Universidade Federal do Espírito Santo
Departamento de Informática

André Barreto

Método das Diferenças Finitas Aplicado a Problemas Bidimensionais

Trabalho 1 de Algoritmos Numéricos 2

Vitória
2016

1 Introdução

O estudo da equação de transporte, também denominada equação da advecção-difusão-reação, continua sendo um ativo campo de pesquisa, uma vez que essa equação é de fundamental importância nos problemas relacionados a aerodinâmica, meteorologia, oceanografia, hidrologia, engenharia química e de reservatórios. A equação de transporte tem características bastante peculiares que fazem com que sua resolução por meios numéricos seja dificultada em situações onde o problema é fortemente convectivo. Por isso, diversos métodos têm sido desenvolvidos e aplicados, com a intenção de superar as dificuldades numéricas impostas por esta equação.

A equação de transporte bidimensional pode ser definida por:

$$-k\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+\beta_x(x,y)\frac{\partial u}{\partial x}+\beta_y(x,y)\frac{\partial u}{\partial y}+\gamma(x,y)u=f(x,y) \text{ em } \Omega \quad (1)$$

Este trabalho tem como objetivo verificar como as formas de armazenamento das estruturas resultantes pela discretização da equação (1) por diferenças finitas pode impactar no tempo de processamento.

Para isto, será implementado em linguagem C um programa capaz de aplicar a discretização da equação de transporte e resolver o sistema resultante discreto. Quanto ao método de resolução, será aplicado o algoritmo SOR (*Sucessive Over Relaxation*) de duas formas: resolvendo o sistema penta-diagonal armazenado de forma esparsa, utilizando cinto vetores; e completamente livre de matriz.

Nas próximas seções estão descritos em detalhes o desenvolvimento, testes e conceitos do trabalho, sendo que o próximo tópico faz um resumo do método das Diferenças Finitas, descrevendo as técnicas e ordem de aproximação usadas.

2 Método das Diferenças Finitas

Método das Diferenças Finitas é um método numérico para resolver equações diferenciais através de aproximações de derivadas por diferenças finitas. Estas aproximações são obtidas pela série de Taylor:

$$f(x+h)=f(x)+f'(x)h+\frac{f''(x)h^2}{2}+\frac{f'''(x)h^3}{6}+o(h^4)$$
$$f(x-h)=f(x)-f'(x)h+\frac{f''(x)h^2}{2}-\frac{f'''(x)h^3}{6}+o(h^4)$$

Portanto, a primeira e segunda derivadas podem ser escritas das seguintes formas:

$$f'(x) = \frac{f(x+h) - f(x)}{h} + o(h) \text{ , diferença atrasada (1a ordem)}$$
$$f'(x) = \frac{f(x) - f(x-h)}{h} + o(h) \text{ , diferença adiantada (1a ordem)}$$
$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + o(h^2) \text{ , diferença central (2a ordem)}$$
$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + o(h^2)$$

O método das Diferenças Finitas em domínios retangulares consiste em **três** etapas principais:

- **Discretização do domínio**, onde os pontos (x, y) são traduzidos em um domínio linear, de forma que $I = (j-1)n + i$ e $1 \leq I \geq n * m$, onde i e j são representam um ponto bidimensional e I uma posição representativa deste ponto.
- **Aproximar a Equação (1) por diferenças finitas**, que consiste em reescrever a equação utilizando as aproximações por diferenças finitas, gerando um sistema linear.
- **Aplicar condições de contorno** que são definidas pelo problema proposto e que impactam diretamente na solução do problema.

3 Implementação

Nesta seção estão apresentados os códigos em C relevantes do sistema que mostra suas principais funcionalidades necessárias para a resolução do Método das Diferenças Finitas em 2D.

4 Experimentos Numéricos

Para verificar a qualidade dos métodos implementados, foram realizados uma séries de testes a um conjunto de experimentos propostos. Em cada experimento, o programa em questão é aplicado em diferentes ordens de sistema e em ambas as versões do SOR apresentadas, onde o foco é a comparação dos tempos de execução em cada caso.

Para isto, em cada experimento a seguir será ilustrado uma tabela com os casos relevantes de teste e o tempo de execução em segundos que cada computação levou. Os tempos de execução apresentados foram recolhidos utilizando o utilitário *time* do Linux que mede o tempo que o processo ficou rodando.

Para simplificar, algumas abreviações foram utilizadas nestas tabelas, a saber:

- n: partições no eixo X;
- m: partições no eixo Y;
- SOR "normal": algoritmo SOR utilizando a estrutura de armazenado da matriz pentadiagonal;
- SOR "livre": algoritmo SOR livre de matriz.

