  
96 × 49

1 nível de  
refinamento

2 níveis de  
refinamento

3 níveis de  
refinamento

Figura 3.19: Resultados obtidos depois de 2 dias de integração.



Malha uniforme  
 $768 \times 385$

Malha uniforme  
 $96 \times 49$

1 nível de  
refinamento

2 níveis de  
refinamento

3 níveis de  
refinamento

Figura 3.20: Resultados obtidos depois de 3 dias de integração.



---

## Refinamento adaptativo

Neste capítulo, discutiremos a possibilidade do uso de refinamento adaptativo nos modelos de previsão do tempo que estudamos nos capítulos anteriores.

Refinamento adaptativo, na sua formulação mais ampla, é um refinamento local cuja posição e tamanho podem variar durante a integração.

Métodos com malhas adaptativas foram utilizados em mecânica dos fluidos computacional entre outros por Berger e Oliger (1984) e Berger e Colella (1989). Estes métodos foram testados em modelos de previsão numérica do tempo regionais com sucesso por Skamarock (1989), Skamarock et al. (1989), Skamarock e Klemp (1993), Dietchmayer e Drogemeier (1992).

Um dos pontos importantes ao se usar refinamento adaptativo, é decidir um critério de adaptatividade. A intenção poderia ser diminuir o erro de truncamento. Neste caso a períodos determinados de tempo, teríamos que calcular uma estimativa do erro de truncamento e o refinamento seria aplicado onde o erro de truncamento fosse maior. Portanto o refinamento adaptativo teria em princípio a função de adicionar resolução onde, com uma malha mais grossa, não se consegue uma boa precisão. Esta abordagem foi investigada em modelos adaptativos para as equações primitivas hidrostáticas por Skamarock et al. (1989).

Outro critério é centrar a malha fina em regiões com variações abruptas na solução (alguma frente), ou onde temos a presença de algum fenômeno que queremos acompanhar com maior detalhe.

Aqui neste capítulo estaremos investigando apenas essa segunda abordagem, tanto para o modelo de vortacidade barotrópica como para o modelo de água-rasa, estudados nos capítulos 2 e 3. Iremos ver, que com a formulação semi-Lagrangeana de nossos modelos com refinamento local a introdução de um refinamento móvel é de simples implementação e pode ser bastante interessante em alguns casos.

## 4.1 Do refinamento fixo ao adaptativo

Para os dois modelos que abordaremos, veremos que do ponto de vista algorítmico não há muita diferença entre usar refinamento fixo ou refinamento adaptativo. Precisaremos, além de calcular a posição do novo refinamento, segundo algum critério de adaptatividade, ser capazes de distinguir dentro do algoritmo quando trabalhamos na malha atual e quando na malha do passo anterior, pois os refinamentos podem ter mudado de posição.

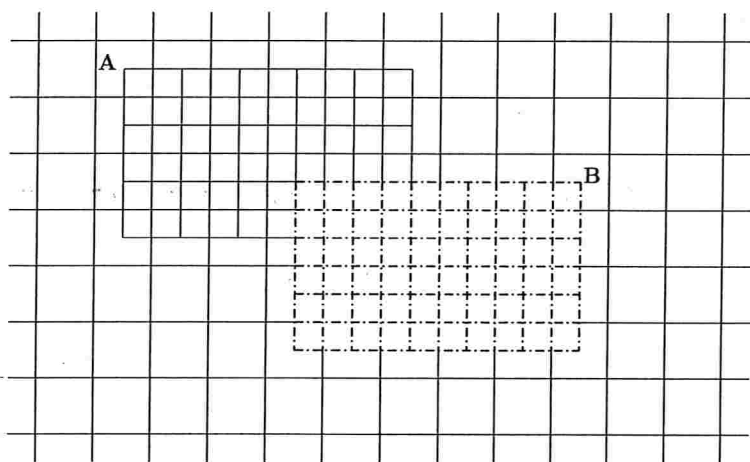


Figura 4.1: *Esquema das malhas.*

A facilidade de adaptação do algoritmo para refinamento fixo ao refinamento adaptativo se deve em grande medida ao fato de estarmos usando o método semi-Lagrangeano. Neste método como visto nos capítulos anteriores, para atualização dos valores das incógnitas num passo no tempo, há a necessidade de interpolar valores do passo anterior. A novidade é que as malhas agora potencialmente diferem de um passo para outro, pois a posição do refinamento pode mudar.

Na figura 4.1 representamos esta situação onde em um passo do tempo o refinamento passa de A para B.

Para efetuarmos a interpolação cúbica dos valores dos pontos de partida, é indiferente se a malha no novo tempo é igual à malha anterior ou não. Apenas os valores na malha anterior serão utilizados, para interpolação referente aos pontos de partida de uma malha nova.

A interpolação nesta malha é calculada exatamente como nos algoritmos anteriores.

O ponto que traz novidade e portanto requer um tratamento diferente é o da extrapolação temporal das velocidades anteriores para o ponto médio da trajetória, uma vez que as duas malhas anteriores podem ser distintas.

Conhecidos os valores das velocidades no tempo  $n$  e  $n - 1$ , queremos extrapolar e assim calcular o valor das velocidades no tempo  $n + \frac{1}{2}$ . Se a malha no tempo  $n - 1$  difere da malha no tempo  $n$ , interpolamos cubicamente os valores da velocidade do tempo  $n - 1$ , para os pontos da malha do tempo  $n$ , (apenas onde as malhas diferem). Assim, ao extrapolar teremos os dois tempos,  $n$  e  $n - 1$ , definidos na mesma malha, sem maiores complicações.

A seguir discutiremos com um pouco mais de detalhe os dois algoritmos.

## 4.2 Algoritmos

### 4.2.1 Vorticidade barotrópica

A equação da vorticidade barotrópica com vento não divergente, (ver capítulo 2) é dada por:

$$\frac{d(\xi + f)}{dt} = 0.$$

No algoritmo, para obtenção da vorticidade no tempo  $n + 1$  a partir do seu valor  $\xi^n$  no tempo  $n$ , temos as etapas:

- i) Determinação da posição da nova malha, segundo o critério de adaptatividade adotado.

Aqui estaremos movendo a posição do refinamento (mantendo fixo o tamanho da área refinada) fazendo o centro da região acompanhar um centro da vorticidade. Outros critérios poderiam ser empregados sem afetar substancialmente o restante do algoritmo.

- ii) Cálculo da função corrente  $\psi^n$  através da resolução da equação

$$\xi^n = \Delta \psi^n$$

com multigrid.

Com o uso de refinamento adaptativo, este passo não muda. Temos  $\xi^n$  conhecida na malha do tempo  $n$ , e calculamos  $\psi^n$  nesta mesma malha, como no modelo para refinamento fixo.

- iii) Conhecida a função corrente  $\psi^n$ , o campo de velocidades  $V^n = (u^n, v^n) = (-\psi_y^n, \psi_x^n)$  é calculado nesta mesma malha da mesma forma que anteriormente.
- iv) Com o novo campo de velocidades, temos que determinar os pontos de partida relativos à nova malha (tempo  $n + 1$ ) que pode diferir da malha do tempo  $n$ .

Neste passo, como comentamos na seção anterior, temos que tomar cuidado na hora de extrapolar os valores de  $V$  no tempo  $n - 1$  e  $n$ , pois podem estar definidos em diferentes malhas.

- v) Por último avaliamos a vorticidade total nos pontos de partida usando interpolação cúbica e assim obtemos a vorticidade total no novo tempo

$$(\xi + f)^{n+1} = (\xi + f)^n.$$

A interpolação é efetuada na malha do tempo  $n + 1$  sem modificações.

#### 4.2.2 Água-rasa

As equações de água-rasa, (ver capítulo 3), são dadas por:

$$\frac{dV}{dt} + f\mathbf{k} \times V + \nabla_H \phi = 0,$$

$$\frac{d \log(\phi)}{dt} + \nabla_H \cdot V = 0,$$

onde  $V = (u, v)$  é o vento,  $\phi$  o geopotencial ( $\phi = gz$ ) e  $f$  o fator de Coriolis.

Para a determinação de  $V$  e  $\phi$  no tempo  $n + 1$ , conhecidos seus valores no tempo  $n$  temos que

- i) Determinar a posição do refinamento no tempo  $n + 1$ .

Aqui estamos movendo a região refinada de forma a seguir um centro de alta ou baixa do geopotencial. Outros critérios poderiam ser usados sem acarretar modificações substanciais no restante do algoritmo.

- ii) Calcular a posição de partida das partículas.

Com o uso de refinamento adaptativo, este cálculo é feito de forma similar à equação da vorticidade. São calculados os pontos de partida da nova malha (tempo  $n + 1$ ), com as devidas adaptações para a extrapolação do vento que, no tempo  $n - 1$  e  $n$ , podemos ter em malhas distintas.

- iii) Calcular os lados direitos das equações.

Calcular os lados direitos das equações na malha, do tempo  $n$  onde as diferentes quantidades estão definidas. Depois interpolamos cubicamente estas quantidades para os pontos de partida da trajetória, sem modificações no algoritmo.

- vi) Cálculo do novo geopotencial.

Isto requer uma manipulação do sistema de equações na malha nova (no tempo  $n + 1$ ) para sua resolução a uma equação escalar para o geopotencial. Esta etapa e a posterior solução da equação é totalmente realizada na malha no instante  $n + 1$ . Como esta tem mesma

estrutura que a malha do instante  $n$ , a solução procede sem modificações em relação ao algoritmo anterior. (Apenas tem que ser passadas as informações da malha nova as rotinas que fazem este cálculo).

- v) Atualização do campo de velocidades usando as equações de momento.

Uma vez computado o valor para  $\phi^{n+1}$ , o campo de velocidades  $V$  é atualizado trivialmente a partir das equações de momento na malha no tempo  $n + 1$ .

Novamente vê-se que uma vez escolhido o critério de adaptatividade para a determinação das malhas, o algoritmo praticamente não requer modificações. Apenas a extrapolação do vento necessita ser adaptada.

## 4.3 Resultados

### 4.3.1 Vorticidade barotrópica

Nesta seção apresentamos alguns resultados obtidos com uso de refinamento adaptativo e os comparamos com resultados, usando refinamento fixo, do capítulo 2.

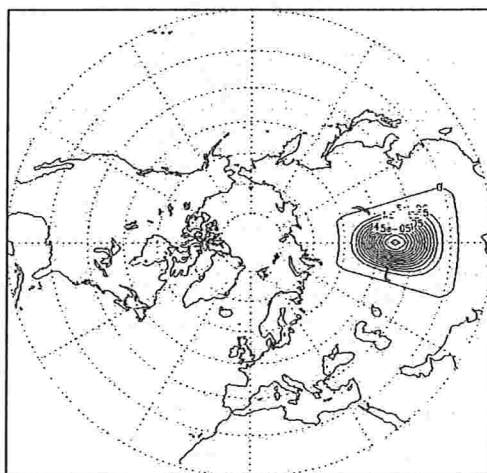


Figura 4.2: Projeção estereográfica no polo norte do dado inicial.

Para isso tomaremos a vorticidade inicial da figura 4.2, uma função que toma o valor zero em toda a esfera, a menos de uma gaussiana em  $\lambda = 90^\circ$  e  $\theta = 60^\circ$ .

Colocamos uma região refinada, englobando a 'Gaussiana' com apenas um nível de refinamento.

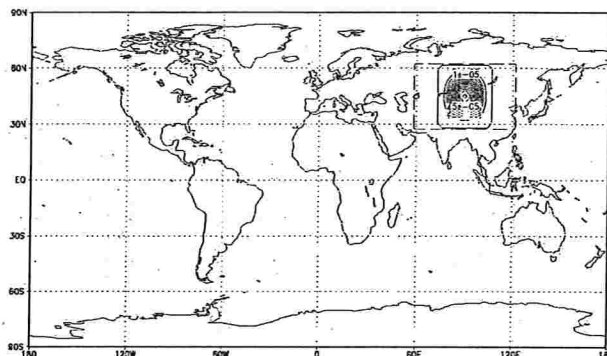


Figura 4.3: Dado inicial em todo o globo e o posição do refinamento local.

