

Análise de multirresolução em malhas diádicas

Douglas A. Castro, Andrielber S. Oliveira, Sônia M. Gomes,
Anamaria Gomide, Jorge Stolfi

30 de março de 2008

Resumo

Em numerosas aplicações, é de interesse a representação multiescala dos fenômenos envolvidos. Neste sentido, consideramos representações em malhas diádicas em que os valores discretos são médias nas células da malha. As principais transformações são operadores de previsão e restrição relacionando as informações entre níveis consecutivos de discretização. As diferenças entre os valores exatos e os obtidos pelo operador de previsão são codificadas e representam os detalhes entre os dois níveis. Definimos esquemas adaptativos em que células são removidas da estrutura onde os detalhes entre dois níveis são insignificantes. A estrutura de árvore binária facilita a codificação e permite operações eficientes neste tipo de representação adaptativa.

1 Introdução

Em simulações computacionais de problemas físicos, como em eletromagnetismo computacional, os campos são tratados em termos de valores discretos associados a uma malha. Várias maneiras podem ser usadas para fazer essa discretização. Neste sentido, nosso interesse é em representações em malhas diádicas do domínio espacial em que os valores discretos são médias nas células da malha.

Dizemos que uma malha hierárquica é diádica se uma célula do nível k é dividida em duas células iguais no nível $k + 1$ por uma reta perpendicular ao eixo $(k \bmod 2)$. Atribuímos a cada célula da grade diádica um índice, que é um valor inteiro positivo. Esta indexação é conhecida como código Morton. Quando é necessário saber com mais detalhes o comportamento de uma função em uma região do domínio, refina-se a malha apenas nesta região. Neste caso temos uma malha diádica irregular G .

Em representações multinível, as principais transformações são operadores de previsão e restrição relacionando as informações entre níveis consecutivos de discretização. As

diferenças entre os valores exatos e os obtidos pelo operador de previsão são codificadas e representam os detalhes entre os dois níveis. Definimos esquemas adaptativos em que células são removidas da estrutura onde os detalhes entre dois níveis são insignificantes. A estrutura de árvore binária facilita a codificação e permite operações eficientes neste tipo de representação adaptativa.

Este relatório está organizado da seguinte forma. Na Seção 2, são introduzidos conceitos básicos de estrutura de árvore binária. Na Seção 3 são introduzidas definições principais de malhas diádicas. A Seção 4 é dedicada à construção de uma análise de multirresolução para médias celulares em malhas diádicas. São definidos operadores de previsão e restrição que são os ingredientes básicos para os algoritmos de análise e síntese. Quando queremos passar de um nível mais grosso da malha para um nível mais fino usamos o operador de previsão. Por outro lado, se queremos passar de um nível mais fino para um mais grosso usamos o operador de restrição. Nossos operadores são construídos de maneira que sejam exatos para polinômios até grau 2 em cada variável.

2 Conceitos básicos

Nesta seção apresentaremos alguns conceitos necessários para as seções subsequentes.

Definição 2.1. *Uma árvore binária é um conjunto finito T de elementos denominados nós, tais que:*

- 1- *ou o conjunto $T = \emptyset$ e é dito árvore vazia;*
- 2- *ou existe um nó especial R , chamado raiz de T e os elementos restantes podem ser divididos em dois subconjuntos disjuntos, T_e e T_d . T_e é a sub-árvore à esquerda e T_d é a sub-árvore à direita de R , as quais também são árvores binárias.*

A Fig. 1 ilustra o conceito de árvore binária.

Em qualquer nó da árvore, as raízes das sub-árvores à esquerda e à direita são os filhos esquerdo e direito, respectivamente, do nó. Os nós que não têm filhos são chamados de *folhas*. Neste trabalho consideramos apenas árvores binárias do tipo 0 – 2. Isto significa que um nó não tem filhos ou tem 2 filhos.

