



Instituto de Química
IQ - UFG



ENGENHARIA QUÍMICA
Universidade Federal de Goiás

Uma visão geral sobre a Fluidodinâmica Computacional (CFD)

Professor Dyrney Araújo dos Santos
Universidade Federal de Goiás

Engenharia Química

Disciplina: Introdução à Fluidodinâmica Computacional

site: www.dyrney.com

1. Visão Geral sobre CFD

1.1 Métodos Tradicionais para o Tratamento de um Problema de Engenharia

➤ Experimentação em laboratório



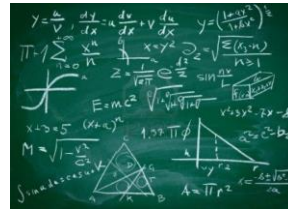
Principais vantagens:

- trata da configuração física real;
- são usados para validar modelos analíticos ou numéricos;
- trata de problemas complexos que podem ser difíceis de modelar matematicamente.

Principais desvantagens:

- alto custo de matéria prima e de equipamentos;
- preparar e executar experimentos pode ser demorado;
- difícil de se controlar todas as variáveis que afetam o fenômeno físico.

➤ Métodos Analíticos



Principais vantagens:

- solução exata do problema físico;
- não requerem equipamentos caros ou recursos computacionais avançados;
- ajudam a entender os princípios físicos subjacentes ao problema;
- rápida solução.

Principais desvantagens:

- implica muitas simplificações, o que pode limitar suas aplicações;
- problemas complexos como fluxos turbulentos ou geometrias irregulares, pode ser intratáveis analiticamente;
- dificuldade em adaptar modelos analíticos para incluir novos parâmetros ou condições.

➤ Experimentação Numérica



Principais vantagens:

- aplica-se a problemas complexos diminuindo custos de projeto;
- fornece informações detalhadas sobre o campo de escoamento em todo o domínio;
- permite estudar condições extremas ou perigosas sem riscos físicos.

Principais desvantagens:

- alto custo de processamento computacional a depender do problema físico;
- curva de aprendizado: dominar ferramentas numéricas e entender suas limitações pode ser desafiador;
- resultados numéricos precisam ser validados com dados experimentais ou analíticos

1. Visão Geral sobre CFD

1.2 Alguns Métodos Complementares Modernos para o Tratamento de um Problema de Engenharia

➤ Machine Learning (ML) e Inteligência Artificial (IA)



Principais vantagens:

- pode lidar com problemas altamente complexos e não lineares;
- aprendizado a partir de dados, sem necessidade de modelos físicos detalhados;
- rápido uma vez que o modelo é treinado.

Principais desvantagens:

- requer grandes volumes de dados de alta qualidade para treinamento;
- pode ser uma "caixa preta", com pouca interpretabilidade física;
- dependência da qualidade dos dados e do treinamento.

➤ Realidade Virtual (VR) e Realidade Aumentada (AR)



Principais vantagens:

- melhora a compreensão de sistemas complexos;
- permite interação imersiva com modelos 3D;
- reduz custos de treinamento e prototipagem.

Principais desvantagens:

- custo inicial de hardware e software;
- requer desenvolvimento de modelos 3D detalhados;
- aplicação ainda limitada em alguns campos da engenharia.

Observação 1: Em muitos casos, uma combinação desses métodos é a abordagem mais eficaz. Por exemplo, usar métodos analíticos para insights iniciais, CFD para explorar cenários complexos e experimentos para a validação final.

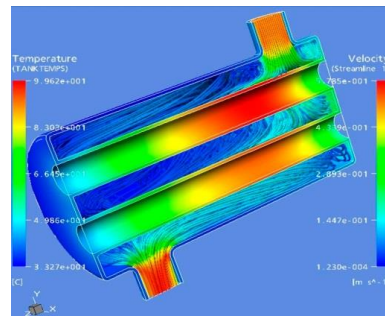
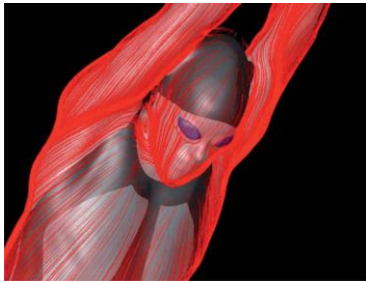
Observação 2: Neste curso iremos nos atentar somente à “Experimentação Numérica”, mais especificamente à Fluidodinâmica Computacional, discutida a seguir .

1. Visão Geral sobre CFD

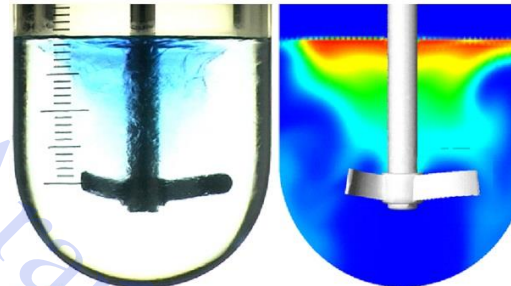
1.3 O que é Fluidodinâmica Computacional?

