Universidade Federal de Uberlândia
Laboratório de Mecânica dos Fluidos

Revisão do método de simulação térmica bidimensional e tridimensional, com implementação de novas rotinas de otimização

Felipe J. O. Ribeiro

Orientador: Prof. Dr. Aristeu da Silveira Neto

25 de janeiro de 2019

Sumário

Análise térmica bidimensional e tridimensional

• A advecção natural é um fenômeno de turbulência clássico. Ele apresenta muitos exemplos tanto industriais quanto práticos do dia a dia. Para se simular tal fenômeno um domínio bi ou tridimensional é necessário (tridimensional se modelos de turbulência são implementados). Dessa forma torna-se uma extensão natural do trabalho desenvolvido até o momento. Apesar de lidar com mudanças de massa dos fluidos, ainda não se chega a se tratar de compressibilidade. Dessa forma, nesse trabalho espera-se:
- Desenvolver modelos 2D e 3D representativos.
- Implementar-se métodos de otimização conforme necessários, como mpi e opengl.
- Desenvolver método de apresentação visual integrado ao fortran.

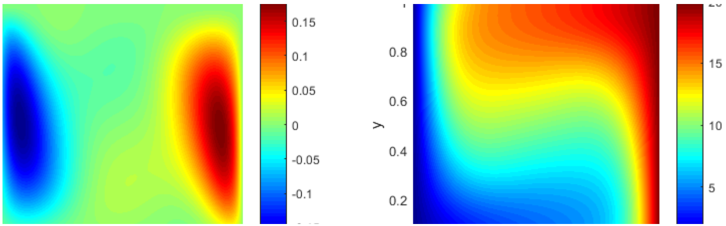

Figura 1: Convecção natural.

Sumário

Domínio bidimensional com advecção imposta

- Converter código bidimensional antigo em MatLab para c++.
- Desenvolver advecção, com velocidades em x e y impostas para todo o domínio.
- Desenvolver biblioteca visual para este estudo com openGL.
- Converter para o fortran os códigos.
- Experimentar plataforma openGL com fortran.
- Estudar acoplamento velocidade pressão.

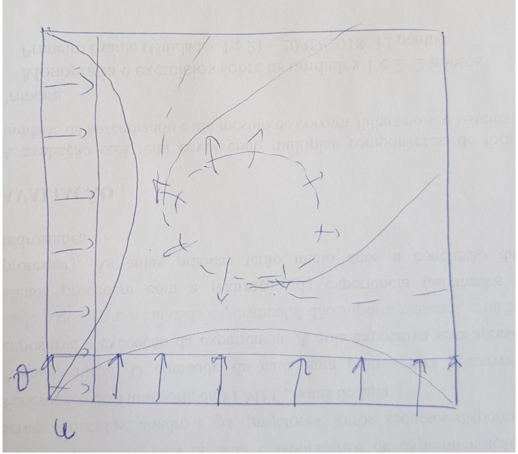

Figura 2: Esquema do professor.

Modelo matemático diferencial

- Foi utilizada a equação da energia térmica como base, com termos advectivos.
- Foi ignorada a dimensão em z, considerando-se auto similaridade neste eixo.

$$\frac{\partial T}{\partial t} + u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} = \alpha \left[\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right] \tag{1}$$

Modelo numérico

- Todas as derivadas parciais de primeira ordem foram discretizadas pelo método de Newton.
- Todas as derivadas parciais de segunda ordem foram discretizadas utilizando-se o método das diferenças centradas.

$$
\begin{aligned}
&\frac{T_i^{(j,k)} - T_{i-1}^{(j,k)}}{\Delta t} + u\frac{T_{i-1}^{(j,k)} - T_{i-1}^{(j-1,k)}}{\Delta x} + v\frac{T_{i-1}^{(j,k)} - T_{i-1}^{(j,k-1)}}{\Delta y} \\
&= \alpha\left[\frac{T_{i-1}^{(j+1,k)} - 2T_{i-1}^{(j,k)} + T_{i-1}^{(j-1,k)}}{\Delta x^2} + \frac{T_{i-1}^{(j,k+1)} - 2T_{i-1}^{(j,k)} + T_{i-1}^{(j,k-1)}}{\Delta y^2}\right]
\end{aligned}
\tag{2}
$$

Modelo numérico

- Com algumas simplificações chega-se na expressão utilizada no código:

$$
\begin{aligned}
T_i^{(j,k)} = T_{i-1}^{(j,k)} \left(1 - 4\frac{\alpha \Delta t}{\Delta s^2} - \frac{(u+v)\Delta t}{\Delta s}\right) + T_{i-1}^{(j-1,k)} \left(\alpha \frac{\Delta t}{\Delta s^2} + u\frac{\Delta t}{\Delta s}\right) \\
+ T_{i-1}^{(j,k-1)} \left(\alpha \frac{\Delta t}{\Delta s^2} + v\frac{\Delta t}{\Delta s}\right) + T_{i-1}^{(j+1,k)} \left(\alpha \frac{\Delta t}{\Delta s^2}\right) + T_{i-1}^{(j,k+1)} \left(\alpha \frac{\Delta t}{\Delta s^2}\right)
\end{aligned}
\tag{3}
$$

Sumário

Sumário

Agradecimentos

PETROBRAS

Formação e Desenvolvimento

CNPq

Conselho Nacional de Desenvolvimento Científico e Tecnológico

CAPES

FAPEMIG

Obrigado.