A cada nó da árvore pode ser atribuído um número inteiro positivo de maneira a ordená-los. Cada um destes inteiros é denominado índice. A atribuição dos índices é feita da seguinte forma recursiva:

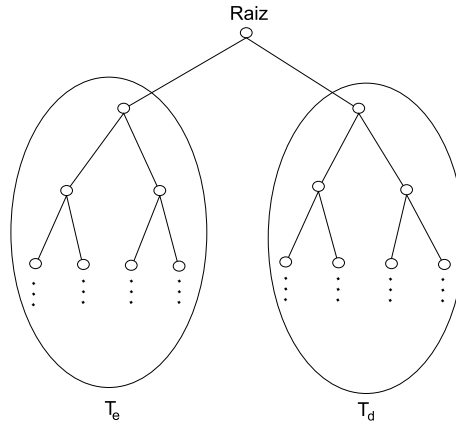


Figura 1: Árvore binária.

$$\begin{aligned}
 I(R) &= 1, \\
 I(n_e) &= 2I(n), \\
 I(n_d) &= 2I(n) + 1;
 \end{aligned}
 \tag{1}$$

em que R é a raiz, n é um nó arbitrário da árvore binária e n_e e n_d são os nós filhos à esquerda e à direita de n , respectivamente. A Fig. 2 ilustra esta enumeração de nós numa árvore do tipo 0 – 2.

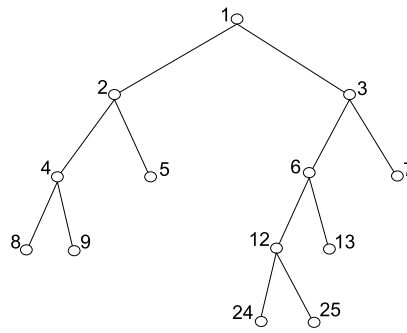


Figura 2: Distribuição de índices na árvore binária.

3 Malhas diádicas

Nesta seção apresentamos a construção de uma família de grades diádicas. Para tal consideramos uma região $R \subset \mathbb{R}^2$, $R = [0, 1] \times [0, 1]$. Começando da raiz $\mathcal{G}_0 = R$, obtemos uma seqüência de grades diádicas $\mathcal{G}_r, r > 0$, fazendo divisões sucessivas das células, sempre pela metade e de maneira cíclica, segundo indicado na Fig. 3. Cada nova divisão é ortogonal a um eixo coordenado distinto, onde atribuímos com $i = 0$ o eixo $\vec{0x}$ e com $i = 1$ o eixo $\vec{0y}$. Cada grade $\mathcal{G}_r$ é formada por componentes de dimensões 0, 1 e 2. Assim uma componente de dimensão 0 é um vértice, de dimensão 1 é uma aresta e de dimensão 2 é uma célula. Note que ao mudar da grade $\mathcal{G}_r$ para $\mathcal{G}_{r+1}$ dobramos a quantidade de componentes.

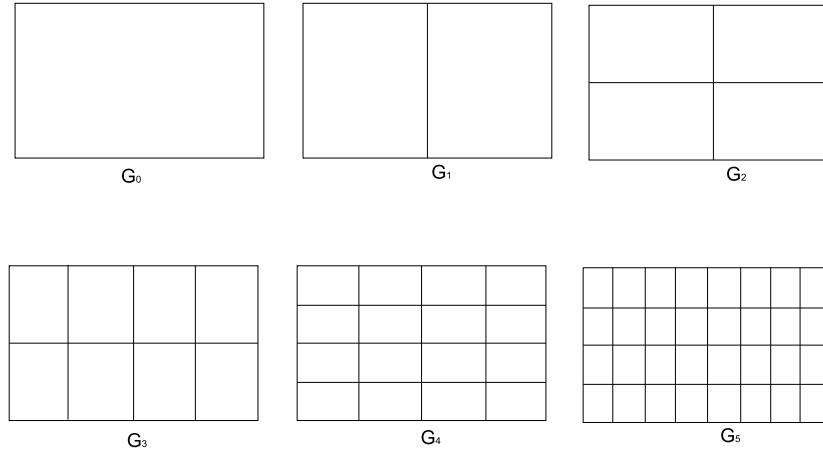


Figura 3: Divisões celulares.

Definição 3.1. Uma multi-grade diádica (multigrid) é o conjunto $\mathcal{G} = \cup_{r=0}^{\infty} \mathcal{G}_r$.