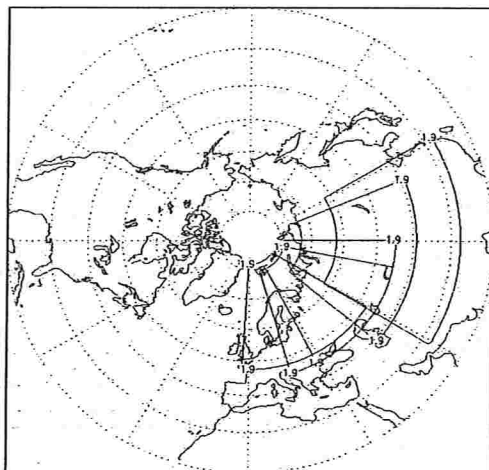


Figura 4.4: Diferentes posições do refinamento adaptativo.

Na figura 4.4 apresentamos algumas das posições que o refinamento local toma ao longo da integração.

Na figura 4.5 na coluna (a) observamos os resultados da vorticidade aproximada usando o refinamento local fixo, depois de 4, 7 e 10 dias de integração. Nas figuras apresentamos também a posição do refinamento local. Utilizamos uma malha base uniforme de  $128 \times 65$  pontos e dobramos a resolução na região refinada.

Observamos que a 'Gaussiana', inicialmente dentro do refinamento, aos 4 dias de integração está praticamente fora da região refinada, na região com resolução menor. A idéia do refinamento adaptativo é que este siga o centro da 'Gaussiana' para assim, manter a mesma resolução em toda a integração.

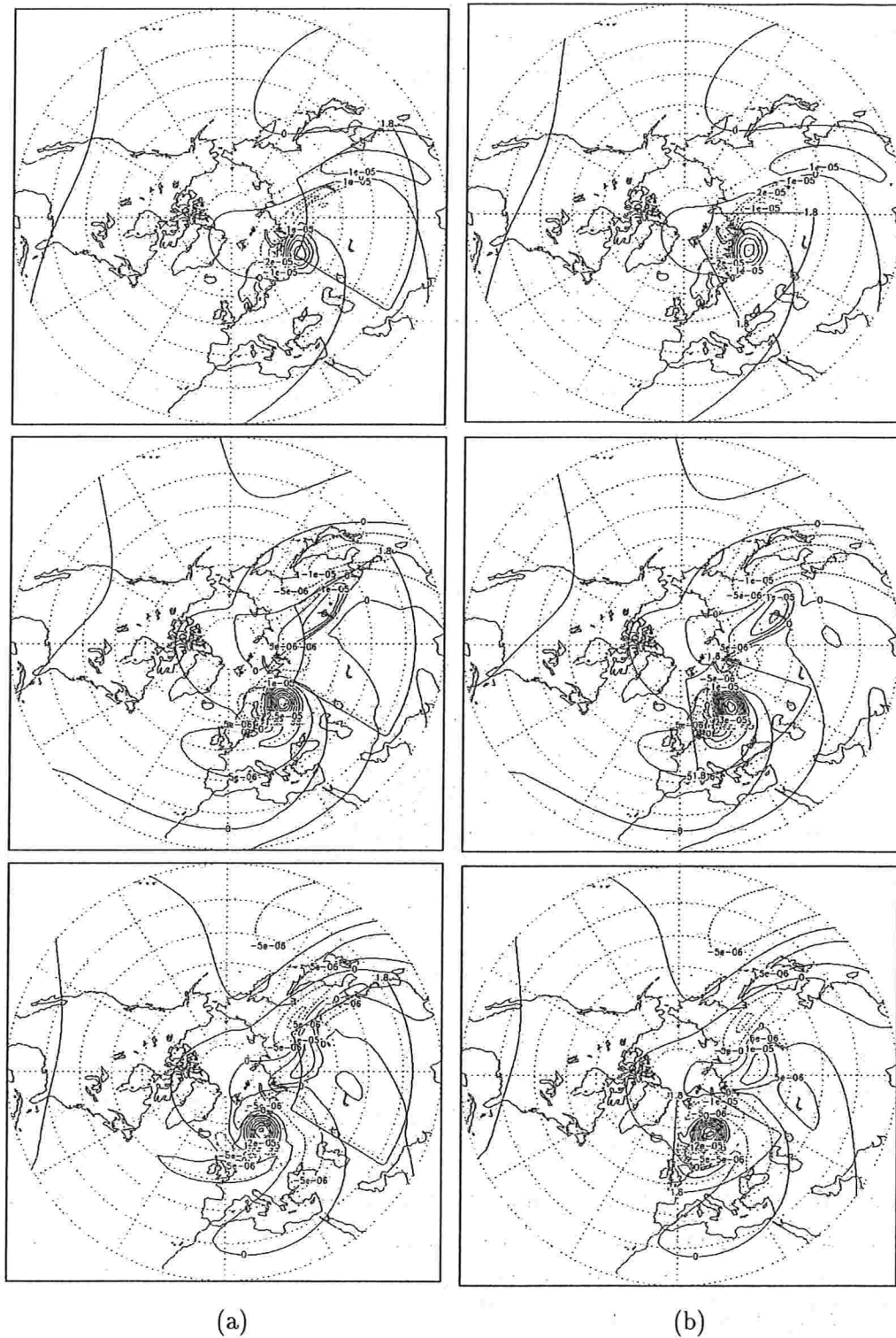


Figura 4.5: Resultados depois de 4, 7 e 10 dias de integração, na coluna (a) usando uma malha com refinamento fixo e na coluna (b) usando refinamento adaptativo. A posição do refinamento também é mostrada. Note como ela segue a 'Gaussiana' com o refinamento adaptativo.

Na figura 4.5 na coluna (b) observamos os resultados da vorticidade aproximada usando o refinamento adaptativo. Inicialmente tomamos o mesmo refinamento que no exemplo com refinamento fixo, a mesma quantidade de pontos e na mesma posição. No topo da coluna temos o resultado depois de 4 dias de integração, a seguir, depois de 7 dias e no final depois de 10 dias de integração. Em cada uma das figuras apresentamos também a posição do refinamento adaptativo e observamos como este acompanha o movimento da 'Gaussiana' durante toda a integração.

Para maior detalhe, rodamos o algoritmo com uma malha de alta resolução de  $1024 \times 513$  pontos (a qual chamaremos de vorticidade referência), e comparamos os resultados, fazendo um corte transversal, na altura máxima da vorticidade referência na latitude =  $64^\circ$ .

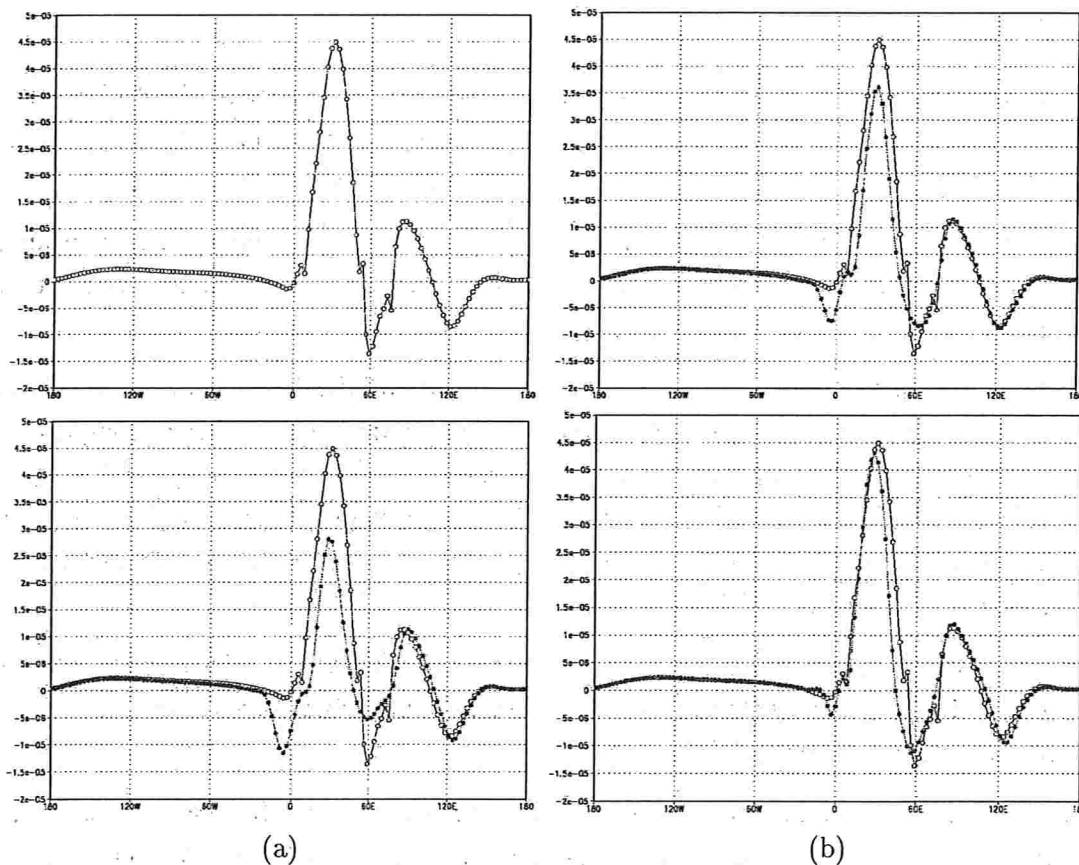


Figura 4.6: Corte transversal na latitude =  $64^\circ$  depois de 10 dias de integração. Na coluna (a) a vorticidade referência e a seguir a referência e a vorticidade usando a malha base de  $128 \times 65$  pontos. Na coluna (b) a vorticidade referência e a obtida com a malha com refinamento fixo e por ultimo a referência e a vorticidade usando refinamento adaptativo.



Na figura 4.6 temos no topo da coluna (a), o corte transversal da vorticidade referência aos 10 dias de integração, na latitude =  $64^\circ$ , a seguir temos o corte transversal da solução de referência e a vorticidade aproximada obtida com uma malha base (uniforme) de  $128 \times 97$  pontos. No topo da coluna (b), podemos observar a vorticidade referência e a obtida usando a malha com refinamento fixo, e por ultimo usando refinamento adaptativo. Em todos os casos estamos usando  $\Delta t = 1h$ .

Podemos observar que, mesmo no caso de refinamento fixo, onde o máximo da vorticidade permanece pouco tempo dentro do refinamento, o uso do refinamento contribuiu para melhora na precisão da solução. No caso do refinamento adaptativo, com pequeno custo adicional, consegue-se uma precisão bem melhor, quase tão boa quanto a obtida com a malha uniformemente fina.

### 4.3.2 Água-rasa

Tomamos como dado inicial, o dado real usado nos exemplos do capítulo 3, (na figura 4.8, podemos observar o geopotencial inicial). Iremos empregar o refinamento adaptativo, para acompanhar o centro de baixa pressão inicialmente localizado ao norte da Europa.

Para efeito de comparação do refinamento adaptativo com o refinamento fixo, utilizaremos três refinamentos diferentes, todos com três níveis de refinamento, os quais são mostrados na figura 4.8.

Primeiro, definimos um refinamento, na baixa do dado inicial e rodamos nosso algoritmo adaptativo por 3 dias de integração. Escolhemos três refinamentos fixos para comparação com os resultado usando o refinamento adaptativo. O primeiro, mostrado na figura 4.8-(a), é o refinamento inicial usado no exemplo adaptativo. Como segundo refinamento fixo, escolhemos o refinamento final, obtido aos três dias de integração com o refinamento adaptativo, cuja localização é mostrada na figura 4.8-(b).

Por último tomamos um terceiro refinamento fixo englobando uma área, contendo todas as diferentes posições de refinamento ocupadas pelo refinamento adaptativo durante os três dias de integração (conforme a figura 4.8-(c)).

Comparamos os resultados para o geopotencial obtidos com as diversas malhas, com uma solução de referência numa malha uniforme de alta resolução, com  $768 \times 385$  pontos. Em todos os exemplos estamos usando  $\Delta t = 1h$ .

Todas as malhas com refinamento, fixo ou adaptativo, empregam uma malha base de  $96 \times 49$  pontos com três níveis sucessivos de refinamento por um fator 2 (veja figura 4.8). A resolução na região mais fina corresponde à resolução da malha uniforme de referência.

Na figura 4.7 observamos algumas das posições ocupadas pela região refinada no nível mais

fino ao longo da integração.

