

# PME 2556 – Dinâmica dos Fluidos Computacional

Aula 5 – Solução da Equação da Quantidade de Movimento

## 5.1 Equação da Quantidade de Movimento

Da equação geral de transporte:

$$\int_S \rho u_j \phi n_j dA = \int_S \Gamma \frac{\partial \phi}{\partial x_j} n_j dA + \int_{\nabla} S_\phi dV$$

Para a Quantidade de Movimento a grandeza  $\phi$  é a velocidade  $u_i$ , e, em geral, se o escoamento for incompressível e as propriedades forem constantes e desprezarmos forças de campo, o termo fonte é dada pelo gradiente de pressão:

$$\int_S u_i \rho u_j n_j dA = \int_S \mu \frac{\partial u_i}{\partial x_j} n_j dA - \int_{\nabla} \frac{\partial p}{\partial x_i} dV$$

## 5.1 Equação da Quantidade de Movimento

Integrando sobre um volume, obtemos a equação algébrica:

$$a_P u_{iP} = \sum a_{viz} u_{i viz} + b$$

Onde os coeficientes  $a_{viz}$  contém um termo de fluxo de massa, ou seja, os coeficientes da matriz do sistema linear são função da própria solução. Isso traz dificuldade para a convergência, exigindo a chamada subrelaxação.

## 5.2 Subrelaxação

Para que, ao longo do processo iterativo, os coeficientes do sistema linear variem lentamente de forma a conseguirmos convergência, a taxa de variação da própria solução é intencionalmente diminuída:

$$\phi_P^{\text{iteração } n} = \phi_P^{\text{iteração } n-1} + \alpha_\phi \left( \phi_P^{\text{calculado}} - \phi_P^{\text{iteração } n-1} \right), \quad \alpha_\phi \leq 1$$

Que fica:

$$\phi_P^{\text{iteração } n} = \alpha_\phi \phi_P^{\text{calculado}} + (1 - \alpha_\phi) \phi_P^{\text{iteração } n-1}, \quad \alpha_\phi \leq 1$$

## 5.2 Subrelaxação

Em geral, a forma mais eficiente de implementar a subrelaxação é através de uma mudança nos coeficientes e carregamentos do sistema linear original:

$$\phi_P^{iteração\ n} = \frac{\sum \alpha_\phi a_{viz} \phi_{viz} + \alpha_\phi b + a_P (1 - \alpha_\phi) \phi_P^{iteração\ n-1}}{a_P}$$

Basta fazer:

$$b_{com\ subrelaxação} = \alpha_\phi b_{sem\ subrelaxação} + a_P (1 - \alpha_\phi) \phi_P^{iteração\ n-1}$$

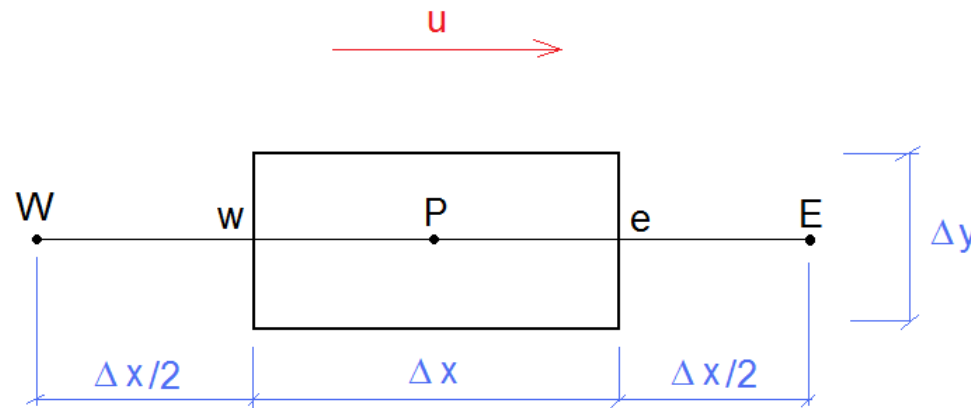
$$a_{viz,com\ subrelaxação} = \alpha_\phi a_{viz,sem\ subrelaxação}$$