Na prática usamos uma grade diádica finita, em que o processo termina depois de um número finito de divisões.

Definição 3.2. Uma malha diádica finita G é um subconjunto de uma grade diádica finita $\mathcal{G}$.

Uma malha diádica pode ser associada a uma árvore binária do tipo 0 – 2. Para este fim, introduzimos os seguintes conceitos.

Definição 3.3. Sejam a, b células da malha G_r . Dizemos que $a =_i b$ se possuem a mesma projeção sobre o eixo i . Dizemos que $a <_i b$ se todo ponto da célula a tem a i -ésima coordenada menor que a i -ésima coordenada de qualquer ponto da célula b .

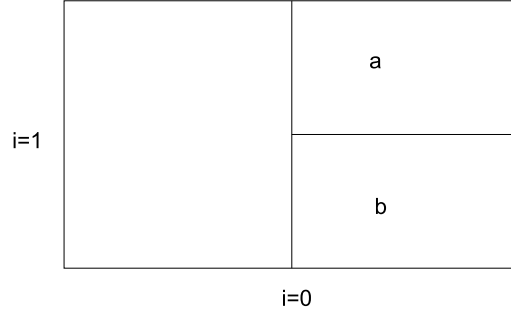


Figura 4: Neste caso, $a =_0 b$ mas $b <_1 a$.

Agora estamos prontos para indexar as células.

Definição 3.4. *O índice da célula raiz é 1. Os filhos da célula de índice I , digamos a e b , com $a <_i b$, possuem índices $2I$ e $2I + 1$ respectivamente.*

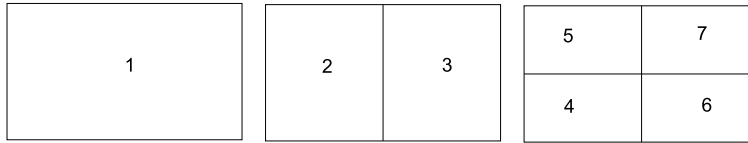


Figura 5: Distribuição dos índices numa grade finita.

As Fig. 5 e 6 ilustram este processo.

Definição 3.5. *O nível ou profundidade de uma célula C é p se ela pertence à grade G_p .*

Por esta definição, vemos que se uma célula está no nível p então seus filhos estão no nível $p + 1$.

Definição 3.6. *Sejam G uma malha diádica finita e $C \in G_r$, $0 \leq r \leq k$. Dizemos que C é uma célula folha se não sofre divisão.*

Um outro conceito que é importante na codificação de uma malha diádica é a posição de uma célula $\delta(C) = (\delta_0, \delta_1)$.

Dado o índice de uma célula podemos determinar tanto sua posição quanto a profundidade. Por outro lado, dado somente o vetor posição não é possível determinar a célula a que fazemos referência, é necessário também que seja dada a sua profundidade.

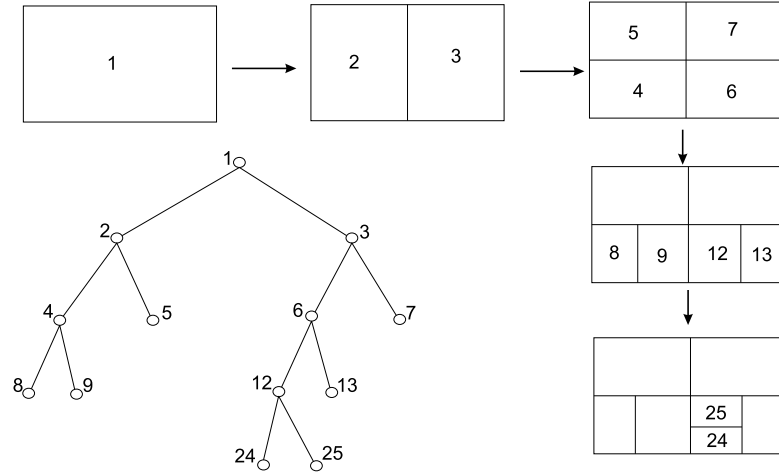


Figura 6: Índices na árvore e na malha.