Fluidodinâmica Computacional (do inglês “Computational Fluid Dynamics – CFD”): é a análise de sistemas envolvendo escoamento de fluido, transferência de calor e fenômenos associados, tais como turbulência e reações químicas, por meio de simulações baseadas em computador. Esta técnica é muito poderosa e abrange uma larga faixa de aplicações industriais e não industriais.

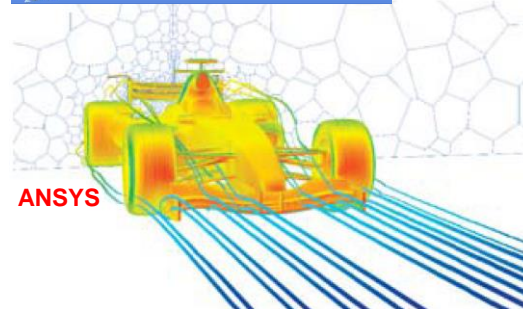
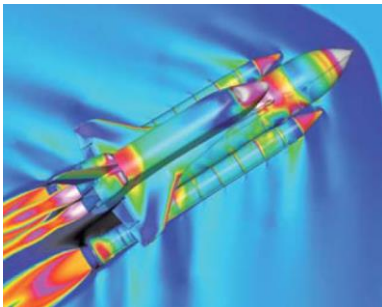
Speedo
International
Limited



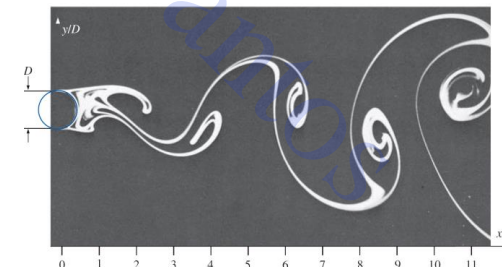
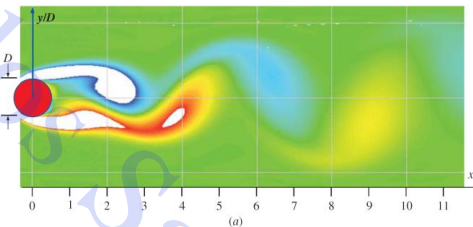
FZR
Forschungszentrum
Rossendorf



NASA



ANSYS



Sadatashi
Taneda

Çengel e Cimbala (2015)

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

Principais tarefas:

- ❑ **Definição da geometria:** criar ou importar o modelo 3D do sistema a ser estudado (ex.: tubulação, asa de avião, trocador de calor, bomba centrífuga etc.);
- ❑ **Geração da malha computacional:** dividir o domínio em pequenos elementos ou células computacionais (malha computacional ou “grid”), onde as equações diferenciais parciais serão resolvidas. A malha pode ser estruturada (organizada) ou não estruturada (mais flexível);
- ❑ **Definição das condições de contorno:** especificar como o fluido se comporta nas fronteiras do domínio (ex.: velocidade de entrada, pressão de saída, paredes adiabáticas, condição de não deslizamento etc.);
- ❑ **Escolha do modelo físico:** selecionar os modelos matemáticos que descrevem o fenômeno (ex.: escoamento laminar ou turbulento, compressível ou incompressível, isotérmico ou não isotérmico etc.);
- ❑ **Especificação das propriedades físicas do fluido:** propriedades físicas constantes ou funções da temperatura e pressão (ex.: viscosidade, densidade, condutividade térmica, calor específico etc.);
- ❑ **Configuração do solver:** escolher o método numérico e os parâmetros de convergência.



Versteeg e Mallasekera (2007): “Over 50% of the time spent in industry on a CFD project is devoted to the definition of the domain geometry and grid generation”