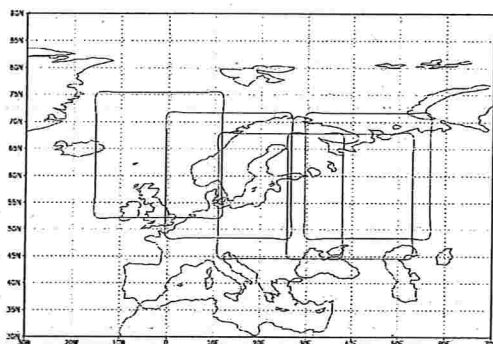


Figura 4.7: Diferentes posições do refinamento adaptativo, ao longo da integração.

Na figura 4.9 podemos observar o geopotencial usando as diferentes malhas, depois de um dia de integração. No topo, temos o resultado com malha com refinamento adaptativo, na coluna (a) observamos primeiro o resultado usando o refinamento fixo, como mostrado no topo da figura 4.8 e a seguir o resultado usando o refinamento fixo 4.8-(b), o centro de baixa está inicialmente fora da região refinada para entrar nela no final da integração.

Na figura 4.9, na coluna (b), temos o geopotencial usando o refinamento fixo maior, mostrado na figura 4.8-(c) e por último o resultado usando a malha uniforme de alta resolução de  $768 \times 385$  pontos.

Na figura 4.10 podemos observar os mesmos gráficos, mas depois de 2 dias de integração e na figura 4.11 aos três dias.

Primeiramente observamos que a solução obtida com os refinamentos menores (coluna (a) das figuras) já após um dia de integração perde muito da precisão. Se o refinamento está à direita, e portanto o centro de baixa do geopotencial está inicialmente numa região de baixa resolução, sua intensidade é logo perdida. Com o refinamento inicial englobando o centro de baixa, a intensidade deste mantém-se mais próxima da solução de referência, mas sua localização é mal resolvida. Após os três dias de integração a qualidade da solução nestas duas malhas é bastante insatisfatória. Ao tomarmos o refinamento fixo em uma região maior, cobrindo a área em que o centro se movimenta obtemos uma solução com boa localização do centro e intensidade aproximadamente correta após um dia de integração. Após dois dias de integração a aproximação ainda é bastante boa, perdendo um pouco de precisão no terceiro dia. Com o uso do refinamento móvel conseguimos a mesma qualidade de solução após um dia e qualidade comparável após dois dias (a intensidade é melhor resolvida mas o posicionamento do centro é ligeiramente pior) e três dias de integração. A vantagem do refinamento móvel neste caso é que precisamos refinar uma

área bem menor para obter uma boa aproximação. Como os custos extras associados ao uso do método adaptativo neste caso são pequenos, obtemos uma boa vantagem computacional, como podemos observar comparando os tempos de CPU na tabela 4.1.

refinamento	tempo de CPU
da figura 4.8-(a)	4.91 seg.
da figura 4.8-(b)	4.86 seg.
da figura 4.8-(c)	8.95 seg.
adaptativo	4.98 seg.
malha uniforme de $768 \times 385$ pontos	144.72 seg.

Tabela 4.1: Tempos de CPU obtidos para os diferentes refinamentos, depois de um dia de integração com  $\Delta t = 1h$ .

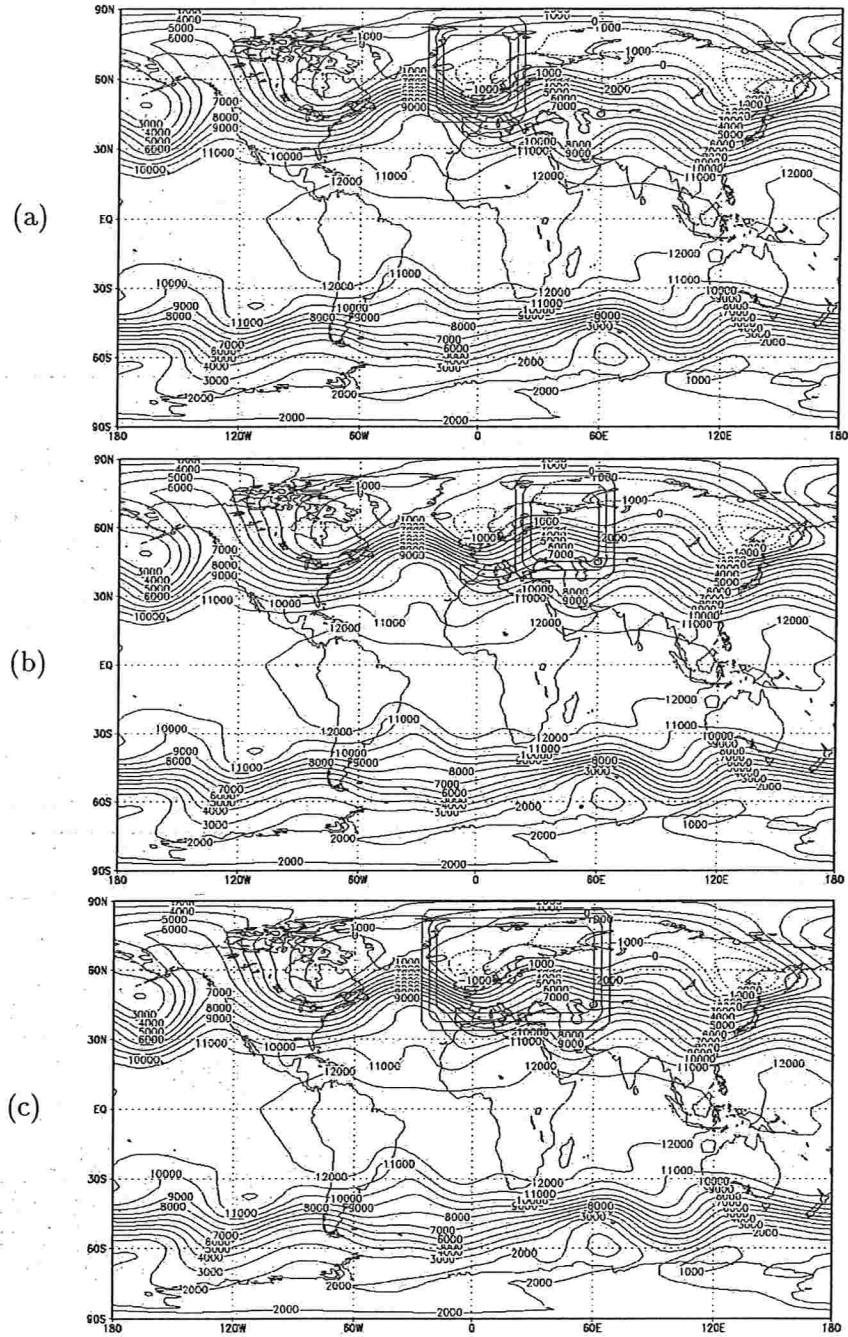


Figura 4.8: Diferentes posições dos refinamentos fixos usados nos testes. O refinamento adaptativo foi inicialmente situado na posição da figura (a). A posição da figura (b) corresponde à localização final do refinamento móvel.

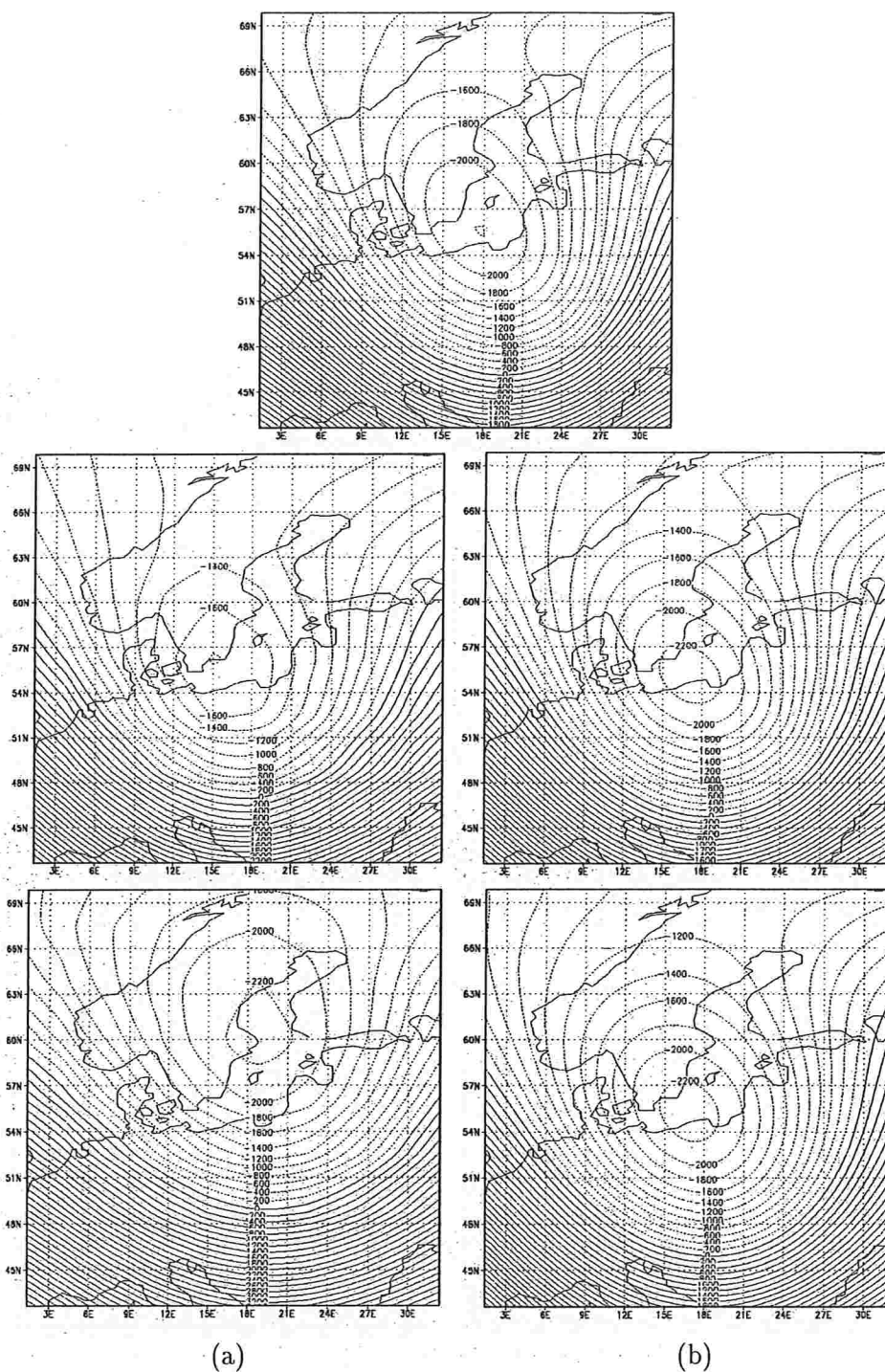


Figura 4.9: Resultados obtidos depois de 1 dia de integração com três níveis de refinamento. No topo o resultado com refinamento adaptativo, na coluna (a) acima temos o resultado usando o refinamento fixo a direita (conforme figura 4.8), abaixo o refinamento fixo a esquerda. Na coluna (b) temos acima o resultado usando o refinamento fixo maior e abaixo a malha de alta resolução.