A forma sistemática de calcular a profundidade a partir do índice I de uma célula é

$$p = \lfloor \log_2 I \rfloor.$$

A justificativa para esta escolha é que todo I está entre 2^r e 2^{r+1} , para algum r , isto é, $2^r \leq I < 2^{r+1}$. Como o $\log$ é uma função crescente temos que $r \leq \log_2 I < r + 1$. Deste modo, a profundidade é tomada como sendo r .

Para montar o vetor posição $\delta(C)$ fazemos o seguinte.

1. Calcula-se a representação binária de I .
2. Elimina-se o dígito mais significativo.
3. Dos dígitos restantes, da esquerda para a direita, escolhem-se, alternadamente, os dígitos que irão formar a representação binária de δ_0 e δ_1 .
4. Transformam-se essas representações para a base decimal.

Analogamente, dado o vetor posição $\delta(C)$ e a profundidade $P(C)$ é possível recuperar o índice da célula C . Para isso, primeiro calculamos quantos bits binários q_i devem ter cada entrada δ_i do vetor posição. Para isso usamos a fórmula

$$q_i = \left\lfloor \frac{P(C)}{2} \right\rfloor + \begin{cases} 1 & \text{se } P(C) \bmod 2 > i, \\ 0 & \text{se } P(C) \bmod 2 \leq i. \end{cases} \quad (2)$$

A seguir, calculam-se as representações binárias de δ_i , completando com zeros à esquerda até obter q_i dígitos. A representação binária de I é assim constituída:

1. Considera-se o primeiro dígito igual a 1
2. De maneira alternada, da esquerda para a direita, fazemos a distribuição dos bits de δ_0 e δ_1

Mostramos essa relação índice $\leftrightarrow$ posição e profundidade através do seguinte exemplo.

Exemplo 3.7. Suponha que é dada uma célula de índice $I = 131$ e queremos descobrir sua posição. Na base 2 temos $I = 10000011$. Note que nesta representação binária temos 7 bits depois do bit mais significativo, isso significa que a profundidade desta célula é 7.

Para montar o vetor $\delta(C)$ fazemos o seguinte:

- Eliminamos o dígito 1 mais significativo.
- Alternadamente escolhemos os dígitos para δ_0 e δ_1 : $I = \overset{*0101010}{10000011}_2$. Na base 2 temos $\delta_0 = 0001$ e $\delta_1 = 001$ e então $\delta(C) = (1, 1)$.

Analogamente, dados o vetor posição $\delta(C) = (1_{10}, 1_{10})$ e a profundidade $P(C) = 7$ é possível recuperar o índice da célula C . Para isso calculamos $q_0 = 3 + 1$ e $q_1 = 3 + 0$. Isto quer dizer que a componente δ_0 do vetor posição δ deve ter 4 dígitos e a componente δ_1 tem 3 dígitos. Dessa forma, na base binária, $\delta(C) = (0001, 001)$. Assim, temos $I = \overset{*0101010}{10000011}_2 = 131_{10}$.

4 Análise multirresolução para médias celulares

Em análise wavelet, funções são representadas por dois tipos de informação [1]. Por um lado estão as informações da função no nível de escala mais refinado. Por outro lado, estão as informações em multinível, que consistem das informações correspondentes ao nível menos refinado e das diferenças de informação entre um nível de escala e o seu consecutivo. Neste contexto, que é conhecida como análise de multirresolução, as principais ferramentas são transformações relacionando esses dois tipos de informação. Por um lado temos o operador de restrição $P_{\ell+1}^\ell$ que produz as informações do nível ℓ a partir das informações do nível $\ell+1$. Outro operador importante é o de previsão $P_\ell^{\ell+1}$ que, a partir das informações do nível ℓ fornece aproximações no nível seguinte mais refinado. É importante que essas transformações possam ser executadas de maneira eficiente, o que significa dizer que os algoritmos devem ser rápidos e estáveis.

Nesta seção apresentamos uma análise de multirresolução para médias celulares em malhas diádicas que seja exata para polinômios de grau menor ou igual a 2 em cada variável.

