



Análise da Convergência da Solução de Equações Lineares Elípticas sob um Esquema de Diferenças Finitas Generalizadas (MDFG)

Daniel R. Izquierdo

ABSTRACT: The Generalized Finite Difference Method as a meshless method alternative is used to solve partial differential equations in domains with high irregular geometry. A proof of convergence of GFDM is given studying the consistency of truncation error of linear elliptic partial equation problems at 2D, using n -degree polynomial. As an example, the convergence of method is calculated for a bi-dimensional Poisson equation problem supported over a disperse nodes net representing a rectangular domain.

Key Words: Difference Finite Method, Meshless Methods, Irregular grids.

Contents

1	Introdução	101
2	Célula computacional	103
3	Polinômios de aproximação	103
4	Aproximação das diferenças finitas generalizadas	104
5	Consistência	106
6	Análise das EDP em duas dimensões.	108
7	Solução de equações elípticas bi-dimensionais com o MDFG:	
	caso equação de Poisson	109
8	Conclusões.	114

1. Introdução

O método de diferenças finitas generalizadas (MDFG) é desenvolvido a partir do método de aproximações por mínimos quadrados localizados (MLS) [21], o qual fornece uma alternativa às interpolações clássicas de funções a partir de seus valores dados em uma série de pontos distribuídos irregularmente. Recentemente, o MLS para a solução numérica de equações diferenciais tem recebido grande destaque, especialmente na literatura de engenharia mecânica, gerando uma série de métodos chamados *meshless*, os quais possibilitam uma aproximação numérica

a partir de um conjunto de pontos que podem não ter como suporte uma malha ou triangulação. A classificação de um método como sendo do tipo meshless ainda não está estabelecido na literatura. Neste trabalho é utilizada a definição dada por Krysl e Belytschko [11], na qual um método é considerado meshless se as bases da aproximação são construídas a partir de um suporte arbitrário gerado por uma coleção de nós distribuídos irregularmente. Uma característica atrativa dos métodos meshless é sua facilidade de adaptação em espaços arbitrários, e a flexibilidade na implementação de esquemas adaptativos.

Os métodos meshless podem ser divididos em duas categorias: os métodos baseados em princípios variacionais e métodos que atuam diretamente nas equações diferenciais governantes. Na primeira categoria se tem os seguintes métodos: *smooth particle hydrodynamics* (SPH) [8] [16]; *diffuse element methods* (DEM) [17]; *element free Galerkin* (EFG) [3]; *reproducing Kernel particle* (RKP) [13]; *h-p cloud method* [6]; *partition of unit finite element method* (PUFEM) [2]; *meshless local Petro-Galerkin* (MLPG) e *local boundary integral equation* (LBIE) [1]. Tais métodos têm como característica comum a utilização de uma integração numérica para o estabelecimento das equações discretas do sistema. Integrações numéricas tem efeitos sob a precisão e convergência da solução destes métodos meshless, mas apresentam dificuldades devido às seguintes razões: as integrações numéricas geralmente são complexas devido à complexidade da função; é muito difícil escolher uma ordem de integração com todos os fatores considerados; além disso, uma baixa ordem de integração gera soluções de baixa precisão, enquanto integrações de alta ordem aumentaram excessivamente o custo computacional.

Na segunda categoria de métodos meshless é considerado o método de diferenças finitas generalizadas (MDFG), no qual um conjunto de equações discretas são estabelecidas diretamente a partir das equações diferenciais. Uma vantagem deste método está na não utilização de integrações numéricas. A idéia de utilizar nós postos irregularmente num domínio para a obtenção das aproximações das diferenças finitas não é nova, Jensen nos anos 70, apresenta um método de diferenças finitas o qual utiliza células (ou estrelas) irregulares com seis pontos [10]. Utilizando a expansão de séries de Taylor ele obtém uma formulação de diferenças finitas a qual aproxima derivadas até de segunda ordem. A principal desvantagem deste método é que apresenta frequentes singularidades ou mal condicionamento da célula. Perrone e Kao sugerem a adição de mais nós na célula e a aplicação de uma média para a geração dos coeficientes das diferenças finitas [19]. Liska e Orkisz tem feito uma interessante contribuição ao desenvolvimento do método no que se refere à seleção de células na tentativa de eliminar os problemas indicados acima, aplicando o método na solução de problemas lineares e não-lineares [12]. Trabalhos como os de Luo e Hüßler [14], Benito [4], Marshall e Grand [15], e Gossler [9] fazem uso do método na construção de esquemas de diferenças finitas, expandido suas aplicações para a solução de diferentes problemas. Nos últimos anos, na aplicação do método explicitamente, Ureña [20] mostra uma análise do erro na solução de problemas de advecção-difusão. Apesar das vantagens oferecidas por este método, como sua facilidade de implementação, flexibilidade e baixo custo, esta categoria de métodos meshless tem recebido pouca atenção comparada com os métodos variacionais da

primeira categoria.

Embora o MDFG seja um método meshless (o qual utiliza um conjunto de nós espalhados no domínio, os quais podem não ter nenhuma conectividade entre eles), é interessante utilizar malhas (estruturadas ou não estruturadas) como suporte dos nós, garantindo uma melhor distribuição dos nós sobre o domínio e uma serie de relações de vizinhança entre eles. O objetivo principal deste trabalho é estudar a convergência do MDFG, quando aplicado na solução de problemas de equações lineares elípticas. Uma análise do erro que o método introduz, dependendo das posições dos nós utilizados nas aproximações.

2. Célula computacional

Dada uma malha $G(V)$ que descomponha o domínio $D \subset \mathbb{R}^d$, gerando um conjunto de vértices $V = \{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ de G . Para uma EDP, cada vértice $\mathbf{v}_o \in V$ tem associada uma *célula computacional* C_o , como o subconjunto de vértices de V utilizados para discretizar a EDP em $\mathbf{v}_o$. Neste trabalho $G(v)$ é uma malha triangular simplicial a qual pode ser estruturada ou não estruturada, onde cada vértice de C_o comparte uma aresta com $\mathbf{v}_o \in V$ (ver figura 1).