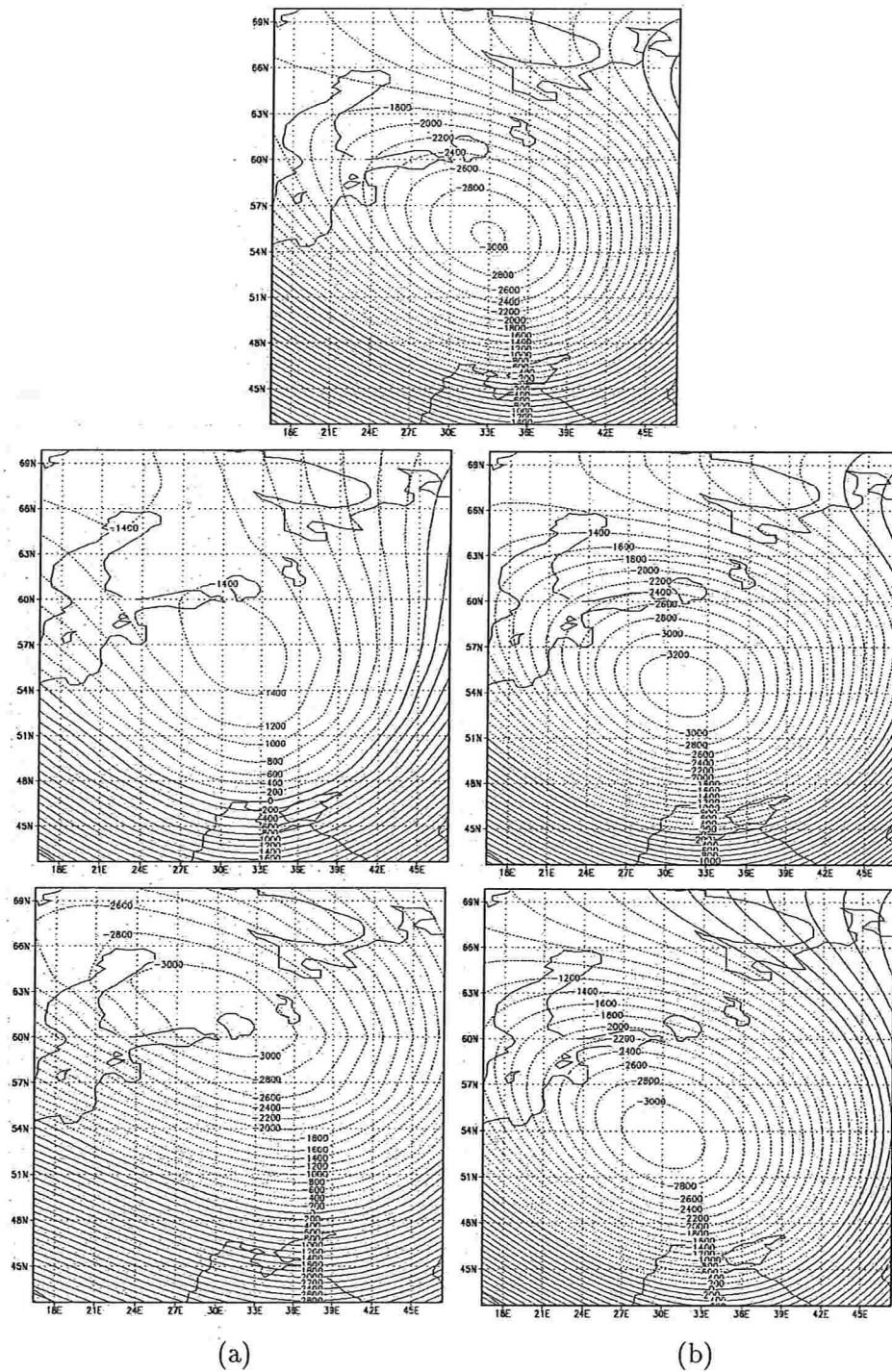


Figura 4.10: Resultados obtidos depois de 2 dias de integração com três níveis de refinamento. No topo o resultado com refinamento adaptativo, na coluna (a) acima temos o resultado usando o refinamento fixo a direita (conforme figura 4.8), abaixo o refinamento fixo a esquerda. Na coluna (b) temos acima o resultado usando o refinamento fixo maior e abaixo a malha de alta resolução.

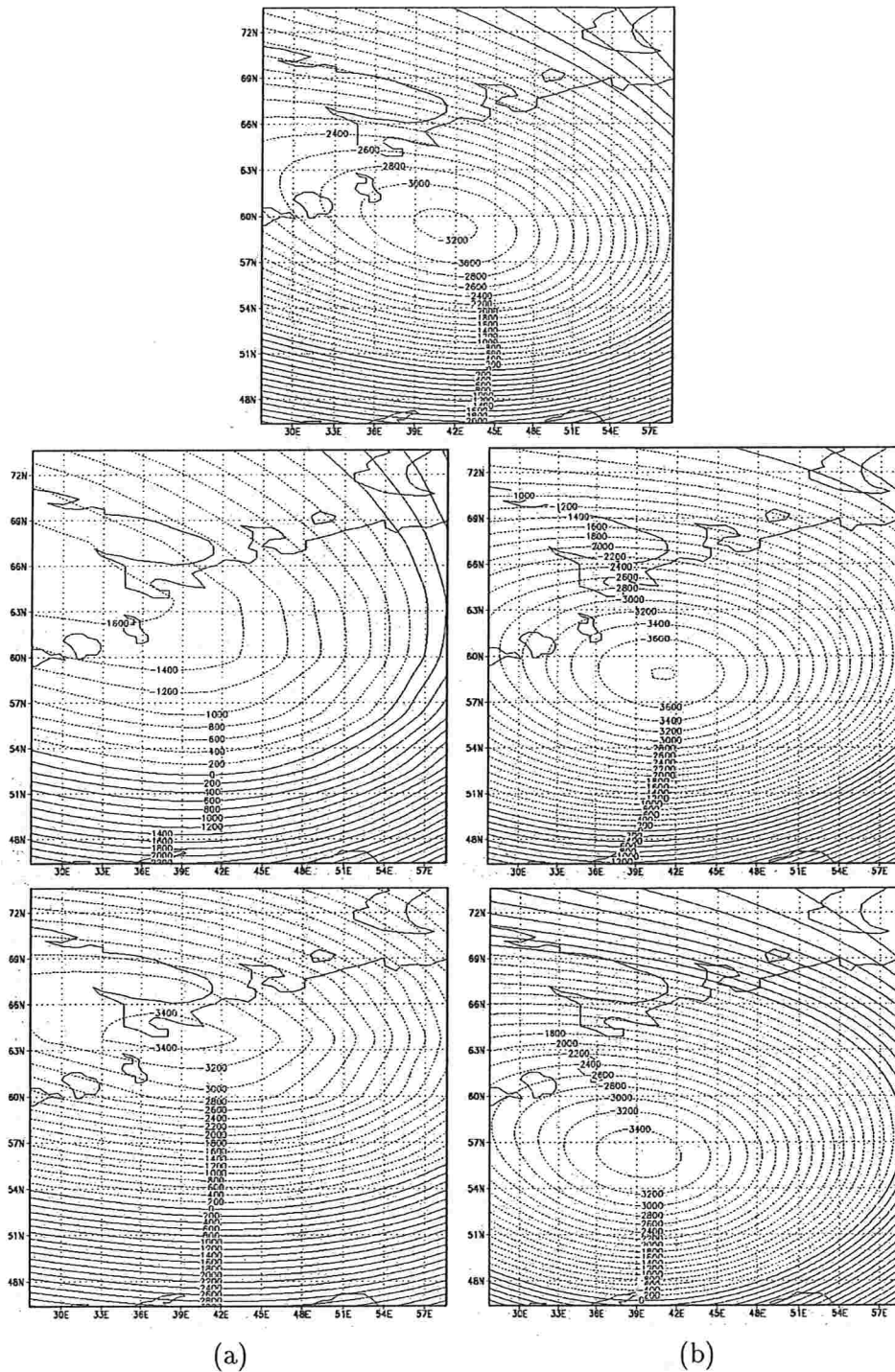


Figura 4.11: Resultados obtidos depois de 3 dias de integração com três níveis de refinamento. No topo o resultado com refinamento adaptativo, na coluna (a) acima temos o resultado usando o refinamento fixo a direita (conforme figura 4.8), abaixo o refinamento fixo a esquerda. Na coluna (b) temos acima o resultado usando o refinamento fixo maior e abaixo a malha de alta resolução.





## Conclusões

Neste trabalho demonstramos a viabilidade do uso de técnicas de refinamento local em modelos globais de previsão do tempo, como uma forma de obtenção de alta resolução em uma região de interesse, de maneira computacionalmente eficiente. Desta forma evita-se problemas com condições de fronteira artificiais, comuns em modelos locais de previsão, e pode-se em princípio utilizar um mesmo modelo para previsões globais e locais. Vemos várias vantagens no uso de refinamento local em comparação a malhas com resolução variável obtidas por transformação ('stretching') de coordenadas, uma vez que nossas malhas são localmente sempre uniformes. O uso de 'stretching' leva a faixas com alta resolução em uma das coordenadas e baixa resolução na outra, trazendo padrões anisotrópicos variáveis ao longo do globo. Isto torna, por exemplo, a resolução de problemas elípticos associados aos métodos semi-implícitos muito mais custosa.

Nosso trabalho foi baseado em uma seqüência de modelos, terminando em um modelo de água-rasa. O modelo semi-implícito semi-Lagrangeano de dois níveis no tempo que desenvolvemos tem propriedades de estabilidade muito boas, não sofrendo restrições do tipo CFL. Demos um tratamento totalmente implícito ao divergente de massa, incluindo sua parte não linear. Por estarmos usando um método multigrid para solução da equação elíptica resultante, pudemos naturalmente adaptá-lo ao caso não linear sem perda de eficiência computacional. O uso de multigrid também foi fundamental para uma implementação eficiente dos modelos com refinamento local. Os métodos diretos rápidos utilizados na resolução de equações elípticas (baseados em FFT ou redução cíclica) não se adaptariam facilmente a um tal problema. O uso do método semi-Lagrangeano, além de fundamental para a boa estabilidade do esquema, também é bastante propício ao uso de refinamento local (temos essencialmente que lidar bem com interpolações em malhas localmente refinadas), tornando relativamente fácil sua implementação. Além disso, o método semi-Lagrangeano também viabilizou o uso de métodos adaptativos com malhas móveis, uma possibilidade que se mostrou interessante para o acompanhamento de fenômenos localizados.

Uma primeira extensão deste trabalho seria a investigação de outros critérios de adaptativi-

dade, permitindo a variação do tamanho da região refinada, que não chegamos a investigar.

Um projeto mais ambicioso seria a extensão do trabalho a um modelo tridimensional baseado nas equações primitivas. Esta tarefa passa pelo desenvolvimento de um modelo global (de diferenças finitas) semi-Lagrangeano para as equações, preferencialmente utilizando multigrid, e envolve sua adaptação ao uso de refinamento local. Desta forma, acreditamos que este trabalho não se encerra aqui, mas que mostra um caminho viável para previsões locais com modelos globais de previsão numérica do tempo.

# Advecção unidimensional

Passaremos para o caso unidimensional, para observar melhor os efeitos de trabalhar numa malha com refinamento local e os efeitos produzidos pela interpolação cúbica (como a presença de um pequeno 'under and overshooting'). Estudaremos algumas soluções alternativas aos problemas observados no capítulo 1, os quais não são muito relevantes, mas que podem ser indesejáveis para quantidades positivas, como umidade.

Primeiro, com um teste simples, queremos descobrir se os efeitos causados pela interpolação cúbica no caso bi-dimensional, são observados em uma só dimensão.

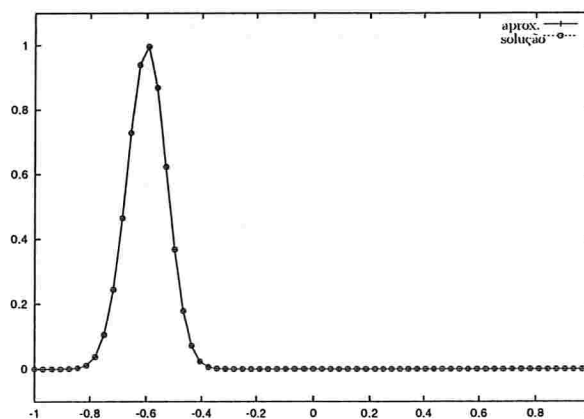


Figura A.1: Dado inicial.

Nesta seção estudaremos a advecção de uma 'Gaussiana' no intervalo  $[-1, 1]$  com condições de fronteira periódica, tal que depois de 250 iterações ela dá uma volta completa. Inicialmente a 'Gaussiana' estará centrada no ponto  $-0.6$  (ver figura A.1). A malha com a qual vamos trabalhar terá 64 pontos e os refinamentos serão de um fator dois. Consideraremos inicialmente a 'Gaussiana' de altura máxima 1 unidade.

A trajetória neste modelo é calculada exatamente, sendo o erro exclusivamente causado pela interpolação. Nestes primeiros exemplos usaremos um polinômio interpolador cúbico de Lagrange.