Recuperando então a forma :

$$a_P \phi_P = \sum a_{viz,com\ subrelaxação} \phi_{viz} + b_{com\ subrelaxação}$$

## 5.3 Acoplamento pressão-velocidade

Para um problema unidimensional:



$$a_P u_{iP} = \sum_{\text{vizinhos}} a_{\text{viz}} u_{\text{viz}} + b \quad \text{Fica:}$$

$$a_P u_P = \sum_{\text{vizinhos}} a_{\text{viz}} u_{\text{viz}} - \frac{\partial p}{\partial x} \Delta x \Delta y \quad \text{Ou:} \quad a_P u_P = \sum_{\text{vizinhos}} a_{\text{viz}} u_{\text{viz}} - (p_e - p_w) \Delta y$$

## 5.3 Acoplamento pressão-velocidade

Interpolando linearmente para obter as pressões nas faces:

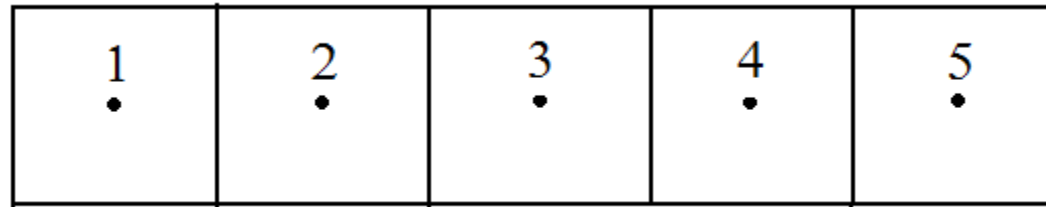
$$a_P u_P = \sum_{vizinhos} a_{viz} u_{viz} - \left( \frac{p_E + p_P}{2} - \frac{p_W + p_P}{2} \right) \Delta y$$

Que resulta:

$$a_P u_P = \sum_{vizinhos} a_{viz} u_{viz} - (p_E - p_W) \frac{\Delta y}{2}$$

## 5.3 Acoplamento pressão-velocidade

Logo, temos:



$$a_2 u_2 = \sum_{\text{vizinhos}} a_{\text{viz}} u_{\text{viz}} - (p_3 - p_1) \frac{\Delta y}{2}$$

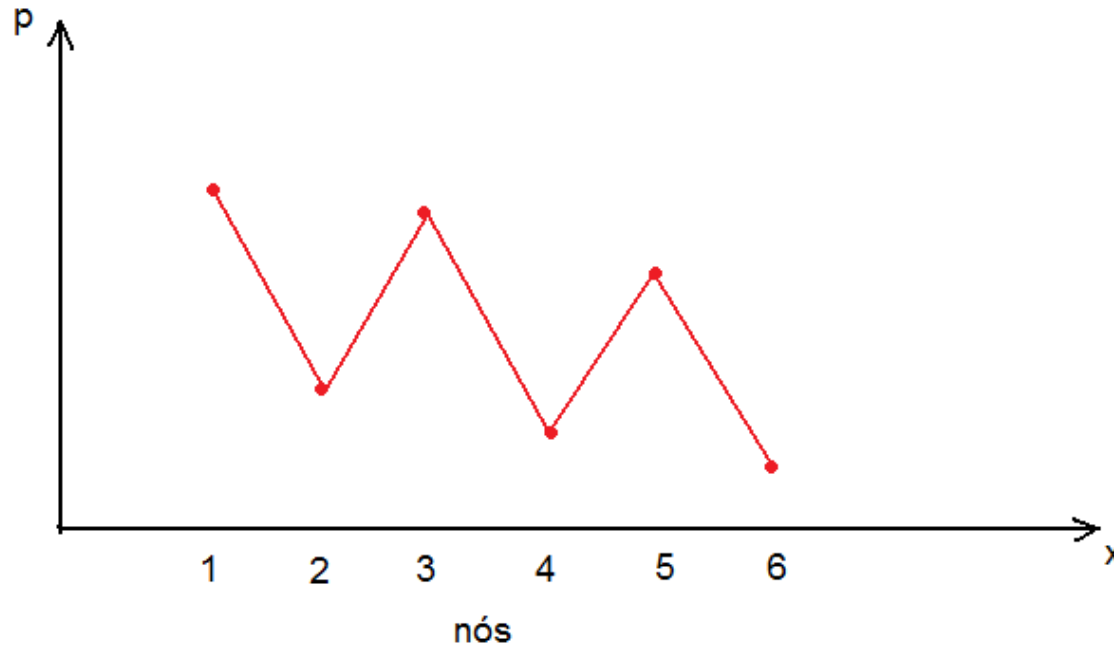
$$a_3 u_3 = \sum_{\text{vizinhos}} a_{\text{viz}} u_{\text{viz}} - (p_4 - p_2) \frac{\Delta y}{2}$$