Se G é uma malha diádica e $C \in G_\ell$ para algum ℓ , definimos a média celular da função f sobre C por

$$\bar{f}_c^\ell = \frac{1}{|C|} \int_{|C|} f(x, y) dx dy, \quad (3)$$

em que $|C|$ é a área da célula C .

No processo de divisão diádica de uma célula C , duas células de igual área C_e e C_d são criadas. Na Fig. 7 são indicadas as duas possibilidades para esta subdivisão. Da aditividade

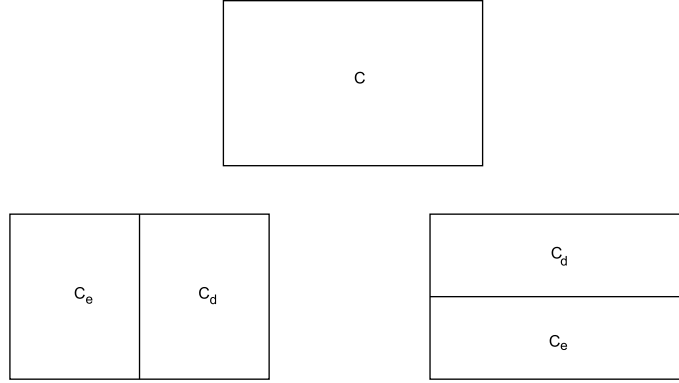


Figura 7: Divisão de uma célula C .

da integral, segue que

$$\bar{f}_C^\ell = \frac{1}{|C|} \int_{|C|} f(x, y) dx dy = \frac{1}{2} (\bar{f}_{C_e}^{\ell+1} + \bar{f}_{C_d}^{\ell+1}). \quad (4)$$

Esta fórmula define o operador de restrição.

Para definir o operador de previsão, vamos considerar primeiro o caso em que a subdivisão da célula é feita perpendicularmente ao eixo $\vec{0x}$, como indicado na Fig. 8. A questão importante é saber como obter aproximações $\tilde{f}_{C_d} \approx \bar{f}_{C_d}$, a partir dos valores $\bar{f}_C$, $\bar{f}_{C^-}$ e $\bar{f}_{C^+}$, em que C^- e C^+ são as células vizinhas à esquerda e à direita de C do mesmo nível ℓ . Sem perda de generalidade, podemos assumir que C_e e C_d estão separadas pelo eixo $\vec{0y}$ e que o vértice inferior esquerdo de C_d está na origem do espaço euclidiano $\mathbb{R}^2$. No texto que se segue consideramos as longitudes Δx e Δy , das arestas de C , e $\delta x = \frac{\Delta x}{2}$, $\delta y = \frac{\Delta y}{2}$, das arestas de C_d (ver Fig. 9).

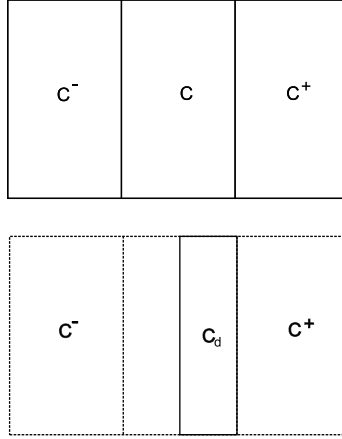


Figura 8: Ilustração de uma célula C , seus vizinhos e filhos.

Propomos um esquema de refinamento interpolatório da forma

$$\tilde{f}_{C_d}^{\ell+1} \approx \bar{f}_C^\ell + \alpha [\bar{f}_{C^+}^\ell - \bar{f}_{C^-}^\ell] \quad (5)$$

em que α deve ser escolhido de modo que a aproximação seja exata para polinômios de até grau 2. Basta testar o esquema de refinamento para os monômios $1, x, y, xy, x^2, y^2$.

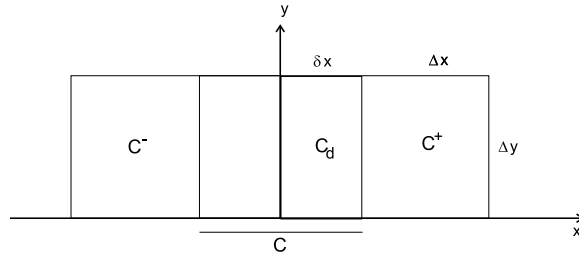


Figura 9: Posição de uma célula C_d em $\mathbb{R}^2$.