Primeiro veremos o comportamento na malha uniforme. A figura A.2 mostra a solução aproximada e a analítica na malha uniforme (sem refinamento local). Na figura (a) temos as duas funções depois de 125 iterações e em (b) depois de 250. Observamos basicamente os mesmos efeitos que no caso bi-dimensional, um rebaixamento na altura máxima da 'Gaussiana' e a aparição de valores negativos perto da base da 'Gaussiana'.

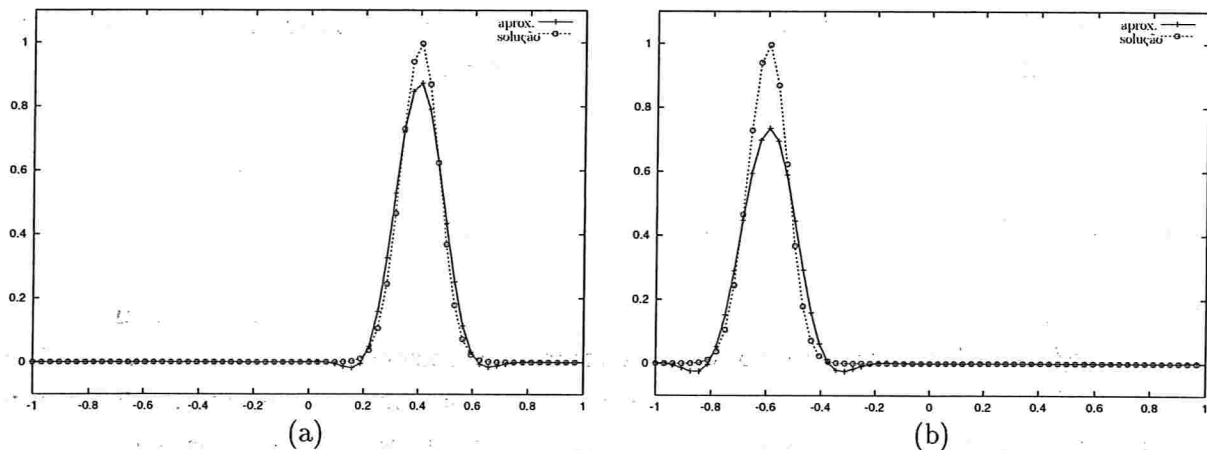


Figura A.2: Solução analítica e função aproximada numa malha uniforme utilizando um interpolador cúbico de Lagrange. (a) depois de 125 iterações e (b) depois de 250 iterações.

Agora agregamos um nível de refinamento. Na figura A.3 podemos observar novamente a função aproximada e a solução analítica depois de 125 iterações e depois de 250.

De novo se observa um pequeno 'under and overshooting' e um rebaixamento na altura máxima da 'Gaussiana', devidos à interpolação. Além disso, não se observa nenhum outro efeito indesejado que possa ser atribuído ao fato de estarmos usando refinamento. Na verdade, no caso de ter uma malha com refinamento, os efeitos da interpolação são atenuados. Por conseguinte decidimos estudar alguns trabalhos onde interpoladores que preservam melhor a forma são usados.

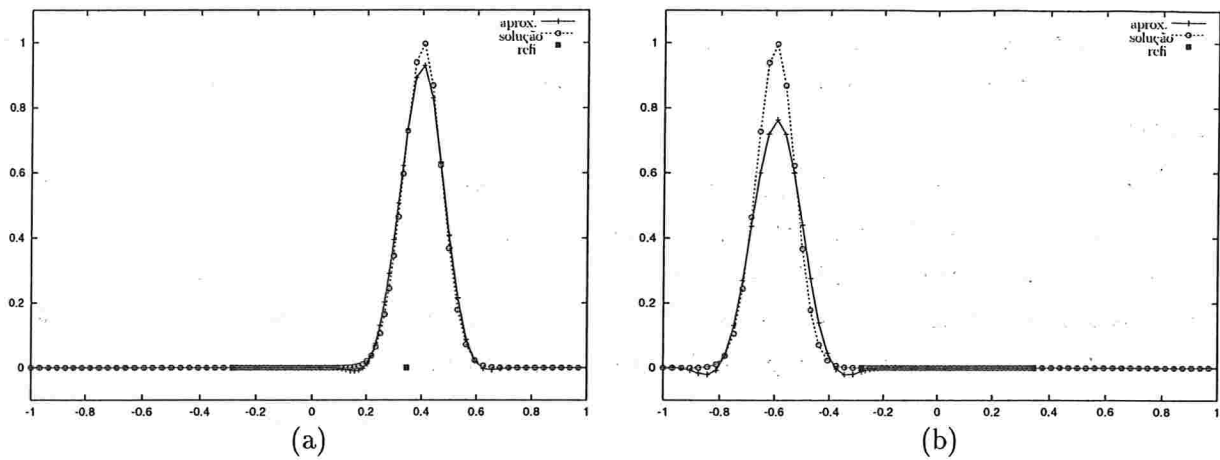


Figura A.3: Solução analítica e função aproximada numa malha com um nível de refinamento utilizando um interpolador cúbico de Lagrange. (a) depois de 125 iterações e (b) depois de 250 iterações. O início e fim da região refinada também são marcadas

## A.1 Interpoladores que preservam a forma

Chamamos de interpoladores que preservam a forma métodos que mantém alguma monotonia ou concavidade/convexidade presente nos dados a serem interpolados. Rasch e Williamson (1990), estudaram diversos interpoladores que preservam forma, baseados em interpolação de Hermite. Para construir qualquer um destes polinômios, precisamos ter como dados, os valores da função nos extremos do intervalo onde encontra-se o ponto a interpolar e os valores da derivada nos extremos do intervalo (ver figura A.4). Como os valores da derivada não estão disponíveis, estes precisam ser estimados, normalmente através de diferenças finitas. Seja  $x$  o ponto onde se quer interpolar, tal que  $x_i \leq x \leq x_{i+1}$  ( $x_1 < x_2 < \dots < x_n$  são os pontos da malha). O interpolador  $P$  tem que satisfazer

$$P(x_k) = f_k$$

e

$$\frac{dP}{dx}(x_k) = d_k$$

com  $f_k = f(x_k)$  e  $d_k \approx f'(x_k)$ , para  $k = i, i + 1$ .

Rasch e Williamson (1990) estudaram vários estimadores de derivada (e com duas restrições para os estimadores, uma de monotonia e a outra de convexidade/concavidade).

Baseados em seus resultados para malhas uniformes, selecionamos o polinômio de Hermite, e os estimadores de derivadas Akima e Hyman, que levam a bons resultados no caso de malhas uniformes. Também colocamos as restrições de monotonia e convexidade/concavidade.

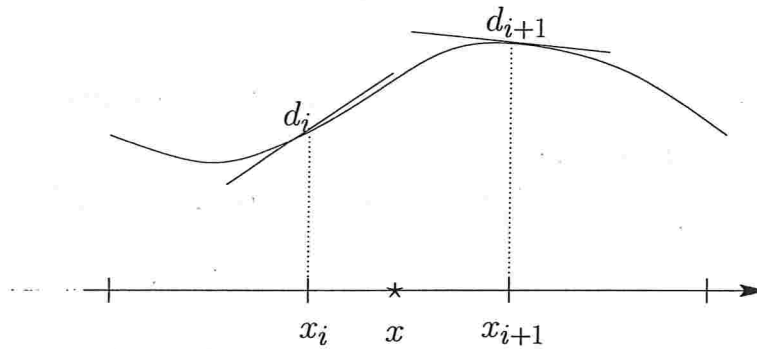


Figura A.4: Esquema de da interpolação de Hermite.

O polinômio de Hermite é dado por

$$P(\theta) = f_{i+1}\theta^3 + (3f_{i+1} - h_id_{i+1})\theta^2(1 - \theta) + (3f_i + h_id_i)\theta(1 - \theta)^2 + f_i(1 - \theta)^3$$

onde

$$h_i = x_{i+1} - x_i$$

$$\theta = \frac{x - x_i}{h_i}$$

e  $d_i$  os estimadores de derivada.

Sejam

$$\Delta_i = \frac{f_{i+1} - f_i}{x_{i+1} - x_i}$$

o estimador de Akima é dado por

$$d_i = \begin{cases} \frac{\alpha\Delta_{i-1} - \beta\Delta_i}{\alpha + \beta} & \text{se } \alpha + \beta \neq 0 \\ \frac{\Delta_{i-1} - \Delta_i}{2} & \text{se } \alpha + \beta = 0 \end{cases}$$

onde

$$\alpha = |\Delta_{i+1} - \Delta_i|, \beta = |\Delta_{i-1} - \Delta_{i-2}|$$

e o estimador de Hyman é

$$d_i = \frac{-f_{i+2} + 8f_{i+1} - 8f_{i-1} + f_{i-2}}{-x_{i+2} + 8x_{i+1} - 8x_{i-1} + x_{i-2}}$$

Nos interpoladores que usam estimadores de derivada, há certas restrições que podem ser impostas, para que eles conservem propriedades de monotonia e convexidade/concavidade presente nos dados. Podemos escolher restringir os estimadores de derivada diferentemente sobre os dois intervalos adjacentes, restringindo-os separadamente e neste caso o interpolador é  $C^0$ . Ou podemos insistir em associar a restrição a ambos intervalos, satisfazendo-as conjuntamente. Em

tal caso o mesmo estimador de derivada é usado para os intervalos adjacentes, e o interpolador é  $C^1$ .

Nos resultados que apresentaremos na próxima seção estaremos usando interpoladores  $C^1$ .

Para que o interpolador seja monótono no intervalo  $[x_{i-1}, x_{i+1}]$  e  $C^1$ , os estimadores de derivadas devem satisfazer a seguinte *Condição Necessária de Monotonia  $C^1$*  (CNM1)

$$\begin{aligned} \text{sign}(\Delta_{i-1}) &= \text{sign}(d_i) = \text{sign}(\Delta_i), & \text{se } \Delta_{i-1}\Delta_i > 0, \\ d_i &= 0, & \text{se } \Delta_{i-1}\Delta_i \leq 0. \end{aligned}$$

Aplicamos esta restrição conforme

$$d_i \leftarrow \begin{cases} d_i, & \text{se } d_i\Delta_j > 0, \\ 0, & \text{se } d_i\Delta_j \leq 0 \end{cases}$$

com  $j = i - 1, i$ .

De Fritsch e Carlson (1980) temos a seguinte condição necessária e suficiente para a monotonia do interpolador cúbico de Hermite. Seja  $\alpha = d_i/\Delta_i$ ,  $\beta = d_{i+1}/\Delta_i$ , então se  $\Delta_i \neq 0$  o interpolador cúbico é monótono se e somente se  $(\alpha, \beta)$  pertencem a  $\mathcal{M}_{ns}$  definido por

$$\mathcal{M}_{ns} = \mathcal{M}_e \cup \mathcal{M}_b$$

onde

$$\mathcal{M}_e(\alpha, \beta) = \{\alpha, \beta : \phi(\alpha, \beta) \leq 0\},$$

$$\mathcal{M}_b(\alpha, \beta) = \{\alpha, \beta : 0 \leq \alpha \leq 3, \quad 0 \leq \beta \leq 3\},$$

e

$$\phi(\alpha, \beta) = (\alpha - 1)^2 + (\alpha - 1)(\beta - 1) + (\beta - 1)^2 - 3(\alpha + \beta - 2).$$

Se  $\Delta_i = 0$  então  $d_i = d_{i+1} = 0$ .

A condição suficiente

$$0 \leq \alpha \leq 3, \quad 0 \leq \beta \leq 3$$

se pode aplicar facilmente. Por tanto um algoritmo para a *Condição Suficiente de Monotonia  $C^1$*  (CSM1) seria

$$d_i \leftarrow \text{sign}(d_i) \min(|d_i|, |3\Delta_{i-1}|, |3\Delta_i|).$$

Se o interpolador é côncavo/convexo no intervalo  $[x_{i-1}, x_i]$  e  $[x_i, x_{i+1}]$  respectivamente o estimador de derivada, deve satisfazer

$$(d_i - \Delta_{i-1})(\Delta_i - d_i) > 0.$$

a qual identificamos como *Condição Necessária de Concavidade/Convexidade  $C^1$*  (CNC1). Tal restrição é aplicada conforme o seguinte algoritmo

$$d_i \leftarrow \begin{cases} d_i, & \text{se } (d_i - \Delta_{i-1})(\Delta_i - d_i) \geq 0, \\ d_{lim}, & \text{se } (d_i - \Delta_{i-1})(\Delta_i - d_i) < 0 \end{cases}$$

onde

$$d_{lim} = \begin{cases} \min(\Delta_{i-1}, \Delta_i), & \text{se } d_i < \Delta_{i-1}, \\ \max(\Delta_{i-1}, \Delta_i), & \text{se } d_i \geq \Delta_{i-1}. \end{cases}$$

## A.2 Teste numérico e resultados

Nesta seção realizamos vários testes, para observar os efeitos de aplicar a interpolação de Hermite com os diferentes estimadores de derivadas e as restrições descritas na seção anterior.