$$a_4 u_4 = \sum_{\text{vizinhos}} a_{\text{viz}} u_{\text{viz}} - (p_5 - p_3) \frac{\Delta y}{2}$$



## 5.3 Acoplamento pressão-velocidade

Note que a velocidade em 2 depende das pressões em 1 e 3, que a velocidade em 3 depende das pressões em 2 e 4 e que a velocidade em 4 depende das pressões em 3 e 5. Não há nenhuma relação entre as pressões em nós adjacentes. Isso faz aparecer um problema chamado “desacoplamento par-ímpar” (*odd-even decoupling* ou, em 2D, *checkerboard pressure field*):



## 5.3 Acoplamento pressão-velocidade

Soluções para o desacoplamento par-ímpar:

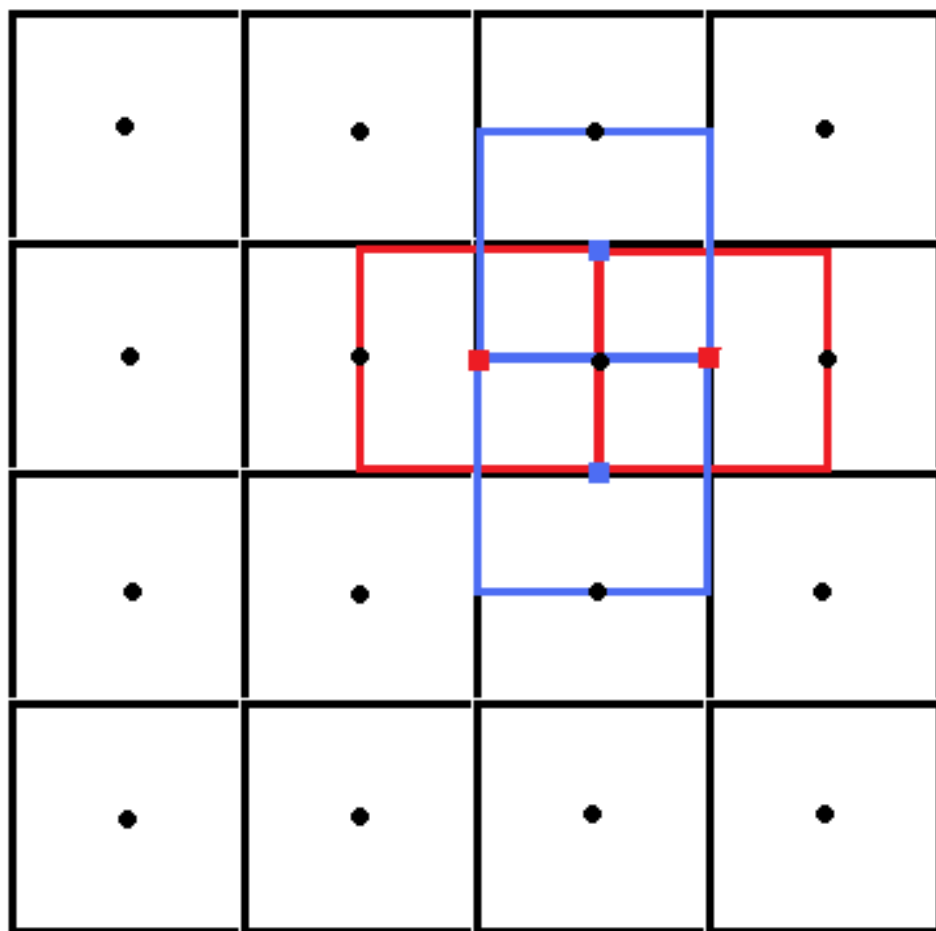
- a) Malha deslocada (staggered grid)
- b) Interpolação de Rhie&Chow

