

Universidade Federal de Minas Gerais - UFMG
Curso de Graduação em Engenharia de Sistemas

EMA255 - Fluidos e Termodinâmica Computacional

Lista de Exercícios de Dinâmica dos Fluidos

Aluno: Raphael Henrique Braga Leivas

Matrícula: 2020028101

Belo Horizonte

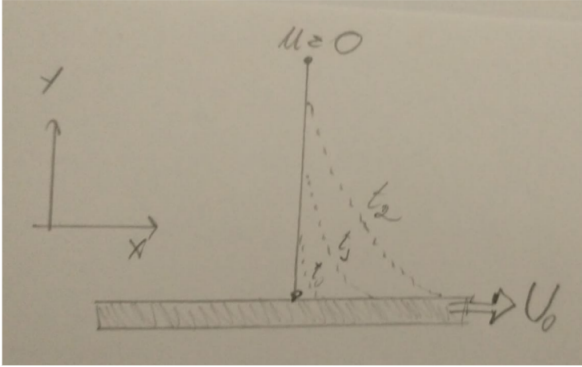
Semestre 2023/1

1 Primeira Questão (6 pts)

1.1 Análise Analítica

Um fluido de viscosidade μ e massa específica ρ está em repouso sobre uma placa horizontal quando ela começa a se mover com velocidade constante U_0 . A Figura 1.1 exibe a geometria do problema.

Figura 1.1: Geometria da questão 1.



Fonte: elaboração própria.

Como mostra a Figura 1.1, a linha média do fluido começa a ser arrastada pelo movimento da placa, e a cada instante de tempo t_0 , t_1 e t_2 o movimento da placa atinge pontos cada vez mais altos da linha média do fluido. Nesse cenário, a velocidade horizontal u do fluido é uma função de duas variáveis: posição vertical y e tempo t .

A equação diferencial que modela o problema é

$$\frac{\partial u}{\partial t} = \frac{\mu}{\rho} \frac{\partial^2 u}{\partial y^2} \quad (1.1)$$

com as seguintes condições de contorno:

$$\begin{cases} u(0, t) = U_0, & \text{Condição de contorno 1} \\ u(H, t) = 0, & \text{Condição de contorno 2} \\ u(y, 0) = 0, & \text{Condição inicial} \end{cases} \quad (1.2)$$

Veja que o problema é modelado por uma equação diferencial de segunda ordem em regime transiente, logo precisamos de duas condições de contorno e uma condição inicial.

No nosso sistema de coordenadas, $y = 0$ corresponde ao ponto do fluido em contato com a placa, e $y = H$ é a altura máxima que iremos considerar na análise. Note que, para H suficiente grande, faz sentido assumir que o movimento da placa não atinge esses pontos do fluido

a uma grande distância da placa, de modo que $u(H, t) = 0$.

O problema dado por (1.1) e (1.2) possui solução analítica dada por

$$u(y, t) = U_0 \left[1 - \operatorname{erf} \left(\frac{y}{2\sqrt{\frac{\mu}{\rho}t}} \right) \right] \quad (1.3)$$

em que erf é a função erro dada por

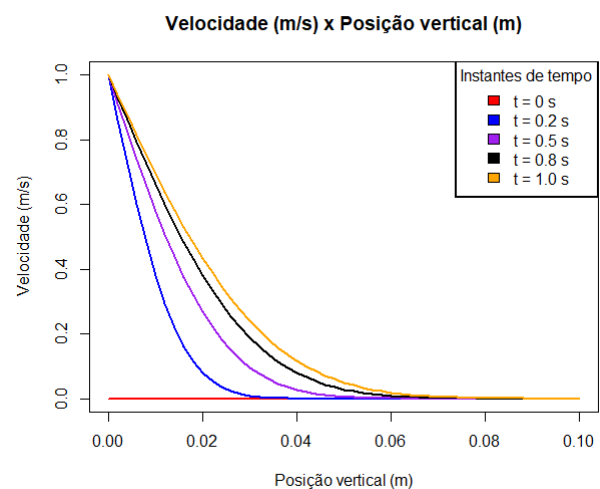
$$\operatorname{erf}(\beta) = \frac{2}{\sqrt{\pi}} \int_0^\beta e^{-x^2} dx \quad (1.4)$$