Escolhemos o mesmo caso usado com a interpolação de Lagrange (conforme figuras A.1 a A.3).

Na figura A.5 apresentamos a função aproximada e a solução analítica obtidas com o uso do interpolador de Hermite no modelo unidimensional de advecção passiva, com o estimador de derivada Akima. Neste caso a malha é uniforme.

Na coluna (a) apresentamos os resultados obtidos depois de 125 iterações e (b) depois de 250. Na primeira linha observamos os resultados usando um interpolador cúbico sem restrições, na segunda linha tal que satisfaz a condição necessária e suficiente de monotonia, na terceira linha tal que satisfaz a condição necessária de concavidade/convexidade e na quarta linha colocamos as duas condições.

Pode-se observar, em todos os casos, que a função aproximada praticamente não assume valores negativos nos pontos perto da base da 'Gaussiana', como na figura A.2. Além disso, em todos os casos a altura máxima da 'Gaussiana' é melhor conservada, se bem que há um deslocamento do centro da 'Gaussiana' nos casos com a condição de monotonia.

Na figura A.6 apresentamos os resultados obtidos ao aplicar o interpolador de Hermite, mas neste caso com o estimador de derivada Hyman. Novamente temos uma malha uniforme.

Na coluna (a) apresentamos os resultados obtidos depois de 125 iterações, e (b) depois de 250. Como na figura anterior, na primeira linha observamos os resultados usando o interpolador cúbico sem restrições, na segunda linha tal que satisfaz a condição necessária e suficiente de monotonia, na terceira linha tal que satisfaz a condição necessária de concavidade/convexidade e finalmente na quarta linha colocamos as duas condições.

Novamente, comparando com a figura A.2 observamos que a altura máxima da 'Gaussiana' é melhor conservada. O centro da 'Gaussiana' é bem conservado em todos os casos, a região



onde a função aproximada assume valores negativos (os pontos perto da base da 'Gaussiana') é menor que no caso onde o interpolador cúbico de Lagrange era usado.

Agora consideremos os resultados obtidos ao usar uma malha com um nível de refinamento (fator 2). Na figura A.7 apresentamos os mesmos gráficos que nas figuras A.5 e A.6 só que neste caso estamos usando uma malha com um nível de refinamento local, aplicando o interpolador de Hermite, com o estimador de derivada Akima. E na figura A.8 temos a malha com um nível de refinamento e o interpolador de Hermite com estimador de derivada de Hyman.

Os resultados mostram que os efeitos obtidos com as várias formas de interpolação de Hermite em malhas uniformes são também alcançados em presença dos refinamentos locais. Desta forma em caso de necessidade ou conveniência de uso deste tipo de interpolação, estas podem ser usadas conjuntamente com refinamentos locais. Estas interpolações são no entanto mais custosas que a interpolação de Lagrange, devido à necessidade de se estimar as derivadas. Os resultados que obtivemos em nossos modelos com interpolação de Lagrange foram suficientemente bons, não justificando uma necessidade de uso dos interpoladores de Hermite.

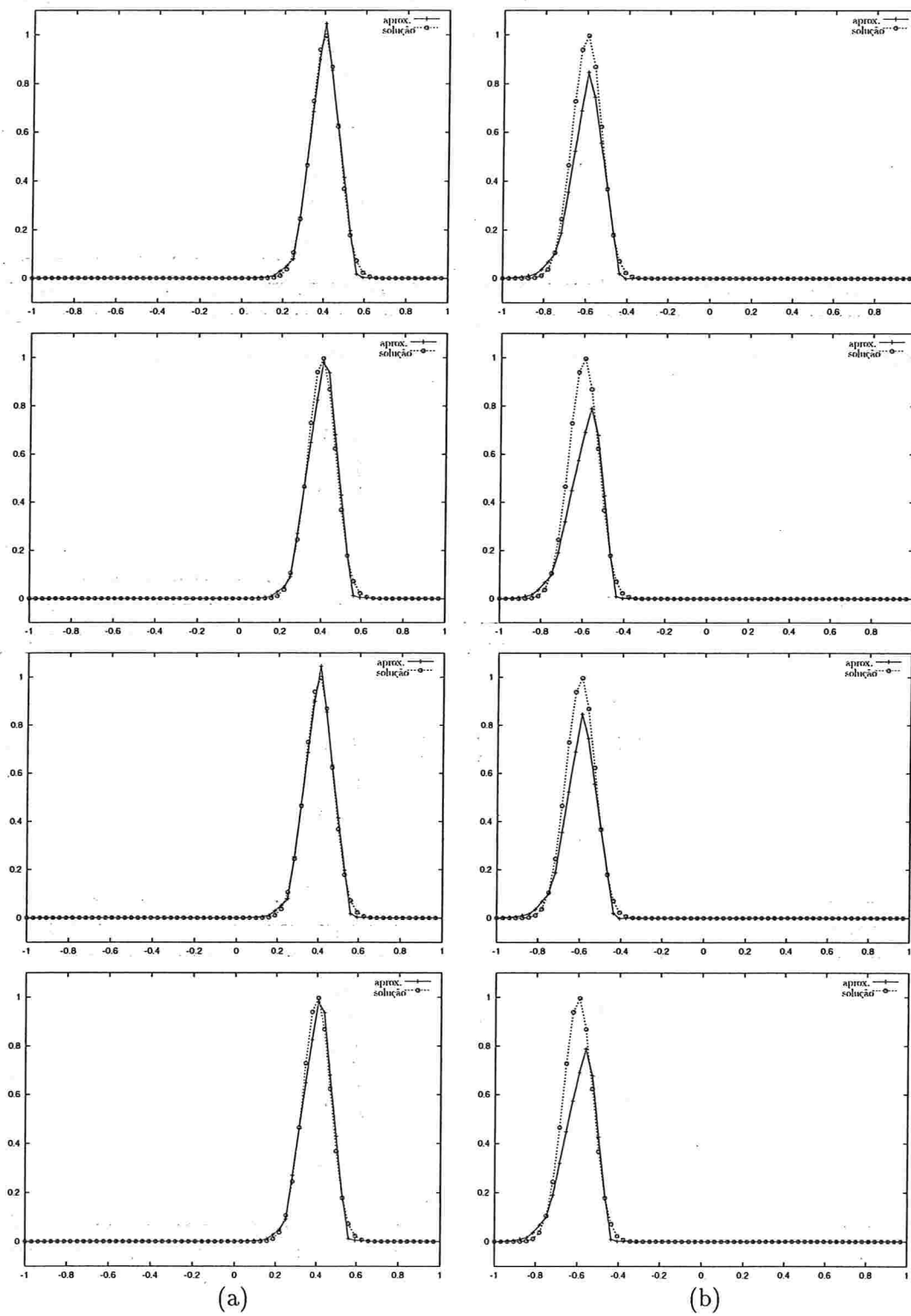


Figura A.5: Solução analítica e função aproximada numa malha uniforme, utilizando estimador Akima. (a) depois de 125 iterações e (b) depois de 250 iterações.

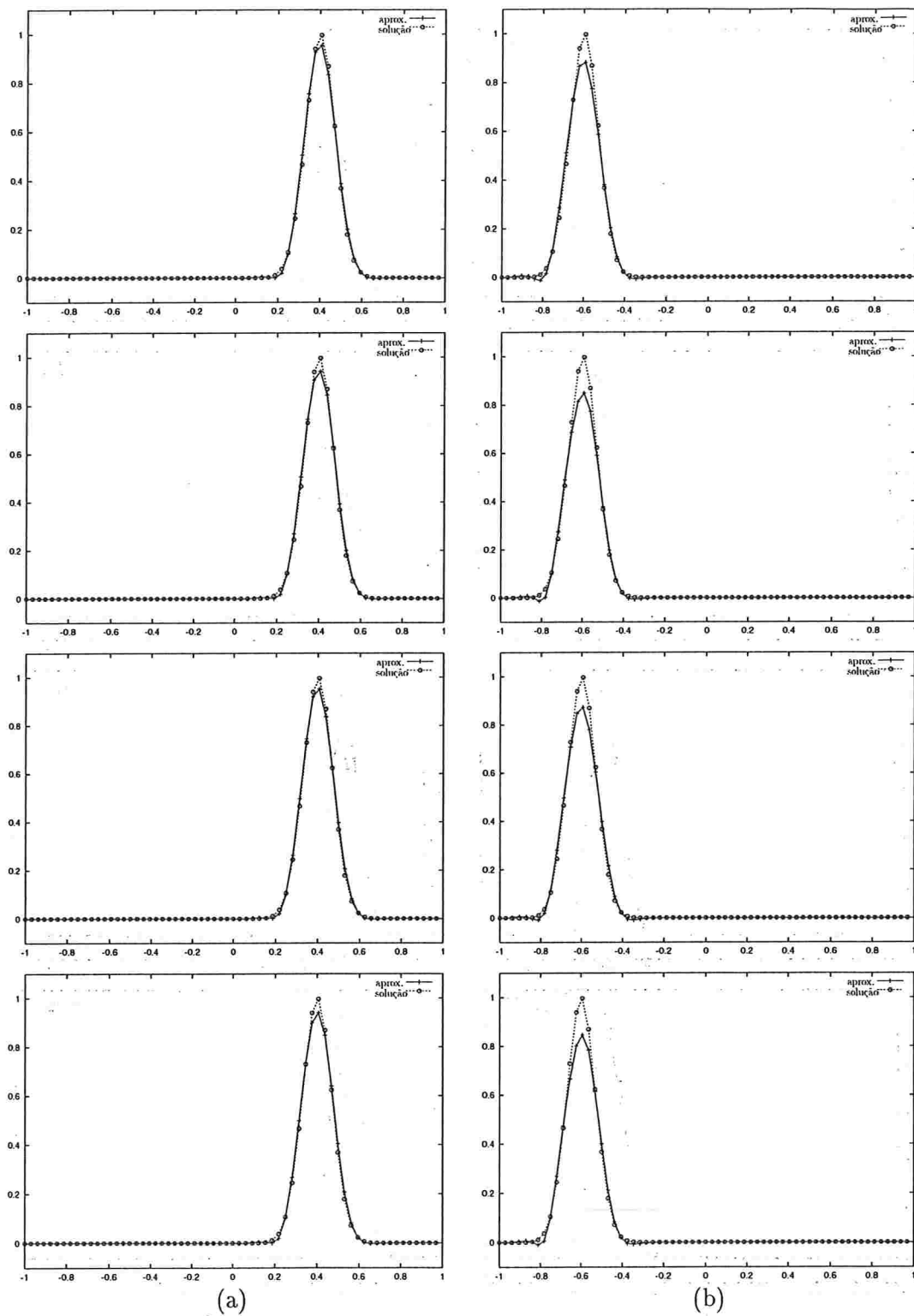


Figura A.6: Solução analítica e função aproximada numa malha uniforme, utilizando estimador de derivada Hyman. (a) depois de 125 iterações e (b) depois de 250 iterações.