## 5.3 Acoplamento pressão-velocidade

Malha deslocada: Ao invés de um arranjo co-aloado (*colocated arrangement*), usa-se um arranjo pelo qual as pressões e componentes da velocidade são dispostas em volumes diferentes, de tal forma que a pressão esteja sempre disponível na face das células usadas para calcular a velocidade, sem necessidade de interpolação.

## 5.3 Acoplamento pressão-velocidade

Ex: Malha deslocada



- nós para pressão  $p$
- nós para velocidade  $u$
- nós para velocidade  $v$

## 5.3 Acoplamento pressão-velocidade

Interpolação de Rhie-Chow: Usa-se um arranjo co-allocado, mas as velocidades são calculadas nas faces dos volumes de controle usando uma interpolação. As velocidades no volume são dadas por:

$$u_P = \underbrace{\frac{\sum a_{viz} u_{viz}}{a_P}}_{\hat{u}_P} - (p_e - p_w) \underbrace{\frac{\Delta y}{a_P}}_{d_P}$$

As velocidades nas faces “w” e “e” da célula “P” são dadas, por exemplo, por interpolações do tipo:

$$u_e = \frac{\hat{u}_P + \hat{u}_E}{2} - \frac{d_P + d_E}{2} (p_E - p_P)$$

$$u_w = \frac{\hat{u}_P + \hat{u}_W}{2} - \frac{d_P + d_W}{2} (p_P - p_W)$$

Note que agora, a pressão em “P” está relacionada com as pressões em volumes adjacentes.

## 5.4 Algoritmo SIMPLE

Sobra uma dificuldade para fazermos a solução de escoamentos incompressíveis: é preciso acoplar velocidades e pressões sem uma equação de estado. Isso pode ser feito através de um algoritmo que, avaliando um balanço de massa na célula, incrementa a pressão caso esse balanço seja negativo ou decrementa a pressão caso esse balanço seja positivo.

## 5.4 Algoritmo SIMPLE

### Semi Implicit Method for Pressure Linked Equations - SIMPLE

Passo 1: Calcula as velocidades com o campo de pressões existente:

$$u_P^* = \frac{\sum a_{viz} u_{viz}^*}{a_P} - d_P (p_e^* - p_w^*)$$

Esse campo de velocidades em geral não satisfaz a continuidade.

## 5.4 Algoritmo SIMPLE

Passo 2: Monta-se uma equação de correção da pressão. Imagine que  $u'$  e  $p'$  sejam correções necessárias para que o campo de velocidades satisfaça a continuidade:

$$u'_P = \frac{\sum a_{viz} u'_{viz}}{a_P} - d_P (p'_e - p'_w)$$

Por simplicidade, fazemos apenas:

$$u'_P = d_P (p'_w - p'_e)$$



## 5.4 Algoritmo SIMPLE

O fluxo de massa numa célula (unidimensional) é dado por:

$$\rho u_e^* \Delta y + \rho u'_e \Delta y - \rho u_w^* \Delta y - \rho u'_w \Delta y = 0$$

Podemos assim escrever um erro de conservação da massa dado por:

$$\rho u'_e \Delta y - \rho u'_w \Delta y = \rho u_w^* \Delta y - \rho u_e^* \Delta y = -\dot{m}^*$$

Substituindo a equação que relaciona correções de velocidade e pressão:

$$\rho d_e (p'_P - p'_E) \Delta y - \rho d_w (p'_W - p'_P) \Delta y = -\dot{m}^*$$