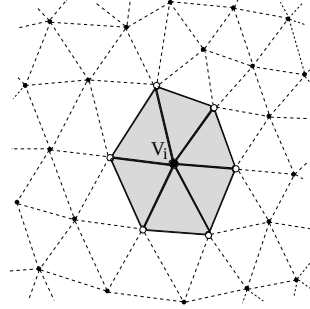


Figure 1: Célula computacional.

Definimos o *raio da célula computacional* de C_o , denotado por ρ_o , como a mínima distância entre $\mathbf{v}_o$ e $\mathbf{v}_k \in C_o$ e a *densidade da malha* como $h = \min\{\rho_o \mid \mathbf{v}_o \in V\}$. Muitos algoritmos de geração de malhas precisam do valor de h como um parâmetro de entrada, permitindo caracterizar a malha.

3. Polinômios de aproximação

O MDFG discretiza as EDP numa serie de vértices sobre o domínio D a partir de um polinômio de aproximação W o qual é calculado utilizando mínimos quadrados. Seja $\mathcal{P}_s^d$ o espaço dos polinômios de grau s d -dimensional (d variáveis) que geram a W , cujas bases estão compostas pelos monômios $P^{(1)}, P^{(2)}, \dots, P^{(n)}$. Para um vetor $\mathbf{r} = \mathbf{r}(x^1, x^2, \dots, x^d) \in \mathbb{R}^d$, cada monômio $P^{(i)}$ pode ser escrito como $P^{(i)} = P^{(i)}(\mathbf{r}) = (x^1)^{\mu_1(i)}(x^2)^{\mu_2(i)} \dots (x^d)^{\mu_d(i)}$, onde $\mu_1(i), \mu_2(i), \dots, \mu_d(i)$ são números naturais não negativos. Por facilidade, cada um dos monômios base $P^{(i)}$ pode ser

associado a um vetor de coordenadas $p_i = p_i(\mu_1(i), \mu_2(i), \dots, \mu_d(i))$, num espaço $\mathbb{N}^d$. Assim, pode ser obtidos imediatamente todos os monômios base de grau r a partir de seus pontos análogos que estão no plano $\mu_1(i) + \mu_2(i) + \dots + \mu_d(i) = r$, no espaço $\mathbb{N}^d$. Se $\mathcal{P}_s^d$ tem uma base composta por todos os monômios com grau igual o menor a s , então se diz que $\mathcal{P}_s^d$ é um espaço polinômial completo.

Lemma 3.1. *Seja um monômio base $P^{(j)}$ do espaço $\mathcal{P}_s^d$. O grau dele, denotado como $N_d(j)$ pode ser calculado a partir da seguinte expressão:*

$$N_d(j) = \min \left\{ m \mid \frac{(d+m)!}{d! m!} \geq j \right\}$$

Proof: É imediata a demonstração já que um espaço polinômial completo $\mathcal{P}_r^d$ tem uma dimensão de $(d+r)!/d! r!$. $\square$

Lemma 3.2. *Qualquer monômio $P^{(i)}$ da base do espaço polinomial completo $\mathcal{P}_s^d$ com $N_d(i) \geq 2$ pode ser construído a partir dos monômios base de grau 1 como: $P^{(i)} = (P^{(1)})^{\mu_1(i)} \dots (P^{(d)})^{\mu_d(i)}$.*

4. Aproximação das diferenças finitas generalizadas

Seja $f : D \subset \mathbb{R}^d \rightarrow \mathbb{R}$ uma função de classe $\mathcal{C}^p$ e $G(V)$ uma decomposição simplicial de D . Seja $\mathbf{v}_o$ um vértice de $G(V)$ posicionado em $\mathbf{r}_o$ e C_o a célula computacional associada a $\mathbf{v}_o$. Considera-se um sistema de coordenadas local proprio de $\mathbf{v}_o$ dado por $\bar{\mathbf{r}} = \mathbf{r} - \mathbf{r}_o = (x^1 - x_o^1, x^2 - x_o^2, \dots, x^d - x_o^d)$. Neste sistema de coordenadas local pode ser aproximado $f(\mathbf{r})$ na vizinhança de $\mathbf{v}_o$ com uma função polinomial $\bar{f}_o(\bar{\mathbf{r}})$, tal que $\bar{f}_o = W_o(\bar{\mathbf{r}})$, onde $W_o \in \mathcal{P}_s^d$. Os monômios da base de $\mathcal{P}_s^d$ de grau 1 podem ser definidos em relação a $\mathbf{v}_o$ por $P^{(i)} = P^{(i)}(\bar{\mathbf{r}}) = (x^i - x_o^i)$, para $i = 1, \dots, d$; e utilizando o lemma 3.2, os demais elementos da base são obtidos. Portanto, o polinômio de aproximação é dado por $W_o(\bar{\mathbf{r}}) = c_0 P^{(0)} + c_1 P^{(1)} + c_2 P^{(2)} + \dots + c_n P^{(n)}$, com $n = \dim \mathcal{P}_s^d$.

Quando $\bar{\mathbf{r}} = \mathbf{0}$ obtemos $W_o = c_0 P^{(0)} = c_0$, portanto, para garantir a aproximação com a função $f(\mathbf{r})$, sempre tem que cumprir-se com $c_0 = f(\mathbf{r}_o)$. É preciso achar o conjunto coeficientes c_i ($i = 1, \dots, n$) que ajustem melhor à função de aproximação $\bar{f}$ com f . Neste trabalho, tal conjunto de coeficientes é calculado pela técnica de mínimos quadrados utilizando os vértices de C_o . Assim, nosso problema consiste em solucionar o seguinte sistema linear:

$$\mathbf{A}\mathbf{c} = \mathbf{b} \tag{4.1}$$

onde $\mathbf{c}^T = (c_1, \dots, c_n)$ é o vetor a ser calculado, $\mathbf{A}$ é uma matriz $n \times n$ simétrica cujos elementos a_{ij} são dados por:

$$a_{ij} = \sum_{\mathbf{v}_k \in C_o} P^{(i)}(\bar{\mathbf{r}}_k) P^{(j)}(\bar{\mathbf{r}}_k); \quad i, j = 1, \dots, n \tag{4.2}$$

e $\mathbf{b}$ é um vetor cujos compoentes b_i são:

$$b_i = \sum_{\mathbf{v}_k \in C_o} (f(\mathbf{r}_k) - f(\mathbf{r}_o)) P^{(i)}(\bar{\mathbf{r}}_k) \quad (4.3)$$