Usando linguagem R e a IDE RStudio, é possível obter a distribuição de velocidades do fluido a partir da equação (1.3), uma vez que a função erro erf já está implementada em pacotes da linguagem. Usamos os seguintes parâmetros:

- Domínio de análise vertical: $H = 10$ cm
- Domínio de análise no tempo: $T = 1$ s
- Viscosidade: $\mu = 0.29$ kg m⁻¹ s⁻¹
- Massa específica: $\rho = 891$ kg m⁻³
- Velocidade da placa: $U_0 = 1$ m/s

A Figura 1.2 exibe os valores de $u(y, t)$ em função do tempo e posição vertical obtidos através da equação (1.3) com os parâmetros acima.

Figura 1.2: Velocidade u do fluido em função da posição e tempo.



Fonte: elaboração própria.

As velocidades obtidas analiticamente na Figura 1.2 são condizentes com o analisado da geometria na Figura 1.1. Conforme o tempo passa, os

pontos do fluido apresentam velocidade u cada vez maior, uma vez que são arrastados pelo movimento da placa. Os pontos em $y > 7$ cm já não sofrem nenhum efeito do movimento da placa até $t = 1$ s.

Analisando os pontos específicos pedidos no exercício, temos

$$u(3 \text{ cm}, 0.5 \text{ s}) = 0.096 \text{ m/s}$$

$$u(3 \text{ cm}, 1 \text{ s}) = 0.24 \text{ m/s}$$

1.2 Análise Numérica

Uma vez entendido o que está acontecendo analiticamente, podemos fazer uma análise numérica do problema usando o método de diferenças finitas. O primeiro passo é discretizar a equação diferencial (1.1), trocando os diferenciais por diferenças.

$$\frac{u^{p+1} - u^p}{\Delta t} = \frac{\mu}{\rho} \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta y)^2} \quad (1.5)$$

Em (1.5) adotamos a seguinte simbologia:

- Sobrescrito p : indica o tempo da velocidade u . p indica o tempo atual, $p + 1$ o próximo tempo
- Subscrito i : indica a posição vertical da velocidade u . i indica a posição atual, $i + 1$ a próxima posição
- Δt : intervalo de discretização no tempo
- Δy : intervalo de discretização no espaço (vertical)

Assim, a linha média da Figura 1.1 é discretizada em $H/\Delta y$ elementos de comprimento Δy , e o tempo é transcorrido a partir de $t = 0$ até $t = T$ em intervalos de Δt . Vamos reescrever (1.5) de modo a expressar a velocidade futura em função da velocidade antiga.

$$u^{p+1} = u^p + \frac{\mu}{\rho} \frac{\Delta t}{(\Delta y)^2} (u_{i+1} - 2u_i + u_{i-1}) \quad (1.6)$$

Em (1.6), usamos o método explícito ao aplicar as diferenças finitas. Portanto, é fundamental analisar os critérios de estabilidade para que a solução converja. Em transferência de calor, vimos que a equação diferencial de condução unidimensional em regime transiente é

$$\frac{1}{\alpha} \frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial y^2} \Rightarrow \frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial y^2} \quad (1.7)$$

onde α é a difusividade térmica do material. Nesse modelo, o critério de estabilidade é, segundo Incropera (2007),

$$\frac{\alpha \Delta t}{(\Delta y)^2} \leq \frac{1}{2} \quad (1.8)$$

Comparando as equações diferenciais (1.1) e (1.7), podemos ver que as constantes α e μ/ρ possuem o mesmo papel, apesar de uma ter um significado físico de transferência de calor e outra um significado de mecânica dos fluidos. Nesse sentido, podemos inferir que o critério de estabilidade de (1.6) é

$$\frac{\mu}{\rho} \frac{\Delta t}{(\Delta y)^2} \leq \frac{1}{2} \quad (1.9)$$

Concluída essa análise, podemos jogar (1.6) no computador para obter a solução numérica do problema. Novamente usando R e a IDE RStudio, o pseudocódigo abaixo mostra o procedimento para resolver o problema numericamente.

```
listaSoluções ← Lista()
i ← 0
for Δt em T do
  i ← i + 1
  soluçãoAntiga ← listaSoluções[i]
  for Δy em H do
    calcule novaVelocidade
  end for
  listaSoluções[i + 1] ← novaVelocidade
end for
return listaSoluções
```