Forma-se uma equação algébrica para a correção de pressão:

$$a_P p'_P = \sum a_{viz} p'_{viz} + b$$

## 5.4 Algoritmo SIMPLE

Passo 3: resolve-se a equação de pressão

Passo 4: corrigem-se velocidades e pressões:

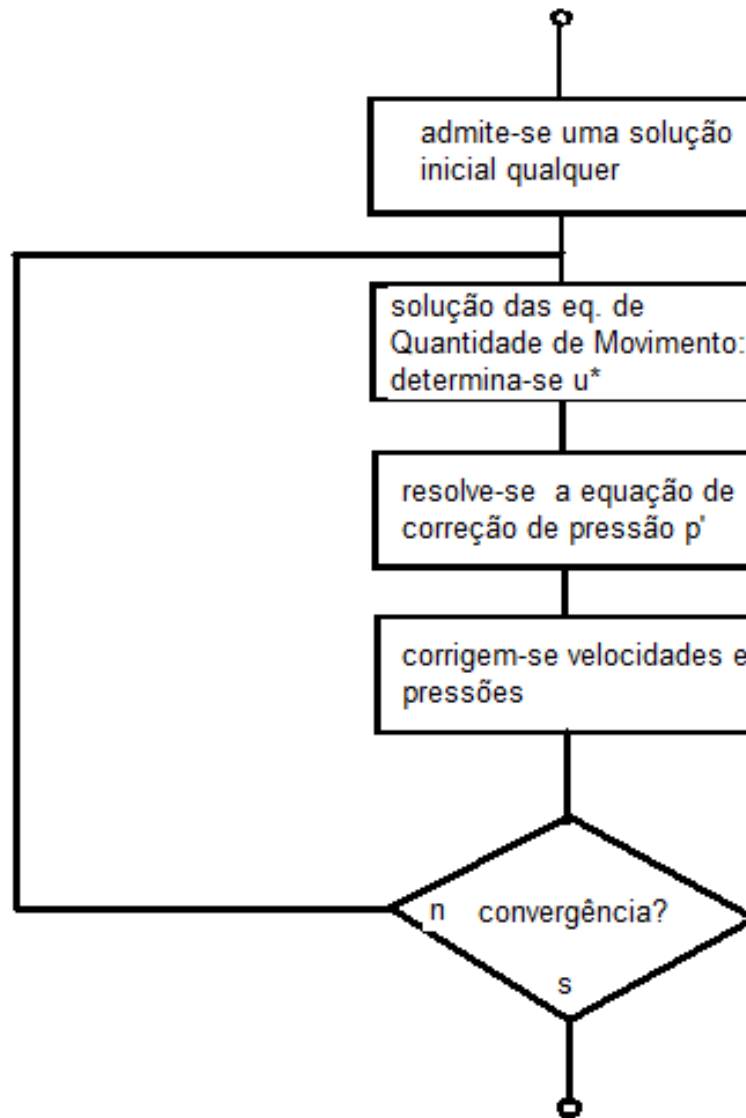
$$u_P = u_P^* + u'_P = u_P^* + d_P (p'_w - p'_e)$$

$$p_P = p_P^* + p'_P$$

## 5.4 Algoritmo SIMPLE

- Subrelaxação é usada na pressão para obter convergência, justamente por causa do termo desprezado ao relacionar correções de pressão com correções de velocidade. Portanto, temos:  
 $p = p^* + \alpha_p p'$ , com  $\alpha_p$  por volta de 0,3. A subrelaxação usada na equação de quantidade de movimento costuma ser maior, com  $\alpha_u$  por volta de 0,7.
- O processo é iterativo, com um algoritmo que pode ser descrito pelos blocos:

## 5.4 Algoritmo SIMPLE



## 5.5 SIMPLEC

No algoritmo SIMPLE, a equação que relaciona correções de velocidade e pressão é obtida fazendo a hipótese que:

$$\frac{\sum a_{viz} u'_{viz}}{a_P} \approx 0$$

De modo que:

$$u'_P = \frac{\sum a_{viz} u'_{viz}}{a_P} - d_P(p'_e - p'_w) \Rightarrow u'_P = d_P(p'_w - p'_e)$$