Utilizando a regra de Laplace, o determinante da matriz $\mathbf{A}$ pode ser escrita como:

$$\det(A) = \sum_{j=1}^n a_{j,i} \det(\mathbf{A}_{[j,i]}) \quad (4.4)$$

onde $\mathbf{A}_{[j,i]}$ é uma matriz $(n-1) \times (n-1)$ obtida de $\mathbf{A}_{[j,i]}$ ao eliminar a linha j e a coluna i . Pela regra de Cramer, os coeficientes c_i são calculados como:

$$c_i = \sum_{j=1}^n b_j S_j^{(i)}, \quad \text{onde } S_j^{(i)} = \frac{\det(\mathbf{A}_{[j,i]})}{\det(\mathbf{A})} \quad (4.5)$$

Sustituindo (4.3) em (4.5) obtemos:

$$c_i = \sum_{j=1}^n \sum_{\mathbf{v}_k \in C_o} [f(\mathbf{r}_k) - f(\mathbf{r}_o)] P^{(j)}(\bar{\mathbf{r}}_k) S_j^{(i)} \quad (4.6)$$

Como $f(\mathbf{r}_k) = f(\mathbf{r}_o + \bar{\mathbf{r}}_k) = f(x_o^1 + P^{(1)}(\bar{\mathbf{r}}_k), x_o^2 + P^{(2)}(\bar{\mathbf{r}}_k), \dots, x_o^d + P^{(d)}(\bar{\mathbf{r}}_k))$, a aproximação de Taylor de $f(\mathbf{r}_k)$ pode ser escrita utilizando o teorema multinomial como:

$$f(\mathbf{r}_k) = \sum_{l=0}^{\infty} \frac{1}{\pi(l)} \prod_{m=1}^d \left(P^{(m)}(\bar{\mathbf{r}}_k) \frac{\partial}{\partial x^m} \right)^{\mu_m(l)} f(\mathbf{r}_o) \quad (4.7)$$

onde $\pi(l) = \prod_{m=1}^d \mu_m(l)!$. Definindo o operador diferencial

$$D_l = \prod_{m=1}^d \left(\frac{\partial}{\partial x^m} \right)^{\mu_m(l)} = \frac{\partial^{N_d(l)}}{(\partial x^1)^{\mu_1(l)} \dots (\partial x^d)^{\mu_d(l)}} \quad (4.8)$$

podemos reescrever (4.7) utilizando o lemma 3.2, como:

$$f(\mathbf{r}_k) = f(\mathbf{r}_o) + \sum_{l=1}^{\infty} \frac{1}{\pi(l)} P^{(l)}(\bar{\mathbf{r}}_k) [D_l f(\mathbf{r}_o)] \quad (4.9)$$

Substituindo (4.9) em (4.6) obtemos:

$$c_i = \sum_{l=1}^{\infty} \frac{1}{\pi(l)} [D_l f(\mathbf{r}_o)] T_{i,l} \quad (4.10)$$

onde:

$$T_{i,l} = \sum_{j=1}^n \sum_{\mathbf{v}_k \in C_o} P^{(l)}(\bar{\mathbf{r}}_k) P^{(j)}(\bar{\mathbf{r}}_k) S_j^{(i)} \quad (4.11)$$

Proposition 4.1. *O termo $T_{i,l}$ está dado por:*

$$T_{i,l} = \frac{\det(\mathbf{K}_i^l)}{\det(\mathbf{A})}$$

onde $\mathbf{K}_i^l$ é obtido apartir da matriz $\mathbf{A}$ ao reemplazar a coluna i pelo vetor β^l , cujos componentes são:

$$(\beta^l)_j = \sum_{\mathbf{v}_k \in C_o} P^{(l)}(\bar{\mathbf{v}}_k) P^{(j)}(\bar{\mathbf{v}}_k) \quad (4.12)$$

Proof: Temos que:

$$T_{i,l} = \frac{1}{\det(\mathbf{A})} \sum_{j=1}^n \left(\det(\mathbf{A}_{[j,i]}) \sum_{\mathbf{v}_k \in C_o} P^{(l)}(\bar{\mathbf{v}}_k) P^{(j)}(\bar{\mathbf{v}}_k) \right) \quad (4.13)$$

Sustituindo com o componente $(\beta^l)_j$ e aplicando a regra deLaplace:

$$T_{i,l} = \frac{1}{\det(\mathbf{A})} \sum_{j=1}^n \det(\mathbf{A}_{[j,i]}) (\beta^l)_j = \frac{\det(\mathbf{K}_i^l)}{\det(\mathbf{A})} \quad (4.14)$$

□

Lemma 4.2. *As componentes $T_{i,l} = \delta_{i,l}$, para todo $l = 1, \dots, n$; onde $\delta_{i,l}$ é o delta de Kronecker.*

Proof: Se $l = i$ temos que $(\beta^l)_j = (\beta^i)_j = a_{i,j}$, então a matriz $\mathbf{K}_i^l = \mathbf{A}$, portanto $T_{i,l} = T_{i,i} = 1$. Quando $i \neq l$, por definição $(\beta^l)_j = a_{l,j}$, então, a matriz $\mathbf{K}_i^l$ terá duas colunas iguais, sendo seu determinante nulo. □

A partir do lema 4.2, se obtém da expressão (4.10):

$$c_i = \frac{1}{\pi(i)} [D_i f(\mathbf{r}_o)] + \sum_{l=1}^{\infty} \frac{T_{i,n+l}}{\pi(n+l)} [D_{(n+l)} f(\mathbf{r}_o)] \quad (4.15)$$

Note que o segundo termo a expressão de acima contem os elementos de ordem superior da aproximação os quais correspondem ao erro de truncamento (ET). Portanto, os coeficientes c_i dos polinômio W_o são iguais às derivadas D_i da função f em $\mathbf{v}_o$ mais um ET.