O pseudocódigo acima está bastante simplificado, mas ele mostra a ideia principal: percorremos dois loops, um para o tempo e um para a posição vertical. Percorremos todos os intervalos verticais, calculando as velocidades, para cada intervalo de tempo. Ao final de cada loop do tempo, salva o novo vetor de velocidades calculado na lista de soluções. No final da execução, a lista de soluções tem $T/\Delta t$ elementos, e cada elemento é um vetor de tamanho $H/\Delta y$.

O código completo desenvolvido em R, que implementa o pseudocódigo acima, pode ser visto no ANEXO A. A Figura 1.3 mostra a distribuição de velocidades obtida com o código do ANEXO A para $\Delta t = 0.05$ s e $\Delta y = 1$ cm.

Figura 1.3: Distribuição de velocidades obtida com a solução numérica.

Posição y (cm)	Velocidade (m/s)				
	0 s	0.25 s	0.5 s	0.75 s	1 s
0	1.00	1.00	1.00	1.00	1.00
1	0.00	0.46	0.59	0.66	0.70
2	0.00	0.14	0.28	0.37	0.44
3	0.00	0.03	0.10	0.18	0.25
4	0.00	0.00	0.03	0.08	0.12
5	0.00	0.00	0.01	0.03	0.05
6	0.00	0.00	0.00	0.01	0.02
7	0.00	0.00	0.00	0.00	0.01
8	0.00	0.00	0.00	0.00	0.00
9	0.00	0.00	0.00	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00

Fonte: elaboração própria.

Note que os valores escolhidos de Δt e Δy atendem ao critério de estabilidade:

$$\frac{\mu}{\rho} \frac{\Delta t}{(\Delta y)^2} = 0.1627 \leq 0.5$$

A tabela da Figura 1.3 foi construída com o pacote `flextable` da linguagem, que formata a lista de soluções na forma de uma tabela.

Os dados obtidos na Figura 1.3 são condizentes tanto com o esperado da geometria quanto do resultado analítico da Figura (1.2). Com o passar no tempo, as velocidades mais distantes da placa aumentam, e pontos muito distantes não sofrem efeito algum.

Em particular, na Figura 1.3, obtemos

$$u(3 \text{ cm}, 0.5 \text{ s}) = 0.10 \text{ m/s}$$

$$u(3 \text{ cm}, 1 \text{ s}) = 0.25 \text{ m/s}$$

através da solução numérica. O erro percentual com a solução analítica é dado por

$$E_{\%} = \frac{V_{\text{numerico}} - V_{\text{analitico}}}{V_{\text{analitico}}} 100\% \quad (1.10)$$

Assim, para $y = 3 \text{ cm}$ em $t = 0.5 \text{ s}$ temos erro

$$E_{\%} = \frac{0.10 - 0.096}{0.096} 100\% = 4.2\%$$

e para $y = 3 \text{ cm}$ em $t = 1 \text{ s}$,

$$E_{\%} = \frac{0.25 - 0.24}{0.24} 100\% = 4.2\%$$

com ambos erros inferiores a 5% em relação ao resultado analítico. Assim, podemos concluir que o modelo numérico é satisfatório.

2 Segunda Questão (9 pts)

2.1 Análise Analítica

O escoamento de um fluido de viscosidade μ e massa específica ρ , plenamente desenvolvido em regime permanente, em coordenadas cilíndricas, é modelado pela equação diferencial

$$-\frac{1}{\rho} \frac{dp}{dz} + \frac{\mu}{\rho} \frac{1}{r} \frac{d}{dr} \left(r \frac{du}{dr} \right) = 0$$

note que podemos cancelar a massa específica ρ , obtendo

$$\frac{1}{r} \frac{d}{dr} \left(r \frac{du}{dr} \right) = \frac{dp}{dz} \frac{1}{\mu} \quad (2.1)$$