- Para $f = 1$ temos,

$$\bar{f}_{C_d}^{\ell+1} = \frac{1}{|C_d|} \int \int dx dy = 1, \quad (6)$$

$$\tilde{f}_{C_d}^{\ell+1} = \frac{1}{|C|} \int_{-\delta x}^{\delta x} \int_0^{\Delta y} dx dy + \alpha \left[\frac{1}{|C^+|} \int_{\delta x}^{3\delta x} \int_0^{\Delta y} dx dy - \frac{1}{|C^-|} \int_{-3\delta x}^{-\delta x} \int_0^{\Delta y} dx dy \right] = 1. \quad (7)$$

Portanto, para uma função constante $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1}$, e a fórmula é sempre exata, para qualquer valor de α .

- Para $f = y$,

$$\bar{f}_{C_d}^{\ell+1} = \frac{\Delta y}{2} = \tilde{f}_{C_d}^{\ell+1} \quad (8)$$

Logo, $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1}$ e a fórmula também é exata para qualquer valor de α

- Analogamente, se $f = y^2$

$$\bar{f}_{C_d}^{\ell+1} = \frac{\Delta y^2}{3} = \tilde{f}_{C_d}^{\ell+1} \quad (9)$$

e a fórmula também é exata para $f = y^2$, independentemente do valor de α .

- Para $f = x$ aparece a primeira restrição sobre α . Neste caso temos

$$\bar{f}_{C_d}^{\ell+1} = \frac{\delta x}{2}, \quad (10)$$

$$\tilde{f}_{C_d}^{\ell+1} = 2\alpha\Delta x. \quad (11)$$

Mas para termos $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1}$ temos que escolher α de forma que

$$\frac{\delta x}{2} = 2\alpha\Delta x. \quad (12)$$

Lembrando que $\delta x = \frac{\Delta x}{2}$, temos

$$\frac{\delta x}{2} = 4\alpha\delta x, \quad (13)$$

que nos fornece a relação

$$\alpha = \frac{1}{8}. \quad (14)$$

Logo, para $f = x$ temos que $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1} \iff \alpha = \frac{1}{8}$.

- No caso em que $f = xy$, temos

$$\bar{f}_{C_d}^{\ell+1} = \frac{\delta x \Delta y}{4}, \quad (15)$$

$$\tilde{f}_{C_d}^{\ell+1} = 2\alpha \Delta y \delta x, \quad (16)$$

Mais uma vez, para termos $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1}$ temos que escolher α tal que

$$\frac{\delta x \Delta y}{4} = 2\alpha \Delta y \delta x, \quad (17)$$

ou seja,

$$\alpha = \frac{1}{8}. \quad (18)$$

Logo, para $f = xy$ temos que $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1} \iff \alpha = \frac{1}{8}$. Note que essa é a mesma exigência do caso em que $f = x$.

- Para $f = x^2$ também não há restrições sobre α , pois

$$\bar{f}_{C_d}^{\ell+1} = \frac{\Delta x^2}{12} = \tilde{f}_{C_d}^{\ell+1}. \quad (19)$$

Logo, para $f = x^2$ vemos que $\bar{f}_{C_d}^{\ell+1} = \tilde{f}_{C_d}^{\ell+1}$, não implicando em restrição sobre α .

Assim, a fórmula

$$\tilde{f}_{C_d}^{\ell+1} = \bar{f}_C^\ell + \alpha [\bar{f}_{C^+}^\ell - \bar{f}_{C^-}^\ell], \quad \alpha = \frac{1}{8}, \quad (20)$$

é exata para $f = 1, x, y, xy, x^2, y^2$.