5. Consistência

O MDFG é consistente se o ET da aproximação é nulo quando a malha utilizada na discretização tende a ser continua, isto é, quando o parâmetro de densidade da malha $h \rightarrow 0$. Para isto, se deve calcular a ordem do ET dependente de h .

A matriz $\mathbf{A}$ pode ser construída como o produto $\mathbf{P}\mathbf{P}^T = \mathbf{A}$, onde $\mathbf{P}$ é uma matriz $n \times n_o$ (onde n_o é o número de vértices na célula computacional C_o associada com $\mathbf{v}_o$, e n é o número de monômios em W_o), cujos elementos são dados como $\mathbf{P}_{k,j} = P^{(j)}(\bar{\mathbf{r}}_k)$. Definindo $\tilde{r}_k^i = (r_k^i - r_o^i)/h$, e fazendo uso do lemma 3.2, obtemos:

$$\begin{aligned} P^{(j)}(\bar{\mathbf{r}}_k) &= \prod_{m=1}^d (r_k^m - r_o^m)^{\mu_m(j)} \\ &= \prod_{m=1}^d h^{\mu_m(j)} (\tilde{r}_k^m)^{\mu_m(j)} = h^{N_d(j)} u_{j,k} \end{aligned} \quad (5.1)$$

onde $u_{j,k} = (\tilde{r}_k^1)^{\mu_1(j)} \dots (\tilde{r}_k^d)^{\mu_d(j)}$. Se $\mathbf{H}$ é uma matriz diagonal com $\text{diag}(\mathbf{H}) = \{h^{N_d(1)}, \dots, h^{N_d(n)}\}$ e $\tilde{\mathbf{P}}$ é a matriz composta pelos elementos $u_{j,k}$ então $\mathbf{P} = \mathbf{H}\tilde{\mathbf{P}}$ e $\mathbf{A} = \mathbf{P}\mathbf{P}^T = (\mathbf{H}\tilde{\mathbf{P}})(\mathbf{H}\tilde{\mathbf{P}})^T = \mathbf{H}\tilde{\mathbf{P}}\tilde{\mathbf{P}}^T\mathbf{H}$, assim $\det(\mathbf{A}) = (\det(\mathbf{H}))^2 \det(\tilde{\mathbf{P}}\tilde{\mathbf{P}}^T)$. Como $\mathbf{H}$ é uma matriz diagonal, obtemos:

$$\det(\mathbf{A}) = h^\gamma \det(\tilde{\mathbf{P}}\tilde{\mathbf{P}}^T) ; \quad \text{com} \quad \gamma = 2 \sum_{j=1}^n N_d(j) \quad (5.2)$$

Um procedimento similar pode ser empregado na matriz $\mathbf{K}_i^l$ obtendo:

$$\det(\mathbf{K}_i^l) = h^{\gamma - N_d(i) + N_d(n+l)} \det(\tilde{\mathbf{P}}\mathbf{R}_i^l) \quad (5.3)$$

onde $\mathbf{R}_i^l$ é obtido da matriz $\tilde{\mathbf{P}}$ ao substituir os elementos $u_{j,i}$ da coluna i pelos elementos $u_{j,n+l}$.

Theorem 5.1. *A ordem do ET da aproximação da D_i derivada de uma função $f : \mathbb{R}^d \rightarrow \mathbb{R}$ utilizando o MDFG com um polinômio de aproximação W_o é $O(h^{N_d(n+1) - N_d(i)})$.*

Proof: Substituindo (5.2) e (5.3) na expressão da proposição 4.1, obtemos:

$$T_{i,n+l} = h^{N_d(n+l) - N_d(i)} \frac{\det(\tilde{\mathbf{P}}\mathbf{R}_i^l)}{\det(\tilde{\mathbf{P}}\tilde{\mathbf{P}}^T)} \quad (5.4)$$

portanto a aproximação (4.15) da D_i derivada da função f será dada:

$$c_i = \frac{1}{\pi(i)} [D_i f(\mathbf{r}_o)] + O(h^{N_d(n+l) - N_d(i)}) \quad (5.5)$$

□

Do teorema 5.1 pode se observar a necessidade de utilizar como polinômios de aproximação aqueles que sejam completos para garantir a consistência do MDFG.

Como exemplo, suponhamos que as derivadas $\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$, $\frac{\partial^2 f}{\partial x^2}$ e $\frac{\partial^2 f}{\partial y^2}$ da função $f : D \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ devem ser aproximadas pelo MDFG num ponto v_o o qual possui uma célula computacional hexagonal. Seja o polinômio de aproximação W_o dado por:

$$W_o = c_1 P^{(1)} + c_2 P^{(2)} + c_3 P^{(3)} + c_4 P^{(4)} + c_5 P^{(5)}$$

De (4.15), calculamos os coeficientes c_i como

$$\begin{aligned} c_1 &= D_1 f + \frac{1}{6} T_{1,6} D_6 f + \frac{1}{2} T_{1,7} D_7 f + \frac{1}{2} T_{1,8} D_8 f \\ &\quad + \frac{1}{6} e_{1,9} D_9 f + T.O.S. \\ c_2 &= D_2 f + \frac{1}{6} T_{2,6} D_6 f + \frac{1}{2} T_{2,7} D_7 f + \frac{1}{2} T_{2,8} D_8 f \\ &\quad + \frac{1}{6} T_{2,9} D_9 f + T.O.S. \\ c_3 &= \frac{1}{2} D_3 f + \frac{1}{6} T_{3,6} D_6 f + \frac{1}{2} T_{3,7} D_7 f + \frac{1}{2} T_{3,8} D_8 f \\ &\quad + \frac{1}{6} T_{3,9} D_9 f + T.O.S. \\ c_5 &= \frac{1}{2} D_5 f + \frac{1}{6} T_{5,6} D_6 f + \frac{1}{2} T_{5,7} D_7 f + \frac{1}{2} T_{5,8} D_8 f \\ &\quad + \frac{1}{6} T_{5,9} D_9 f + T.O.S. \end{aligned}$$