Com um gradiente de pressão constante, temos que a velocidade axial u é uma função de apenas uma variável r . Assim, temos as condições de contorno:

$$\begin{cases} u(0) = \text{simetria} & \Rightarrow \frac{du}{dr} = 0, \\ u(R) = 0 \end{cases} \quad (2.2)$$

onde R é o raio da tubulação por onde o fluido está escoando. Quando $r = R$ estamos em contato com as paredes da tubulação, que naturalmente têm velocidade nula. A primeira condição de contorno (simetria) é consequência do sistema de coordenadas cilíndricas adotado.

Como (2.1) é uma EDO de primeira ordem, podemos tentar obter uma solução analítica por integração direta.

$$\int \frac{d}{dr} \left(r \frac{du}{dr} \right) dr = \int r \frac{dp}{dz} \frac{1}{\mu} dr$$

$$r \frac{du}{dr} = \frac{r^2}{2} \frac{dp}{dz} \frac{1}{\mu} + C_1$$

$$\int \frac{du}{dr} dr = \int \frac{r}{2} \frac{dp}{dz} \frac{1}{\mu} + \frac{C_1}{r} dr$$

$$u(r) = \frac{r^2}{4} \frac{dp}{dz} \frac{1}{\mu} + C_1 \ln(r) + C_2 \quad (2.3)$$

A equação (2.3) é a solução geral para a EDO (2.1). Para que ela atenda às condições de contorno de (2.2), de imediato temos $C_1 = 0$, pois o logaritmo natural não está definido para $r = 0$, sendo que fisicamente já sabemos que há uma velocidade no centro da tubulação. Assim, (2.3) se reduz a

$$u(r) = \frac{r^2}{4} \frac{dp}{dz} \frac{1}{\mu} + C_2$$

A constante C_2 pode ser identificada usando a condição de contorno $u(R) = 0$:

$$0 = \frac{R^2}{4} \frac{dp}{dz} \frac{1}{\mu} + C_2$$

$$C_2 = -\frac{R^2}{4} \frac{dp}{dz} \frac{1}{\mu}$$

Assim, a solução analítica do problema é

$$u(r) = \frac{r^2}{4} \frac{dp}{dz} \frac{1}{\mu} - \frac{R^2}{4} \frac{dp}{dz} \frac{1}{\mu}$$

$$u(r) = \frac{1}{4} \frac{dp}{dz} \frac{1}{\mu} (r^2 - R^2) \quad (2.4)$$

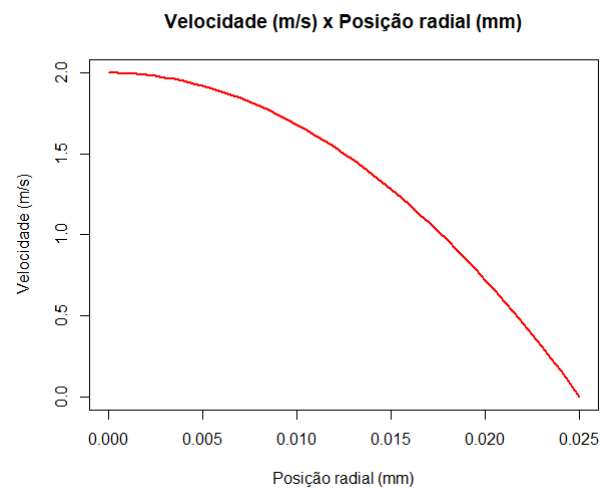
No problema temos os seguintes parâmetros:

- Raio do tubo: $R = 25 \text{ mm}$
- Viscosidade: $\mu = 1.01 \cdot 10^{-3} \text{ kg m}^{-1} \text{ s}^{-1}$
- Gradiente de pressão: $\frac{dp}{dz} = -12.928 \text{ N/m}$
- Massa específica: $\rho = 998 \text{ kg m}^{-3}$

Com esses parâmetros, a Figura 2.1 mostra o perfil de velocidade dentro da tubulação conforme a solução analítica em (2.4).