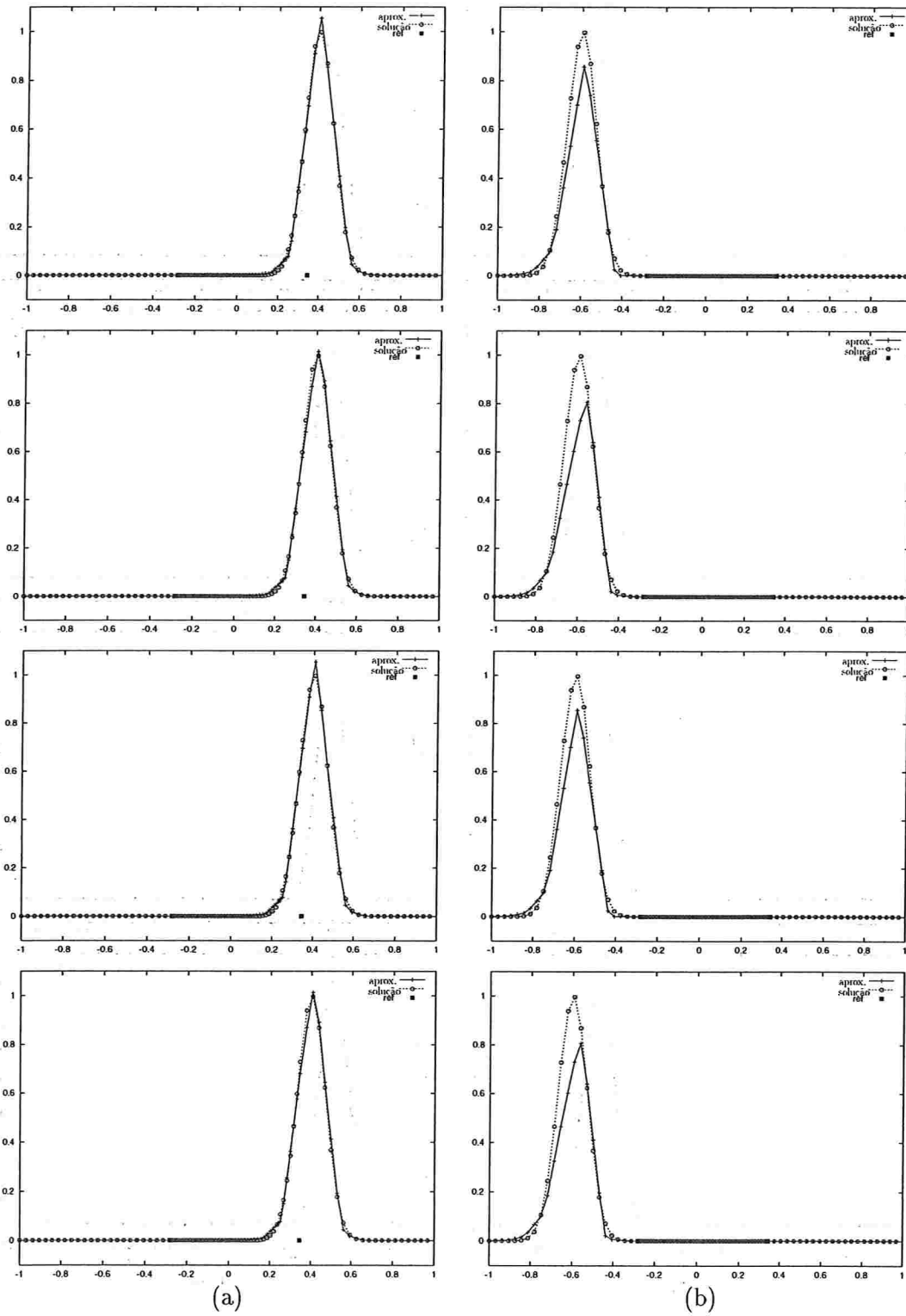


Figura A.7: Solução analítica e função aproximada na malha com um nível de refinamento, utilizando estimador Akima. (a) depois de 125 iterações e (b) depois de 250 iterações.

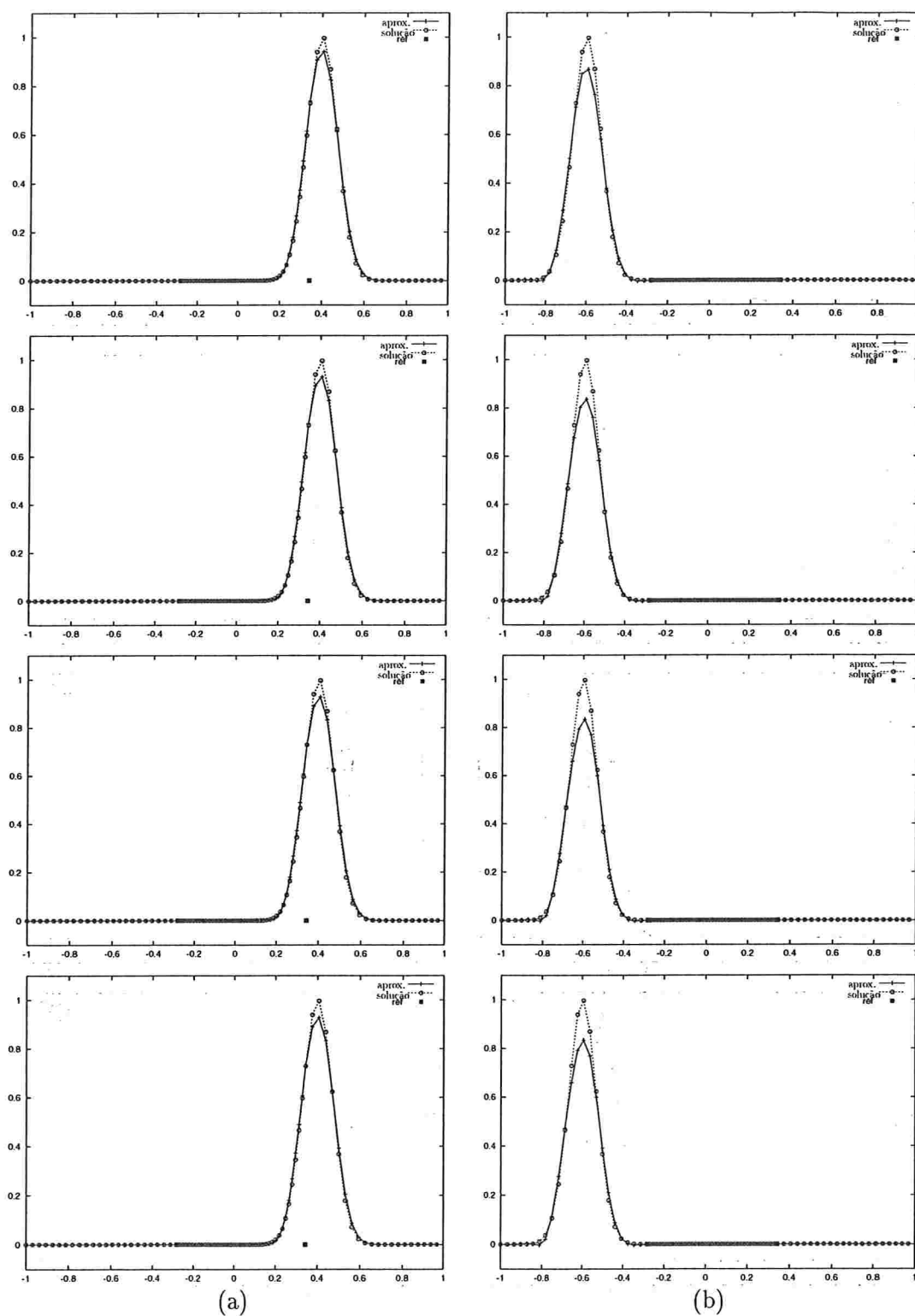


Figura A.8: Solução analítica e função aproximada na malha com um nível de refinamento, utilizando estimador de derivada Hyman. (a) depois de 125 iterações e (b) depois de 250 iterações.



## Transformação do vetor horizontal para o plano tangencial

Damos aqui os detalhes da transformação do vetor horizontal no ponto de partida ou chegada da trajetória (representado por  $R_1$ ) para sua representação no plano tangente do ponto médio de trajetória, ou seja na posição  $R$  na figura 3.1.

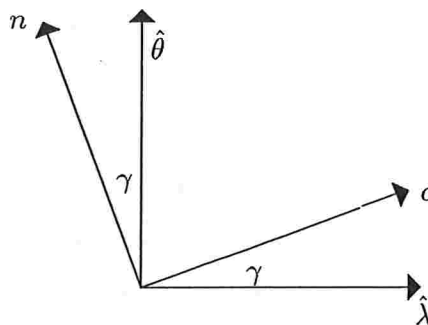


Figura B.1: Relação entre os vetores unitários  $(\hat{\lambda}, \hat{\theta})$  e  $(c, n)$ .

Como descrevemos na seção 3.2, sejam  $(\hat{\lambda}, \hat{\theta})$  as coordenadas do plano tangente em  $R$  (ponto médio da trajetória), definimos como  $\mathbf{c}$  um vetor unitário ao longo da direção do vetor  $\mathbf{v}$  (o vento no ponto médio da trajetória) e  $\mathbf{n}$  o vetor unitário normal à esquerda de  $\mathbf{c}$ , como mostra a figura B.1 onde  $\gamma$  é o ângulo entre  $\mathbf{c}$  e  $\hat{\lambda}$ . Por conseguinte,  $\mathbf{c}$  e  $\mathbf{n}$  formam outra base do plano tangente ao ponto médio da trajetória.

Portanto podemos escrever

$$\mathbf{c} = \cos \gamma \hat{\lambda} + \sin \gamma \hat{\theta} \quad (\text{B.1})$$

$$\mathbf{n} = -\sin \gamma \hat{\lambda} + \cos \gamma \hat{\theta} \quad (\text{B.2})$$

e

$$\mathbf{v} = (u^2 + v^2)^{1/2} \mathbf{c}$$

onde

$$\gamma = \tan^{-1} \frac{u}{v}.$$

Seja  $\alpha$  o deslocamento angular ao longo do grande círculo de trajetória entre  $R$  e  $R_1$ . Supondo que o movimento tem velocidade constante ao longo da trajetória temos

$$\alpha = (u^2 + v^2)^{1/2} \Delta t / a.$$

Além disso, da figura 3.1 podemos deduzir que

$$R_1 = \cos \alpha R + a \sin \alpha \mathbf{c}.$$

De (B.1) temos

$$R_1 = \cos \alpha R + a \sin \alpha (\cos \gamma \hat{\lambda} + \sin \gamma \hat{\theta}). \quad (\text{B.3})$$

Com esta equação relacionamos as coordenadas de  $R_1$  com as da velocidade,  $\alpha$  e  $\gamma$ , e as coordenadas  $(\lambda, \theta)$  da posição  $R$ . Para isso, introduzimos um sistema de coordenadas Cartesiano com origem no centro da esfera e com a base ortonormal  $(\mathbf{I}, \mathbf{J}, \mathbf{K})$ . A expressão das coordenadas esféricas  $(\hat{\lambda}, \hat{\theta}, \mathbf{k})$  no sistema cartesiano é dada por

$$\hat{\lambda} = -\sin \lambda \mathbf{I} + \cos \lambda \mathbf{J}$$

$$\hat{\theta} = -\cos \lambda \sin \theta \mathbf{I} - \sin \lambda \sin \theta \mathbf{J} + \cos \theta \mathbf{K}$$

$$\mathbf{k} = \cos \lambda \cos \theta \mathbf{I} + \sin \lambda \cos \theta \mathbf{J} + \sin \theta \mathbf{K}.$$

Também podemos escrever  $R$  e  $R_1$  neste sistema:

$$R = \cos \lambda \cos \theta \mathbf{I} + \sin \lambda \cos \theta \mathbf{J} + \sin \theta \mathbf{K}$$

$$R_1 = \cos \lambda_1 \cos \theta_1 \mathbf{I} + \sin \lambda_1 \cos \theta_1 \mathbf{J} + \sin \theta_1 \mathbf{K}.$$

Resolvendo (B.3) nas componentes cartesianas, obtemos

$$\cos \lambda_1 \cos \theta_1 = \cos \lambda \cos \theta + \sin \alpha (\sin \lambda \cos \gamma + \cos \lambda \sin \theta \sin \gamma)$$

$$\sin \lambda_1 \cos \theta_1 = \sin \lambda \cos \theta + \sin \alpha (\cos \lambda \cos \gamma - \sin \lambda \sin \theta \sin \gamma)$$

$$\sin \theta_1 = \sin \theta + \cos \theta \sin \alpha \sin \gamma. \quad (\text{B.4})$$



Simplificando, podemos escrever as duas primeiras equações como

$$\cos(\lambda_1 - \lambda) \cos \theta_1 = \cos \theta \sin \alpha \sin \gamma \quad (\text{B.5})$$

$$\sin(\lambda_1 - \lambda) \cos \theta_1 = \sin \alpha \cos \gamma. \quad (\text{B.6})$$