Calculando os elementos $T_{i,l}$ a partir de (5.4), onde h é comprimento do v_o aos nós de sua célula, obtemos:

$$\begin{array}{cccc} T_{1,6} = \frac{3}{4}h^2 & T_{1,7} = 0 & T_{1,8} = \frac{1}{4}h^2 & T_{1,9} = 0 \\ T_{2,6} = 0 & T_{2,7} = \frac{1}{4}h^2 & T_{2,8} = 0 & T_{2,9} = \frac{3}{4}h^2 \\ T_{3,6} = 0 & T_{3,7} = 0 & T_{3,8} = 0 & T_{3,9} = 0 \\ T_{5,6} = 0 & T_{5,7} = 0 & T_{5,8} = 0 & T_{5,9} = 0 \end{array}$$

Obtendo finalmente:

$$\begin{aligned} c_1 &= D_1 f + \frac{1}{8}h^2 (D_6 f + D_8 f) + O(h^3) \\ c_2 &= D_2 f + \frac{1}{8}h^2 (D_7 f + D_9 f) + O(h^3) \\ c_3 &= \frac{1}{2} D_3 f + O(h^2) \\ c_5 &= \frac{1}{2} D_5 f + O(h^2) \end{aligned}$$

Portanto, fazendo uso de uma célula hexagonal para cada vértice, o MDFG terá uma consistência de $O(h^2)$.

6. Análise das EDP em duas dimensões.

Por simplicidade vamos considerar problemas bi-dimensionais, assim, as EDP serão equações envolvendo duas variáveis independentes x e y e derivadas parciais de uma função real $f = f(x, y)$. A forma mais geral de uma EDP bi-dimensional de ordem s é:

$$\varphi \left(x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial y^2}, \dots, \frac{\partial^s f}{\partial x^s} \right) = 0$$

É bastante comum o caso de problemas práticos importantes onde a equação diferencial parcial é de ordem $s = 2$ e linear nas derivadas de ordem 2, portanto:

$$a \frac{\partial^2 f}{\partial x^2} + b \frac{\partial^2 f}{\partial x \partial y} + c \frac{\partial^2 f}{\partial y^2} = \varphi \left(x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) \quad (6.1)$$

Além disso, uma equação diferencial que representa um modelo de um problema físico, terá adicionalmente condições de fronteira e condições iniciais. Geralmente, as condições iniciais e as condições de fronteira são funções que envolvem a função f e suas derivadas de primeira ordem no domínio D e na fronteira ∂D respectivamente.

Ao solucionar numericamente uma equação como (6.1), o domínio D deve inicialmente ser representado por um conjunto de nós V , sobre tais pontos utilizamos o MDFG para discretizar as EDP, as quais são solucionadas utilizando as condições iniciais e as condições de fronteira.

Para cada nó $v_o \in V$, devemos procurar um polinômio de aproximação W_o a partir do qual, possamos obter os elementos c_i associados às derivadas D_i . Uma boa opção é utilizar um polinômio W_o que pertença ao espaço polinomial $\mathcal{P}_{s'}^2$, onde $s' \geq s = 2$. Por exemplo, considere os polinômios $W_o^2 \in \mathcal{P}_2^2$ e $W_o^3 \in \mathcal{P}_3^2$. No primeiro caso, ao utilizar W_o^2 , teremos que a ordem do erro de truncamento das aproximações das derivadas de primeira e segunda ordens são $O(h^2)$ e $O(h)$ respectivamente (quando $e_i^{(n+l)} \neq 0$), enquanto ao utilizar W_o^3 , a ordem do erro de truncamento das derivadas de primeira e segunda ordens serão de $O(h^3)$ e $O(h^2)$, respectivamente. Portanto as aproximações das derivadas utilizando o polinômio de aproximação W_o^3 são muito melhores que as obtidas pelo polinômio W_o^2 . A ordem de complexidade do MDFG é $O(n^3 n_v)$, onde n_v é o número total de nós do conjunto V , e n é a dimensão do espaço polinomial $\mathcal{P}_s^d$. Tal complexidade é fácil de ser demonstrada pois devemos solucionar para cada nó de V , o sistema linear dado por (4.1), para o qual pode ser utilizado um método como o LU que tem ordem $O(n^3)$. Assim, as complexidades do método ao utilizar os polinômios de aproximação W_o^2 (com $n = 5$) e W_o^3 (com $n = 9$) em cada um dos nós de V são $O(125 n_v)$ e $O(729 n_v)$ respectivamente; ou seja, o método terá um custo computacional quase seis vezes maior ao utilizar como polinômio de aproximação W_o^3 que ao utilizar W_o^2 . Portanto, a escolha do espaço dos polinômios de aproximação são sujeitas aos critérios de aproximação da solução contra custo computacional. Ao longo deste trabalho, utilizaremos como polinômios de aproximação, os pertencentes ao espaço $\mathcal{P}_2^2$.

7. Solução de equações elípticas bi-dimensionais com o MDFG: caso equação de Poisson

O modelo mais comum de EDPs elípticas é a equação de Poisson. Para um problema bi-dimensional ela é dada como:

$$-\nabla^2 f(\mathbf{r}) = F(\mathbf{r}), \quad \mathbf{r} \in D \quad (7.1)$$

Problemas que requerem a solução da equação de Poisson, estão sujeitas a diferentes tipos de condições de fronteira. No problema de Dirichlet, os valores

da solução da equação de Poisson nas fronteiras estão condicionadas a valores prescritos,

$$f = g(\mathbf{r}) , \quad \mathbf{r} \in \partial D \quad (7.2)$$

No problema de Neumann, a derivada normal de f é especificada sob ∂D , ou seja,

$$\frac{\partial f}{\partial n} = h(\mathbf{r}) , \quad \mathbf{r} \in \partial D \quad (7.3)$$

Utilizando as aproximações c_3 e c_5 para as derivadas de segunda ordem de f em um nó v_i , a equação (7.1) é discretizada como:

$$-2(c_3 + c_5) = F(\mathbf{r}_i) , \quad \forall v_i(\mathbf{r}_i) \in V \quad (7.4)$$

onde V é o conjunto de nós que estão discretizando o domínio D .