Os testes nesta seção foram executados em uma máquina com as seguintes configurações:

Ubuntu 14.04 64-bit
Intel Core i7-3770 CPU @ 3.40GHz x 4
8GB de memória

Quanto aos parâmetros do algoritmo SOR, os testes foram executados com os seguintes:

$$\begin{aligned} \omega &= 1.6 \\ toler &= 0.00001 \\ iterMax &= 1000000 \end{aligned}$$

onde ω é o coeficiente de relaxação, $toler$ é a tolerância do erro de aproximação e $iterMax$ o número máximo de iterações.

4.1 Validação 1 - Problema simples com solução trivial

Este é um experimento simples para testes do sistema onde deve-se determinar a distribuição de calor em uma chapa de metal, com faces termicamente isoladas e com espessura desprezível, sendo que a temperatura é conhecida em todas as faces da chapa. Neste caso, a equação (1) é dada por:

$$-\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = 0 \text{ em } \Omega \tag{2}$$

Sendo T_0 a temperatura nas faces, espera-se que os valores no interior da placa sejam igualmente T_0 em todos os pontos da discretização.

Veja na tabela 1 os tempos de execução deste experimento por valores n e m e versão do algoritmo SOR.

n	**m**	**SOR**	**Tempo**
25	100	normal livre	0m0.034s 0m0.088s
100	100	normal livre	0m0.220s 0m0.526s
500	1000	normal livre	4m16.887s 10m2.833s
10000	1000	normal livre	OOOOOOO OOOOOOO

Tabela 1: Testes - Validação 1

4.2 Validação 2 - Problema com solução conhecida

Neste experimento, deve-se determinar a solução aproximada para $u(x, y)$ em $\Omega = (0, 1) \times (0, 1)$ considerando na Eq. (1):

$$\begin{aligned} k &= 1 \\ \beta_x(x, y) &= 1 \\ \beta_y(x, y) &= 20y \\ \gamma(x, y) &= 1 \\ f(x, y) \text{ tal que } u(x, y) &= 10xy(1-x)(1-y)e^{x^{4.5}} \\ &\text{é a solução exata} \end{aligned} \tag{3}$$

e sabendo que $u(x, y) = 0$ no contorno de Ω.

Veja na tabela 2 os tempos de execução deste experimento por valores n e m e versão do algoritmo SOR.

4.3 Aplicação Física 1 - Resfriador bidimensional

Este experimento consiste em uma aplicação dos métodos em questão para resfriar uma massa aquecida. Exemplos podem incluir o resfriamento de chips de computadores ou amplificadores elétricos. O modelo matemático que descreve a transferência de calor nas direções x e y é dado por:

$$-k\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + \frac{2c}{T}u = \frac{2c}{T}u_{ref} = 0 \text{ em } \Omega = (0, L) \times (0, W) \tag{4}$$

n	m	SOR	Tempo
25	100	normal livre	0m0.041s 0m1.695s
100	100	normal livre	0m0.162s 0m9.440s
500	1000	normal livre	3m34.692s OOOOOOO
10000	1000	normal livre	OOOOOOO OOOOOOO

Tabela 2: Testes - Validação 2

onde k é a condutividade térmica constante, c é o coeficiente de transferência de calor, T é a altura do resfriador e u_{ref} é a temperatura de referência. Deve-se encontrar a temperatura no interior do resfriador considerando as seguintes condições de contorno:

$$\begin{aligned} u(x,0) &= 70 \\ u(x,W) &= 70 \\ u(0,y) &= 200 \\ k\frac{\partial u}{\partial n}(L,y) &= c(u_{ref} - u(L,y)) \end{aligned}$$

Veja na tabela 3 os tempos de execução deste experimento por valores n e m e versão do algoritmo SOR.

n	m	SOR	Tempo
25	100	normal livre	0m0.041s 0m0.095s
100	100	normal livre	0m0.240s 0m0.580s
500	1000	normal livre	2m52.629s 6m55.552s
10000	1000	normal livre	OOOOOOO OOOOOOO

Tabela 3: Testes - Aplicação Física 1

4.4 Aplicação Física 2 - Escoamento em Águas Subterrâneas

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Veja na tabela 4 os tempos de execução deste experimento por valores n e m e versão do algoritmo SOR.

n	**m**	**SOR**	**Tempo**
25	100	normal	OOOOOOO
		livre	OOOOOOO
100	100	normal	OOOOOOO
		livre	OOOOOOO
500	1000	normal	OOOOOOO
		livre	OOOOOOO
10000	1000	normal	OOOOOOO
		livre	OOOOOOO

Tabela 4: Testes - Aplicação Física 2

5 Conclusão

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.