O perfil de velocidade da Figura 2.1 é consistente com o esperado fisicamente: a velocidade é máxima no centro do tubo, e zero nas paredes na tubulação, atendendo também às condições de contorno de (2.2). Além disso, igual visto durante as aulas da disciplina, temos um perfil parabólico para o escoamento plenamente desenvolvido na tubulação.

Figura 2.1: Solução analítica da questão 2.



Fonte: elaboração própria.

2.2 Análise Numérica

Concluída a análise analítica, podemos realizar uma análise numérica do problema. De modo semelhante ao feito na Questão 1, o primeiro passo é discretizar a equação diferencial (2.1) via diferenças finitas.

$$\frac{d}{dr} \left(r \frac{du}{dr} \right) = r \frac{dp}{dz} \frac{1}{\mu}$$

aplicando a regra do produto no primeiro membro da equação,

$$\frac{du}{dr} + r \frac{d^2u}{dr^2} = r \frac{dp}{dz} \frac{1}{\mu}$$

agora substituímos os diferenciais por diferenças finitas:

$$\frac{u_{i+1} - u_i}{\Delta r} + r \left(\frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta r)^2} \right) = r \frac{dp}{dz} \frac{1}{\mu}$$

$$u_{i+1} - u_i + r \left(\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta r} \right) = (\Delta r) r \frac{dp}{dz} \frac{1}{\mu}$$

agrupando as velocidades,

$$u_{i+1} \left(1 + \frac{r}{\Delta r} \right) + u_i \left(-1 - 2\frac{r}{\Delta r} \right) + u_{i-1} \left(\frac{1}{\Delta r} \right) = (\Delta r) r \frac{dp}{dz} \frac{1}{\mu} \quad (2.5)$$

Note que (2.5) é a forma implícita de diferenças finitas para os nós internos do domínio. Os

termos ligados às velocidades são os coeficientes do sistema linear, e o segundo membro de (2.5) tem os coeficientes independentes do sistema linear.

Antes de tentar montar o sistema linear em software, precisamos identificar as equações dos nós do contorno da geometria. Na parede, temos condição de contorno $u(R) = 0$, logo a equação para $i = R$ é

$$u_i = 0 \quad (2.6)$$

Além disso, na linha de centro temos simetria, dada matematicamente por

$$\frac{du}{dr} = 0$$

Durante a análise analítica, vimos que

$$\frac{du}{dr} = \frac{r}{2} \frac{dp}{dz} \frac{1}{\mu} + \frac{C_1}{r}$$

com $C_1 = 0$,

$$\frac{du}{dr} = \frac{r}{2} \frac{dp}{dz} \frac{1}{\mu}$$

Discretizando a expressão acima, e usando $r = \Delta r/2$ para analisarmos o que está acontecendo no nó central, temos

$$\begin{aligned} \frac{u_{i+1} - u_i}{\Delta r} &= \frac{\Delta r/2}{2} \frac{dp}{dz} \frac{1}{\mu} \\ u_{i+1} - u_i &= \frac{(\Delta r)^2}{4} \frac{dp}{dz} \frac{1}{\mu} \end{aligned} \quad (2.7)$$

Assim, quando estivermos percorrendo cada Δr no domínio, identificamos se ele é um nó interno, nó central ou nó em contato com as paredes. Em seguida usamos a seguinte metodologia:

- Se estamos em um nó interno, calcula os coeficientes através de (2.5)
- Se estamos na parede, calcula os coeficientes através de (2.6)
- Se estamos no nó central, calcula os coeficientes através de (2.7)

Uma vez calculados todos os coeficientes, teremos uma matriz de coeficientes A de dimensão $N \times N$, onde $N = R/\Delta y$, e um vetor de coeficientes independentes B de dimensão $N \times 1$. O vetor solução X , que contém as velocidades u de cada nó, tem dimensão $N \times 1$ e é calculado através da equação matricial