Cada um dos termos c_3 e c_5 pode ser expresso como uma relação linear dos valores de f no nó v_i e nos nós $v_k \in C_i$,

$$\begin{aligned} c_3 &= \bar{w}_{i,i}^3 f(\mathbf{r}_i) + \sum_{v_k \in C_i} \bar{w}_{i,k}^3 f(\mathbf{r}_k), \\ c_5 &= \bar{w}_{i,i}^5 f(\mathbf{r}_i) + \sum_{v_k \in C_i} \bar{w}_{i,k}^5 f(\mathbf{r}_k). \end{aligned} \quad (7.5)$$

Então, no nó v_i o operador $\nabla^2 f(\mathbf{r}_i)$ pode ser escrito por:

$$\begin{aligned} \nabla^2 f(\mathbf{r}_i) &= 2(\bar{w}_{i,i}^3 + \bar{w}_{i,i}^5) f(\mathbf{r}_i) \\ &\quad + \sum_{v_k \in C_i} 2(\bar{w}_{i,k}^3 + \bar{w}_{i,k}^5) f(\mathbf{r}_k) \\ &= w_{i,i} f(\mathbf{r}_i) + \sum_{v_k \in C_i} w_{i,k} f(\mathbf{r}_k) \end{aligned} \quad (7.6)$$

onde $w_{i,j} = 2(\bar{w}_{i,j}^3 + \bar{w}_{i,j}^5)$ para $j = i, k$.

Proposition 7.1. *Os termos $w_{i,i}$ e $w_{i,k}$ associados a v_i e $\forall v_k \in C_i$, cumprem a seguinte desigualdade:*

$$|w_{i,i}| \leq \sum_{v_k \in C_i} |w_{i,k}| \quad (7.7)$$

Proof: Para qualquer nó $v_i \in V$, os componentes c_3 e c_5 pertencem ao vetor solução c do sistema linear (4.1) como:

$$c = \mathbf{A}^{-1}b = (\mathbf{H}\tilde{\mathbf{P}}\tilde{\mathbf{P}}^T\mathbf{H})^{-1}b = (\mathbf{H}^{-1}\tilde{\mathbf{P}}^{-T}\tilde{\mathbf{P}}^{-1})\mathbf{H}^{-1}\mathbf{b}$$

onde o produto $\mathbf{H}^{-1}\mathbf{b}$, é um vetor, cujo j -ésimo elemento é

$$b'_j = \sum_{v_k \in C_i} (f(\mathbf{r}_k) - f(\mathbf{r}_i)) u_{j,k}$$

então, teremos que $\mathbf{c} = \mathbf{H}^{-1} \tilde{\mathbf{A}}^{-1} \mathbf{b}'$. Calculando os termos c_3 e c_5 se obtém:

$$c_3 = \sum_{v_k \in C_i} f(\mathbf{r}_k) \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{3,j} u_{j,k} \right) - f(\mathbf{r}_i) \sum_{v_k \in C_i} \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{3,j} u_{j,k} \right)$$

e

$$c_5 = \sum_{v_k \in C_i} f(\mathbf{r}_k) \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{5,j} u_{j,k} \right) - f(\mathbf{r}_i) \sum_{v_k \in C_i} \left(\frac{1}{h^2} \sum_{j=i}^n \tilde{u}_{5,j} u_{j,k} \right)$$

assim, o operador ∇^2 será escrito como:

$$\begin{aligned} \nabla^2 = 2 c_3 + 2 c_5 = & \sum_{v_k \in C_i} f(\mathbf{r}_k) \left(\frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \right) \\ & - f(\mathbf{r}_i) \sum_{v_k \in C_i} \left(\frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \right) \end{aligned}$$

Portanto, de (7.6) obtemos:

$$w_{i,i} = - \sum_{v_k \in C_i} \left(\frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \right) \quad (7.8)$$

$$w_{i,k} = \frac{2}{h^2} \sum_{j=i}^n (\tilde{u}_{3,j} + \tilde{u}_{5,j}) u_{j,k} \quad (7.9)$$

ou seja:

$$w_{i,i} = - \sum_{v_k \in C_i} w_{i,k}$$

aplicando o valor absoluto se obtém imediatamente a relação (7.7). $\square$

Assim, por exemplo, os termos $w_{i,j}$ obtidos por uma célula regular C(I) de 6 nós, utilizando um polinômio de aproximação P_2^2 , são dados por: $w_{i,k} = 2/3$, para $\forall v_k \in C_i$, e $w_{i,i} = -4$, para o nó v_i .

Sucessivamente, se calcula para cada um dos nós de V , obtendo-se um problema simultâneo de n_v equações algébricas com n_v desconhecidos valores de $f(v_i)$. Matematicamente, nosso problema (7.1) pode ser expresso como:

$$\begin{aligned}
m_{1,1} f(\mathbf{r}_1) + m_{1,2} f(\mathbf{r}_2) + \cdots + m_{1,n_v} f(\mathbf{r}_{n_v}) &= F(\mathbf{r}_1) \\
m_{2,1} f(\mathbf{r}_1) + m_{2,2} f(\mathbf{r}_2) + \cdots + m_{2,n_v} f(\mathbf{r}_{n_v}) &= F(\mathbf{r}_2) \\
&\vdots \\
m_{1,1} f(\mathbf{r}_1) + m_{1,2} f(\mathbf{r}_2) + \cdots + m_{1,n_v} f(\mathbf{r}_{n_v}) &= F(\mathbf{r}_n)
\end{aligned} \tag{7.10}$$

onde $m_{i,i} = -w_{i,i}$, $m_{i,k} = -w_{i,k}$ para $v_k \in C_i$, e os demais elementos matriciais serão nulos. Este sistema linear pode ser escrito compactamente como $\mathbf{M}\mathbf{f} = \mathbf{F}$.