## 5.5 SIMPLEC

No algoritmos SIMPLEC, fazemos a seguinte subtração dos dois lados da equação:

$$u'_P - \frac{\sum a_{viz} u'_P}{a_P} = \frac{\sum a_{viz} u'_{viz}}{a_P} - \frac{\sum a_{viz} u'_P}{a_P} - d_P(p'_e - p'_w)$$

Isso resulta:

$$u'_P \left( 1 - \frac{\sum a_{viz}}{a_P} \right) = \frac{\sum a_{viz} (u'_{viz} - u'_P)}{a_P} - d_P(p'_e - p'_w)$$

Que é simplificado para:

$$u'_P \left( 1 - \frac{\sum a_{viz}}{a_P} \right) = d_P(p'_w - p'_e)$$

## 5.5 SIMPLEC

Podemos escrever a equação:

$$u'_P \left( 1 - \frac{\sum a_{viz}}{a_P} \right) = d_P (p'_w - p'_e)$$

Como:

$$u'_P a_{P_{SIMPLEC}} = d_P (p'_w - p'_e)$$

Onde:

$$a_{P_{SIMPLEC}} = 1 - \frac{\sum a_{viz}}{a_{P_{SIMPLE}}}$$

## 5.5 SIMPLEC

O algoritmo SIMPLEC é baseado na simplificação:

$$\frac{\sum a_{viz} (u'_{viz} - u'_p)}{a_p} \approx 0$$

Que é mais razoável que a simplificação feita no SIMPLE. Assim, o algoritmo SIMPLEC funciona com subrelaxação menos rigorosa para a pressão. Pode-se usar coeficiente de relaxação  $\alpha_p$  próximo de 1.



## 5.6 PISO (Pressure Implicit with Splitting of Operators)

O algoritmo PISO, para cada iteração, resolve duas vezes a equação da quantidade de movimento e a equação de correção de pressão.

Passo 1: Como no algoritmo SIMPLE, resolvemos:

$$u_P^* = \frac{\sum a_{viz} u_{viz}^*}{a_P} - d_P (p_e^* - p_w^*)$$

## 5.6 PISO (Pressure Implicit with Splitting of Operators)

Passo 2: Calculamos o erro de conservação da massa e resolvemos a equação de correção da pressão como no algoritmo SIMPLE.

Passo 3: Corrigimos velocidades e pressões como no algoritmo SIMPLE.

$$u_P^{**} = u_P^* + d_P(p'_w - p'_e)$$

$$p_P^{**} = p_P^* + p'_P$$

## 5.6 PISO (Pressure Implicit with Splitting of Operators)

Passo 4: A equação da Quantidade de Movimento é resolvida novamente (de forma não iterativa)

$$u_P^{***} = \frac{\sum a_{viz} u_{viz}^{**}}{a_P} - d_P (p_e^{**} - p_w^{**})$$

## 5.6 PISO (Pressure Implicit with Splitting of Operators)

Passo 5: Calculamos o novo erro de conservação da massa e resolvemos novamente a equação de correção da pressão, obtendo uma nova correção de pressão  $p''$ .

Passo 6: Corrigimos novamente velocidades e pressões.

$$u_P = u_P^{***} + d_P (p_w'' - p_e'')$$

$$p_P = p_P^{**} + p_P''$$

## 5.6 PISO (Pressure Implicit with Splitting of Operators)

O algoritmo PISO não é vantajoso sobre o algoritmo SIMPLE em soluções de escoamentos em regime permanente, mas em geral é mais eficiente para escoamentos em regime transiente.

# Bibliografia

Versteeg, H,K; Malalasekera, W., “An introduction to Computational Fluid Dynamics: The Finite Volume Method”, 2nd Edition, Pearson Education Limited, 2007.

Maliska, C.R., “Transferência de Calor e Mecânica dos Fluidos Computacional, LTC editora, 1995.

Apsley, D., CFD Lecture Notes, University of Manchester, Spring 2007.

ANSYS CFX 14 Theory Guide, 2012.