$$X = A^{-1}B \quad (2.8)$$

O pseudocódigo abaixo mostra a ideia principal do software desenvolvido em R que implementa essa solução numérica. O software completo pode ser visto no ANEXO B.

```

N ← R/Δy
A ← Matriz(N × N)
B ← Vetor(N × 1)
i ← 0
for cada Δy em R do
  if Δy atual é nó interno then
    A[linha i] ← equação (2.5)
    B[i] ← equação (2.5)
  end if
  if Δy atual é parede then
    A[linha i] ← equação (2.6)
    B[i] ← equação (2.6)
  end if
  if Δy atual é nó central then
    A[linha i] ← equação (2.7)
    B[i] ← equação (2.7)
  end if
  i ← i + 1
end for
X ← inversa(A) * B
return X

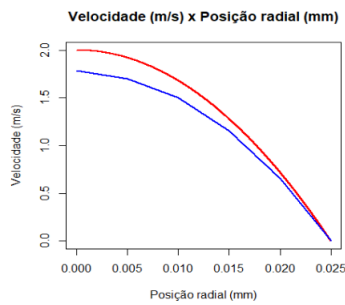
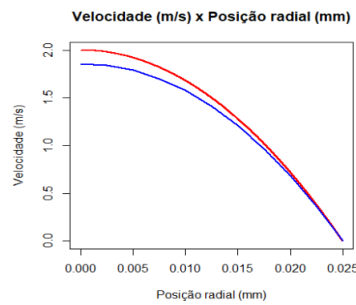
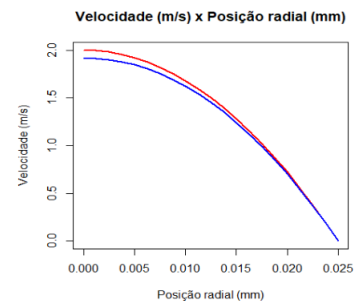
```

Ao contrário do problema 1, temos apenas um loop no código. Não temos marcha no tempo uma vez que o regime de escoamento é permanente. Nesse caso, basta percorrer cada intervalo Δy em R , identificar a natureza nesse nó, calcular os coeficientes, jogar nas matrizes A e B e resolver a equação matricial.

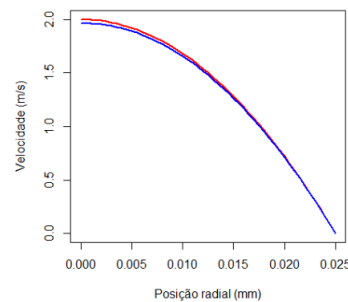
A Figura 2.2 mostra uma comparação entre os valores numéricos obtidos no código com a solução analítica da Figura 2.1 para diversas malhas.

Na Figura 2.2 (a) temos apenas 5 nós, e a solução obtida está longe da analítica. Além disso, é possível ver as retas das diferenças finitas na curva azul, uma vez que N é baixo, e assim não temos uma curva suave como a analítica. Conforme N aumenta nas Figuras 2.2 (b), (c) e (d), a solução numérica vai se aproximando da analítica, ficando cada vez mais suave, até ficar praticamente idêntica à analítica para $N = 100$.

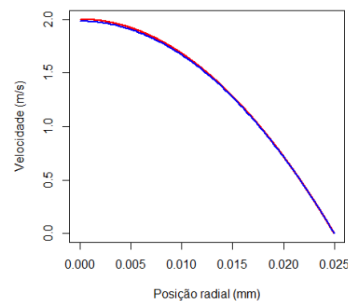
Com base nos gráficos da Figura 2.2, temos que modelo numérico desenvolvido é satisfatório para $N \geq 50$. Note que usamos o método implícito, e portanto não precisamos nos preocupar com critérios de estabilidade como na Problema 1. Nesse sentido,

Figura 2.2: Solução numérica (azul) e solução analítica (vermelha) para vários valores de N .(a) $N = 5$ (b) $N = 10$ (c) $N = 20$ 

Velocidade (m/s) x Posição radial (mm)

(d) $N = 50$

Velocidade (m/s) x Posição radial (mm)

(e) $N = 100$

Fonte: elaboração própria.

não temos restrições teóricas para o valor máximo de N . Contudo, o uso de N muito elevando aumenta o custo computacional do software, principalmente da matriz de coeficientes que possui N^2 elementos. Assim, não é necessário, com base nos dados obtidos, usar N muito maior que 100 para esse modelo.