No caso de um problema com condições de fronteira de Dirichlet, temos que:

$$f(\mathbf{r}_j) = g(\mathbf{r}_j), \quad v_j \in \partial D \tag{7.11}$$

portanto, podem ser eliminados de (7.10) as equações algébricas relacionadas com $v_j \in \partial D$, ficando um novo sistema linear dado como

$$\mathbf{M}'\mathbf{f}' = \mathbf{y} \tag{7.12}$$

onde $\mathbf{f}'$ é um vetor com os valores desconhecidos da função f sobre os nós do interior do domínio D , ou seja $\mathbf{f}' = \{f(\mathbf{r}_l) \mid \forall v_l \in (D - \partial D)\}$; e $\mathbf{y}$ é um vetor dado por:

$$y_l = \begin{cases} F(\mathbf{r}_l), & \text{se } v_l \text{ não é adjacente a } \partial D, \\ F(\mathbf{r}_l) + \sum_{v_k \in (C_l \cap \partial D)} w_{l,k} g(\mathbf{r}_k), & \text{caso contrário.} \end{cases} \tag{7.13}$$

Definition 7.2. A matriz $\mathbf{G} \in \mathbb{R}^n \times \mathbb{R}^n$ é monotônica se e só se

$$\mathbf{G}u \leq \mathbf{G}v \quad \Rightarrow \quad u \leq v \quad \forall u, v \in \mathbb{R}^n$$

denotando “ $\leq$ ” como:

$$u \leq v \quad \Leftrightarrow \quad u^i \leq v^i, \quad i = 1, 2, \dots, n.$$

Definition 7.3. A matriz $\mathbf{G} = (g_{i,j})$ de um sistema linear

$$\sum_{j=1}^n g_{i,j} x_j = f_i, \quad i = 1, 2, \dots, n,$$

é irredutível se e só se não existe nenhuma transformação linear $x'_i = \sum_{j=1}^n \beta_{i,j} x_j$, tal que o sistema de equações equivalente

$$\sum_{j=1}^n g'_{i,j} x'_j = f'_i, \quad i = 1, 2, \dots, n,$$

possa ser substituído por dois sistemas independentes

$$\sum_{j=1}^k g'_{i,j} x'_j = f'_i, \quad i = 1, 2, \dots, k,$$

$$\sum_{j=k+1}^n g'_{i,j} x'_j = f'_i, \quad i = k+1, \dots, n,$$

O seguinte teorema é dado para nossa exposição:

Theorem 7.4. *Se cada um dos termos $w_{i,k}$ são não negativos e a matriz $\mathbf{M}'$ é irredutível, então $\mathbf{M}'$ é monotônica.*

Proof: As seguintes condições tem que ser validadas [18]:

$$\begin{aligned} \text{(i)} \quad & m'_{i,j} \leq 0 \quad \forall i, j = 1, \dots, N, \quad i \neq j, \\ \text{(ii)} \quad & m'_{i,i} > 0 \quad \forall i = 1, \dots, N, \\ \text{(iii)} \quad & \sum_j m'_{i,j} \geq 0 \quad \forall i = 1, \dots, N, \\ \text{(iv)} \quad & \exists i \quad \sum_i m'_{i,j} > 0. \end{aligned} \tag{7.14}$$

□

No seguinte teorema é formulada uma estimacão da convergência do MDFG aplicada a problemas elípticos:

Theorem 7.5. *Se a matriz $\mathbf{M}'$ é monotônica, então o MDFG converge linearmente, isto é:*

$$\max_{v_i \in V} |f(\mathbf{r}_i) - \bar{f}(\mathbf{r}_i)| \leq Ch \tag{7.15}$$

onde C é uma constante positiva independente do parâmetro de comprimento da malha h e $\bar{f}$ denota o valor calculado da solução f .

Proof: A prova deste teorema é feita no trabalho de Demkowicz [7].

□

Portanto, para solucionar o problema (7.1) pelo MDFG, todas as células tem que cumprir as condições (7.14) para garantir a convergência.

Exemplo numérico.

Para testar o apresentado anteriormente, utilizamos uma equação de Poisson cuja solução é conhecida, e posteriormente é estudada a convergência do método. Utilizamos para isto a seguinte equação:

$$\nabla^2 f(x, y) = 6xy(x^2 + y^2 - 2), \quad (x, y) \in D = [-1, 1] \times [-1, 1]$$

com condição de Dirichlet na fronteira:

$$f(x, y) = 0, \quad (x, y) \in \partial D$$

cujas soluções analíticas são:

$$f(x, y) = xy(x^2 - 1)(y^2 - 1)$$

Para solucionar o sistema linear (7.12) é utilizado o método iterativo Gauss-Seidel. Com uma condição de suficiência para garantirmos que o método Gauss-Seidel é convergente, exigimos que o sistema seja irredutível e que os elementos matriciais $m'_{i,j}$ da matriz $\mathbf{M}'$ cumpram com:

$$|m'_{i,i}| \geq \sum_{j=1, j \neq i}^n |m'_{i,j}| \quad (7.16)$$

para todos os i , e que para pelo menos um i ,

$$|m'_{i,i}| > \sum_{j=1, j \neq i}^n |m'_{i,j}| \quad (7.17)$$

Então, se pudermos garantir que para todos os nós $v_i \in V$, os termos $a_{i,k}$ dados em (7.9), são não negativos, ou seja, ocorre a igualdade na equação (7.7), a relação (7.16) é cumprida, e para qualquer i -ésima linha associada à solução num nó v_i adjacente à fronteira ∂D , cumprirá imediatamente (7.17).

São utilizadas discretizações do domínio criadas a partir de malhas geradas pelo algoritmo de Chew [5], com diferentes valores de h ; garantindo que as células cumpram com a não negatividade dos termos $a_{i,k}$. Na figura 2 é apresentado o erro, o qual é calculado utilizando a métrica d_2 . A convergência neste exemplo concorda com o resultado teórico dado no teorema 7.5.

8. Conclusões.

Neste trabalho foi estudado um Método de Diferenças Finitas Generalizadas (MDFG) o qual é classificado dentre o grupo de métodos meshless, já que permite calcular as derivadas de uma função f numa coleção de nós espalhados sob um domínio. É demonstrado que as aproximações das derivadas calculadas pelo MDFG são consistentes quando é utilizado um polinômio de aproximação completo (ou seja, é formado por todos os monômios base de grau igual ou menor ao grau do polinômio). Assim, quando é utilizado um polinômio não completo, as aproximações das derivadas podem ser inconsistentes.