Por simetria, no caso em que a divisão ocorre paralelamente ao eixo $\overrightarrow{0x}$, como representado na Fig. 10, a fórmula

$$\tilde{f}_{C_d}^{\ell+1} = \bar{f}_C^\ell + \alpha [\bar{f}_{C^+}^\ell - \bar{f}_{C^-}^\ell], \quad \alpha = \frac{1}{8}, \quad (21)$$

é exata para todos os polinômios de grau menor ou igual a 2.

Como sabemos aproximar $\tilde{f}_{C_d}^{\ell+1}$, podemos falar também em erro de aproximação. Este erro é medido pelo coeficiente wavelet, que é definido por

$$d_{C_d}^\ell = \bar{f}_{C_d}^{\ell+1} - \tilde{f}_{C_d}^{\ell+1}. \quad (22)$$

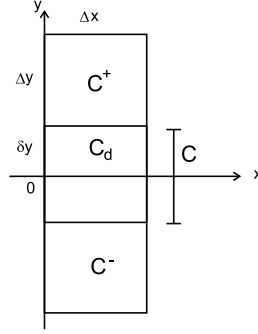


Figura 10: Posição de uma célula C_d em $\mathbb{R}^2$.

Ou seja, os coeficientes wavelet são definidos como erro de previsão.

Com os operadores de restrição e previsão podemos fazer os algoritmos de análise e síntese entre dois níveis de refinamento. Na análise passamos de uma malha mais fina para uma mais grossa. Na síntese passamos da malha mais grossa para a mais fina.

Análise:

$$\bar{f}_C^\ell = \frac{1}{2}[\bar{f}_{C_e}^{\ell+1} + \bar{f}_{C_d}^{\ell+1}] \quad (23)$$

$$d_C^\ell = \bar{f}_{C_d}^{\ell+1} - \tilde{\bar{f}}_{C_d}^{\ell+1}. \quad (24)$$

Síntese:

$$\bar{f}_{C_d}^{\ell+1} = d_C^\ell + \tilde{\bar{f}}_{C_d}^{\ell+1} \quad (25)$$

$$\bar{f}_{C_e}^{\ell+1} = 2\bar{f}_C^\ell + \bar{f}_{C_d}^{\ell+1}. \quad (26)$$

O algoritmo de análise pode ser apresentado em multinível pelo diagrama

$$\begin{array}{ccccccc} \bar{f}^{\ell+1} & \longrightarrow & \bar{f}^\ell & \longrightarrow & \bar{f}^{\ell-1} & \longrightarrow & \dots \longrightarrow \bar{f}^0 \\ & \searrow & \searrow & \searrow & \searrow & & \\ & d^\ell & d^{\ell-1} & d^{\ell-2} & d^0 & & \end{array} \quad (27)$$

e o de síntese se expressa como

$$\begin{array}{ccccccc}
 \bar{f}^0 & \longrightarrow & \dots & \longrightarrow & \bar{f}^{\ell-1} & \longrightarrow & \bar{f}^{\ell} & \longrightarrow & \bar{f}^{\ell+1} \\
 & \nearrow & & \nearrow & & \nearrow & & \nearrow & \\
 d^0 & & d^{\ell-2} & & d^{\ell-1} & & d^{\ell} & &
 \end{array} \tag{28}$$

5 Trabalhos futuros

Neste relatório descrevemos uma estrutura de malhas diádicas e uma análise de multirresolução para médias celulares definidas nas células dessa malha. O próximo passo consiste na definição de algoritmos adaptativos que permitam reduzir a malha pela eliminação de nós associados a células em que o coeficiente wavelet possa ser desprezado. A implementação computacional requer uma codificação cuidadosa.

Referências

- [1] A. Cohen. Wavelet Methods in Numerical Analysis. Handbook of Numerical Analysis, Ph. Ciarlet and J. L. Lions eds., 1998.
- [2] C. G. S. CARDOSO, *Grades diádicas adaptativas para simulação de escoamento de petróleo*, Dissertação de Mestrado, IC-Unicamp, 2004.
- [3] C. G. S. CARDOSO, M. C. CUNHA, A. GOMIDE, D. J. SCHIOZER, J. STOLFI *Finite Elements on Dyadic Grids with Applications*, Mathematics and Computers in Simulation 73 (2006) 87-104