Um Estudo da Instabilidade de Saffman-Taylor com Fluido Magnético e, ou Anisotrópico

Ataias Pereira Reis, Yuri Dumaresq Sobral, Francisco Ricardo da Cunha
Universidade de Brasília
Departamento de Matemática
Brasília, Brasil
ataiasreis@gmail.com

***Abstract*—Criação de algoritmos para resolver equações diferenciais parciais. Uso de diferenças finitas para resolver as equações de Laplace e Poisson, usando método ímplicito e explícito, usando bibliotecas de álgebra linear de código aberto. Análise da diferença de tempo de resolução entre os métodos implícito e explícito, e no caso do método explícito. Uso do método iterativo e de resolução direta de sistema linear.**

***Index Terms*—Ferrofluidos, Navier Stokes, EDP, Poisson, Laplace, Diferenças finitas, Eigen**

I. Introdução

A instabilidade de Saffman-Taylor, o endedamento, ou *fingering*, ocorre na superfície de contato entre dois fluidos. Ocorre quando um fluido menos viscoso é injetado para deslocar um outro mais viscoso (na situação inversa, do fluido mais viscoso usado para movimentar o outro, a interface é estavel, não ocorre o endedamento). Também pode ocorrer movida pela gravidade, ao invés de injeção de um fluido em outro. Neste caso, se a interface separando os fluidos de diferentes densidades está direcionada na horizontal, e o mais pesado em cima do outro. Este tipo de problema ocorre com a injeção de água em tubos de petrolíferas marítimas, para fazer o óleo subir. Este tipo de efeito é negativo na extração do petróleo, pois ocorrem bolhas de um líquido no outro que causam problemas para se obter somente o óleo.

Para o estudo dessa instabilidade, que não é nada trivial, precisa-se de ferramentas númericas. A equação de Navier Stokes deve ser discretizada e então resolvida numericamente. Ela é uma equação diferencial parcial altamente não-linear e de difícil resolução. No caso da instabilidade de Saffman-Taylor, a fronteira se movimenta, que é a interface entre os dois fluidos, então também precisa-se de um método que lide bem com isso.

Tais ferramentas numéricas e decisão de métodos/algoritmos a se utilizar são a primeira etapa neste projeto. A apresentação no resto deste relatório mostrará até onde se alcançou na codificação dos algoritmos para resolução do problema proposto, que inicia-se com o estudo de equações diferenciais parciais e de diferenças finitas, o método utilizado para discretização, e o uso em partes da equações de Navier Stokes até chegar em sua totalidade.

II. Metodologia

A metologia aqui proposta tenta-se seguir uma ordem mais ou menos cronológica das tarefas e do que estava sendo utilizado ou não.

1) Equação de Laplace
2) Ferramenta
3) Método Explícito
4) Método Implícito
5) git e python
6) Navier Stokes

A. Equação de Laplace

1) Forma contínua: Primeiramente, teve-se a familiarização do aluno com a resolução de equações diferenciais parciais, realizando um curso na UnB, onde foram vistas principalmente a equação de Laplace, Poisson, Calor e Onda. O método de resolução aprendido foi por separação de variáveis, o método de Fourier. Após isso, o primeiro problema proposto foi resolver uma destas equações numericamente, a escolhida foi a equação de Laplace em duas dimensões, que segue abaixo:

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \tag{1}$$

A primeira etapa na resolução desta equação de maneira numérica é discretizá-la, e isto quer dizer limitar o domínio, escolher um número de pontos em cada dimensão, e representar a equação como operações mais simples que o computador possa entender, diferentes da derivada. O escolhido foi utilizar as diferenças finitas.

2) Forma discreta - explícito: Para a discretização da equação 1, usa-se o método mencionado antes, que são as diferenças finitas. Este método basicamente faz a seguinte aproximação para a derivada:

$$f'(x) = \lim_{h\to 0} \frac{f(x+h)-f(x)}{h} = \frac{d}{dx}f(x) \approx \frac{f_i - f_{i-1}}{\Delta x} \tag{2}$$

Os valores de f_i são valores discretizados da função $f(x)$, ou seja, $f_i = f(i\Delta x)$. Agora a distância entre dois pontos passa a ser de Δx, e daí a derivada se calcula como a diferença entre um ponto e outro, dividida pela distância entre eles. Há vários tipos de diferenças: progressivas, regressivas, centrais e podem ter mais pontos, e diferentes ordens de erro podem ser obtidas dependendo dos valores de Δx e do número de pontos usados nos cálculos. Para Laplace, utilizou-se diferenças centrais de segunda ordem. Neste caso, o erro é da ordem de Δx^2.

$$f''(x) \approx \frac{f(x+\Delta x) - 2f(x) - f(x-\Delta x)}{\Delta x^2} \tag{3}$$

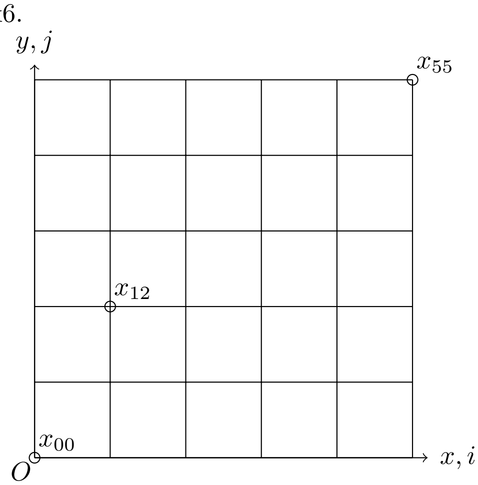
Utilizando essa equação para laplace, chegamos na seguinte discretização, considerando $\Delta x = \Delta y$ e que a variável x varia com pontos indicados pelo indice i e a variável y com o índice j:

$$\frac{u_{i+1j} + u_{i-1j} + u_{ij+1} + u_{ij-1} - 4u_{ij}}{\Delta x^2} = 0 \tag{4}$$

Reordenando essa equação:

$$u_{ij} = \frac{u_{i+1j} + u_{i-1j} + u_{ij+1} + u_{ij-1}}{4} \tag{5}$$

3) Domínio: A equação da fórmula 5 está na forma explícita, na qual um ponto se obtém a partir de outros pontos, mas como quando se muda o valor de um ponto, ele influencia no cálculo de outros, esta fórmula deve ser aplicada muitas vezes, até se alcançar a convergência em todos os pontos do domínio. Entretanto, se apenas aplica-se essa equação a todos os pontos do domínio, já se vê um problema, pois ela sempre pede mais pontos para os lados. Por isso, além da equação, necessita-se das condições de contorno para o problema, onde-se são determinados os valores nas fronteiras do domínio. O domínio escolhido é um quadrado, e abaixo tem-se o exemplo de uma malha 6x6.



III. Resultados

A. Programas

B. *Acessar dois Kinects*

IV. Conclusão

Agradecimentos

The authors would like to thank...

References

[1] Getting Started with Ubuntu 10.10
[2] OpenKinect