Como  $(\mathbf{c}, \mathbf{n}, \mathbf{k})$  formam uma base, podemos escrever  $\hat{\lambda}_1$  e  $\hat{\theta}_1$  como

$$\begin{aligned} \hat{\lambda}_1 &= (\hat{\lambda}_1 \cdot \mathbf{c})\mathbf{c} + (\hat{\lambda}_1 \cdot \mathbf{n})\mathbf{n} + (\hat{\lambda}_1 \cdot \mathbf{k})\mathbf{k} \\ \hat{\theta}_1 &= (\hat{\theta}_1 \cdot \mathbf{c})\mathbf{c} + (\hat{\theta}_1 \cdot \mathbf{n})\mathbf{n} + (\hat{\theta}_1 \cdot \mathbf{k})\mathbf{k}. \end{aligned}$$

Usando (B.1) e (B.2), a transformação de coordenadas esféricas para coordenadas Cartesianas, e (B.4), (B.5) e (B.6) obtemos

$$\cos \theta_1 \hat{\lambda}_1 = \cos \theta \cos \gamma (\cos \alpha \mathbf{c} - \sin \alpha \mathbf{k}) + (\sin \theta \sin \alpha - \cos \theta \cos \alpha \sin \gamma) \mathbf{n}$$

$$\cos \theta_1 \hat{\theta}_1 = -(\sin \theta \sin \alpha - \cos \theta \cos \alpha \sin \gamma) (\cos \alpha \mathbf{c} - \sin \alpha \mathbf{k}) + \cos \theta \cos \gamma \mathbf{n}.$$

Agora, observando a figura (3.1) podemos notar que

$$\mathbf{c}_1 = \cos \alpha \mathbf{c} - \sin \alpha \mathbf{k}.$$

Substituindo nas equações anteriores obtemos

$$\cos \theta_1 \hat{\lambda}_1 = \cos \theta \cos \gamma \mathbf{c}_1 + (\sin \theta \sin \alpha - \cos \theta \cos \alpha \sin \gamma) \mathbf{n} \quad (\text{B.7})$$

$$\cos \theta_1 \hat{\theta}_1 = -(\sin \theta \sin \alpha - \cos \theta \cos \alpha \sin \gamma) \mathbf{c}_1 + \cos \theta \cos \gamma \mathbf{n}. \quad (\text{B.8})$$

Portanto, acabamos de escrever  $\hat{\lambda}_1$  e  $\hat{\theta}_1$  em função de  $\mathbf{c}_1$  e  $\mathbf{n}$  como queríamos. Como foi explicado na seção 3.2 o próximo passo é rotacionar  $\mathbf{c}_1$  na direção de  $\mathbf{c}$  para produzir os vetores unitários  $(\hat{\lambda}_1)_R$  e  $(\hat{\theta}_1)_R$  no plano tangente no ponto médio da trajetória, a posição  $R$ . Isto é feito só substituindo  $\mathbf{c}_1$  por  $\mathbf{c}$ . Para concluir, escrevemos  $\mathbf{c}$  e  $\mathbf{n}$  em função de  $\hat{\lambda}$  e  $\hat{\theta}$  e obtemos que

$$(\hat{\lambda}_1)_R = \frac{Y \hat{\lambda} + X \hat{\theta}}{\cos \theta_{R_1}}$$

$$(\hat{\theta}_1)_R = \frac{Y \hat{\theta} - X \hat{\lambda}}{\cos \theta_{R_1}}$$

onde

$$X = \sin \alpha \cos \gamma \sin \theta + (1 - \cos \alpha) \sin \gamma \cos \gamma \cos \theta$$

$$Y = \cos \theta - \sin \alpha \sin \gamma \sin \theta - (1 - \cos \alpha) \sin^2 \gamma \cos \theta.$$

Na hora de preparar o algoritmo para o cálculo da projeção no plano tangente, faremos algumas mudanças no cálculo de  $\alpha$  e  $\gamma$ . Primeiro lembremos que só conhecemos o ponto médio da trajetória da malha para  $\phi$ , e portanto, para obter o valor do vento nos pontos médios das trajetórias para as malhas de  $u$  e  $v$  teremos que calcular estes pontos médios, do mesmo jeito que para o cálculo dos pontos de partida destas malhas (ver seção 3.5).

Para evitar o cálculo do valor do vento no ponto médio da trajetória, usaremos outra alternativa para obter  $\alpha$  e  $\gamma$ . O que faremos é utilizar as equações (B.5) e (B.6) e o fato que  $\alpha$  é o deslocamento angular da partícula, ou seja, o ângulo entre o ponto médio da trajetória e o ponto de partida ou o ponto de chegada, para encontrar um cálculo alternativo dos ângulos.

De (B.5) temos que

$$\sin \alpha \sin \gamma = (\cos(\lambda_1 - \lambda) \cos \theta_1) / \cos \theta$$

e de (B.6)

$$\sin \alpha \cos \gamma = \sin(\lambda_1 - \lambda) \cos \theta_1$$

portanto

$$\tan \gamma = \frac{\cos(\lambda_1 - \lambda)}{\cos \theta \sin(\lambda_1 - \lambda)}$$

ou seja

$$\gamma = \tan^{-1} \left( \frac{\cos(\lambda_1 - \lambda)}{\cos \theta \sin(\lambda_1 - \lambda)} \right).$$

Para o cálculo do  $\alpha$  vamos usar o fato que  $\alpha$  é o ângulo entre  $R$  e  $R_1$ , portanto usando a fórmula do cosseno entre dois ângulos obtemos que

$$\alpha = \cos^{-1} (\cos(\lambda_1 - \lambda) \cos \theta_1 \cos \theta + \sin \theta_1 \sin \theta)$$

---

## Bibliografia

- Arakawa, A. 1966: *Computational design for long-term numerical integrations of the equations of atmospheric motion*, J. Comput. Phys., 1, 119-143.
- Barros, S.R.M., Dee, D.P., Dickstein, F., 1990: *A multigrid solver for semi-implicit global shallow-water models*, Atmos. Ocean 28, 24-47.
- Barros, S.R.M., 1991: *Multigrid methods for two-and three-dimensional Poisson-type equations on the sphere*, J. Comput. Phys. 92, 313-348.
- Bates, J.R., Semazzi, F.H.M., Higgins, R.W., Barros, S.R.M., 1990: *Integration of the shallow water equations on the sphere using a vector semi-Lagrangian scheme with a multigrid solver*, Mon. Wea. Rev., 118, 1615-1627.
- Berger, M., Olinger, J., 1984: *Adaptive mesh refinement for hyperbolic partial differential equations*, J. Comput. Phys. 53, 484-512.
- Berger, M., Colella, P., 1989: *Local adaptive mesh refinement for shock hydrodynamics*, J. Comput. Phys. 82, 64-84.
- Brandt, A., 1977: *Multi-level adaptive solutions to boundary-value problems*, Math. Comp. 31, 333-390.
- Brandt, A., 1984: *Multigrid Techniques: 1984 guide with applications to fluid dynamics*, GMD-Studie 85, GMD, St. Augustin, Germany.
- Brandt, A., Diskin B., 1994: *Multigrid solver on decomposed domains*, Cont. Math. 157, 135-151.
- Côté, J., Staniforth, A., 1988: *A two-time-level semi-Lagrangian semi-implicit scheme for spectral models* Mon. Wea. Rev., 116, 2003-2012.
- Côté, J., Staniforth, A., 1990: *An accurate and efficient finite-element global model of the shallow-water primitive equations*, Mon. Wea. Rev. 118, 2707-2717.

- Courtier, P., Geleyn, J.F., 1988: *A global numerical weather prediction model with variable resolution: application to the shallow-water equations*, Quart. J. Roy. Meteor. Soc. 114, 1321-1346.
- Cullen, M.J.P., 1991: *The unified forecast/climate model*, in Proceedings of a Seminar on Numerical methods in atmospheric models, ECMWF, V. 2, 233-260.
- Dietachmayer, G., Droegemeir, K., 1992: *Application of continuous dynamic grid adaption techniques to meteorological modeling Part I: basic form and accuracy*, Mon. Wea. Rev. 120, 1675-1706.
- Fritsch, F.N., Carlson, R.E., 1980: *Monotone piecewise cubic interpolation*, SIAM J. Numer. Anal. 17, 238-246.
- Fulton, S.R., Cielsielski, Schubert, W. H., 1986: *Multigrid methods for elliptic problems: A review*, Mon. Wea. Rev. 114, 943-959.
- Geleyn, J.F., 1998: *Adaptation of spectral methods to non-uniform mapping (global and local)*, in Proceedings of a Seminar on Recent developments in numerical methods for atmospheric modeling, ECMWF, 7-11 September 1998, 226-265.
- Gustafsson, N., 1991: *The HIRLAM model*, in Proceedings of a Seminar on Numerical methods in atmospheric models, ECMWF, V. 2, 115-145
- Hackbusch, W., 1985: *Multi-grid methods and applications*, Springer-Verlag Berlin.
- Haltiner, G.J., Williams, R.T., 1980: *Numerical prediction and dynamic meteorology* Wiley New York.
- Hortal M., Simmons A.J., 1991: *Use of Reduced Gaussian Grids in Spectral Models*, Mon. Wea. Rev. 119, 1057-1074.
- Mc Donald, A., 1989: *A Semi-Lagrangian and semi-implicit two-time-level integration scheme*, Mon. Wea. Rev. 114, 824-830.
- Mc Donald, A., Bates, J.R., 1989: *Semi-lagrangian integration of a gridpoint shallow water model on the sphere*, Mon. Wea. Rev. 117, 130-137.
- Phillips, N. A., 1959: *Numerical integration of the primitive equations on the hemisphere*, Mon. Wea. Rev. 87, 333-345.
- Rasch, P., Williamson, 1990: *On shape-preserving interpolation and semi-Lagrangian transport*, SIAM J. Sci. Stat. Comput. 11, 656-687.
- Ritchie, H., 1987: *Semi-lagrangian advection on a Gaussian grid*, Mon. Wea. Rev., 115, 608-619.
- Ritchie, H., 1988: *Application of the semi-Lagrangian method to a spectral model of the shallow*

*water equations*, Mon. Wea. Rev. 116, 1587-1598.

Robert, A., 1981: *A stable numerical integration scheme for the primitive meteorological equations*, Atmos. Ocean, 19, 35-46.

Robert, A., 1982: *A semi-Lagrangian and semi-implicit numerical integration scheme for the primitive meteorological equations*, Jpn. Meteor. Soc. 60, 319-325.

Ruge, J., McCormick, S., Yee, S., 1995: *Multilevel adaptive methods for semi-implicit solutions of shallow-water equations on a sphere*, Mon. Wea. Rev. 123, 2197-2205.

Skamarock, W., 1989: *Truncation error estimates for refinement criteria in nested and adaptive models*, Mon. Wea. Rev. 117, 872-886.

Skamarock, W., Oliger, J., Street, R., 1989: *Adaptive grid refinement for numerical weather prediction*, J. Comput. Phys. 80, 27-60.

Skamarock, W., Klemp, J., 1993: *Adaptive grid refinement for 2-dimensional and 3-dimensional non hydrostatic atmospheric flow*, Mon. Wea. Rev. 121, 788-804.

Staniforth, A., Côté, J., 1991: *Semi-Lagrangian integration schemes for atmospheric models - a review*, Mon. Wea. Rev. 119, 2206-2223.

Staniforth, A., Côté, J., Gravel, S., Méthot, A., Patoine, A., and Roch, M., 1998 *Environment Canada's gem (global environmental multiscale) model*, in Proceedings of a Seminar on Recent developments in numerical methods for atmospheric modeling, ECMWF, 7-11 September 1998, 144-171.

Staniforth, A., Mitchell, H., 1978: *A variable-resolution finite element technique for regional forecasting with the primitive equations*, Mon. Wea. Rev. 106, 439-447.

Temperton, C., Staniforth, A. 1987: *An efficient two-time-level semi-Lagrangian semi-implicit integration scheme*, Q. J. R. Meteorol. Soc. 113, 1025-1039.

Williamson, D., Rasch, P., 1989: *Two-dimensional semi-Lagrangian transport with shape-preserving interpolation*, Mon. Wea. Rev. 117, 102-129.