Um critério eficiente de seleção dos nós de uma célula de um nó é proposto, baseado nas relações implícitas de conectividade que possui uma malha utilizada como suporte dos nós no domínio. Aplicando algoritmos eficientes de geração de malhas, pode-se garantir uma boa distribuição dos nós, permitindo reduzir os erros da aproximação. A geometria de células bi-dimensionais não afetam significativamente o ET, quando a célula possui mais de n nós (onde n é a quantidade de monômios base que compõem o polinômio de aproximação) e a distribuição desses nós tem baixa irregularidade. O MDFG aplicado na solução de problemas elípticos com condições de fronteira de Dirichlet são convergentes.

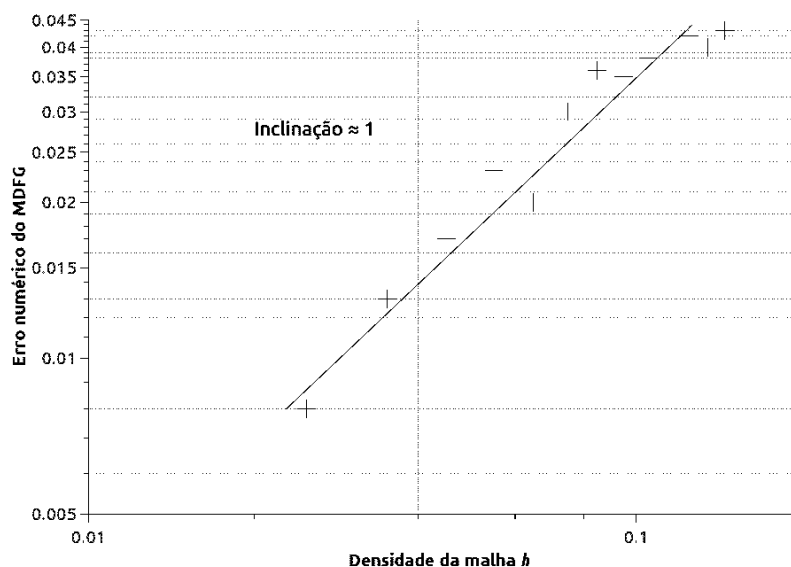


Figure 2: Resultado do teste de convergência do MDFG num problema elíptico.

Agradecimientos.

Agradeço a Luis Gustavo Nonato do *ICMC-UPS* e Norberto Mangiavacchi do *UERJ*, pelo apoio no desenvolvimento deste trabalho. O presente estudo teve o apoio financeiro do CNPq.

References

1. Atluri, S.N., Zhu, T., *A new meshless local Petrov-Galerkin (MLPG) approach*, Comput. Mech. 22, 117-127, (1998).
2. Babuška, I., Melenk, J.M., *The partition of unity method*, Int. J. Numer. Meth. Eng. 40, 727-758, (1997).
3. Belytschko, T., Krongauz, Y., Fleming, D., Krysl, P., *Meshless methods: An Overview and Recent Developments*, Comput. Meth. Appl. Mech. Engrg. 139, 2-47, (1996).
4. Benito, J.J., Ureña, F., Gavete, L., *Influence of Several Factors in the Generalized Finite Difference Method*, Applied Mathematical Modelling 25, 1039-1053, (2001).
5. Chew, L.P., *Constrained Delaunay Triangulations*, Algorithmica 4:(1), 97-108, (1989).
6. Duarte, C.A.M., Oden, J.T., *h-p Cloud - An h-p meshless method*, Numer. Meth. Partial Diff. Eqs. 12, 673-705, (1996).
7. Demkowicz, L., Karafiat, A., Liszka, T., *On some convergence results for FDM with irregular mesh*, Computer Methods in Applied Mechanics and Engineering 42, 343-355, (1984).
8. Gingold, R.A., Monaghan, J.J., *Smoothed particle hydrodynamics: theory and application to non-spherical stars*, Mon. Not. Roy. Astron. Soc. 181, 375-389, (1977).

9. Gossler, A., *Moving Least-Squares: A Numerical Differentiation Method for Irregularly Spaced Calculation Points*, Sandia technical Report, SAND2001-1669, (2001).
10. Jensen, P.S., *Finite difference techniques for variable grids*, Comp. Structures 2, 17-29, (1972).
11. Krysl, P., Belytschko, T., *An efficient linear-precision partition of unity basis for unstructured meshless methods*, Commun. Num. Meth. Eng. 16(4), 239-255, (1999).
12. Liszka, T., Orkisz, J., *The finite difference method at arbitrary irregular grids and its application in applied mechanics*, Computers and Structures 11, 83-95, (1980).
13. Liu, W.K., Jun, S., Zhang, Y.F., *Reproducing kernel particle methods*, Int. J. Numer. Meth. Fluids 20, 1081-1106, (1995).
14. Luo, Y., Häussler, U., *A generalized finite-difference method based on minimizing global residual*, Comput. Methods Appl. Mech. Engrn. 191, 1421-1438, (2002).
15. Marshall, J.S., Grand, J.R., *A Lagrangian Vorticity Collocation Method for Viscous, Axisymmetric Flows With and Without Swirl*, J. Comput Phys. 138, 302-330, (1997).
16. Monaghan, J.J., *An introduction to SPH*, Comput. Phys. Comm. 48, 89-96, (1988).
17. Nayroles, B., Touzot, G., Villon, P., *Generalizing the finite element method: diffuse approximation and diffuse elements*, Comput. Mech. 10, 307-318, (1992).
18. Nguyen, H., Reynen, J., *A space-time finite element method for hyperbolic equation*, Proc. 3rd Internat. Symp. on Numerical Methods in Engineering, Paris, (1983).
19. Perrone, N., Kao, R., *A general finite difference method for arbitrary mesh*, Computers and Structures 5, 647-664, (1975).
20. Ureña F., Benito, J.J., Gavete, L., *Application of the generalized finite difference method to solve the advection-diffusion equation*, Journal of Computational and Applied Mathematics 235, 1849-1855, (2011).
21. Shepard, D., *A two dimensional function for irregularly spaced data*, ACM National Conference, (1968).

Daniel R. Izquierdo
Facultad de Ciencias,
Universidad de Ciencias Aplicadas y Ambientales U.D.C.A,
Bogotá, Colombia,
orcid.org/0000-0001-5847-2252.
E-mail address: dizquierdo@udca.edu.co