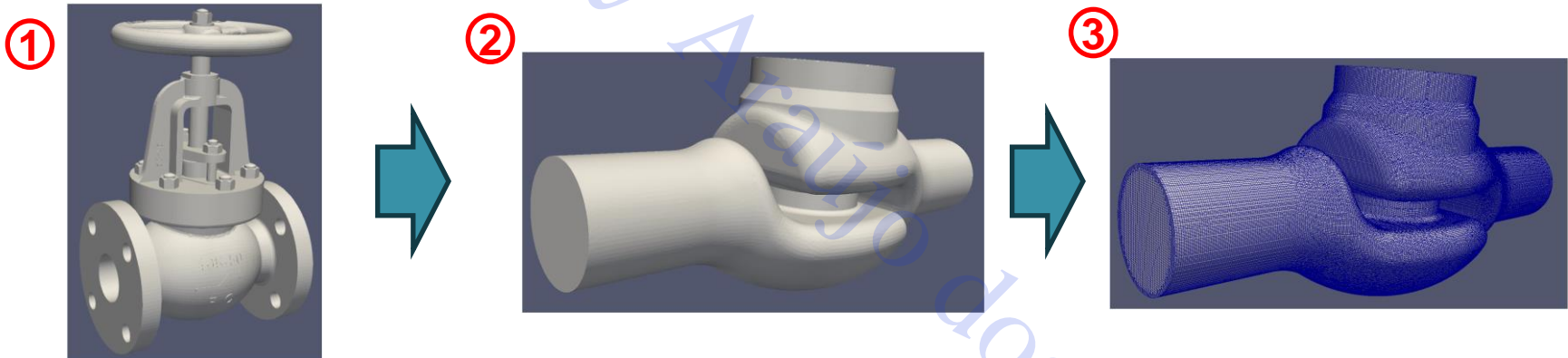
1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

EXEMPLO: cálculo do escoamento incompressível, isotérmico e laminar de um fluido newtoniano no interior de uma válvula do tipo globo.



1: Modelo 3D de uma válvula globo contendo todos os seus itens. Este modelo 3D ainda deve ser simplificado pois o domínio de cálculo de interesse é somente a região por onde o fluido escoava;

2: Modelo 3D de uma válvula globo simplificado. Foram desprezados todos os itens desnecessários ao estudo do escoamento do fluido, tais como, flanges, parafusos, volante etc. Somente o interior da válvula foi considerado, ou seja, região onde o fluido escoava;

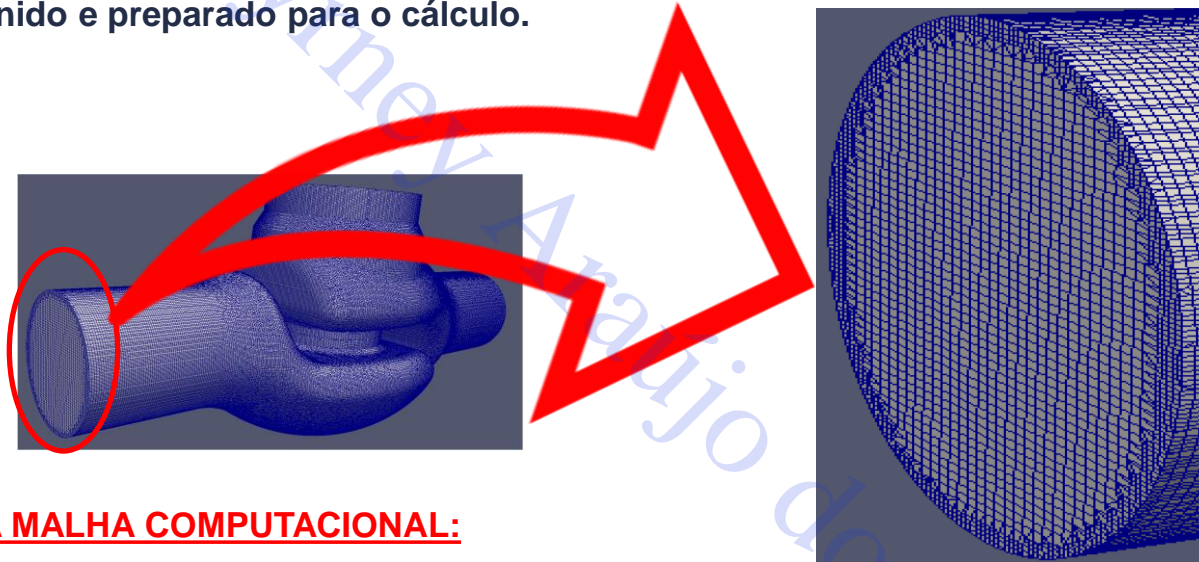
3: Divisão do domínio de cálculo em vários subdomínios ou elementos. Cada um destes subdomínios é conhecido como “célula computacional”, sendo o conjunto de todas as células computacionais denominada “malha computacional”.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

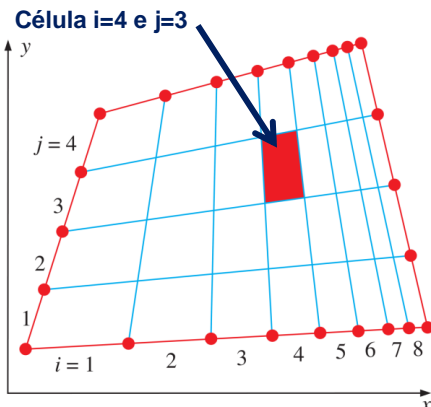
O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

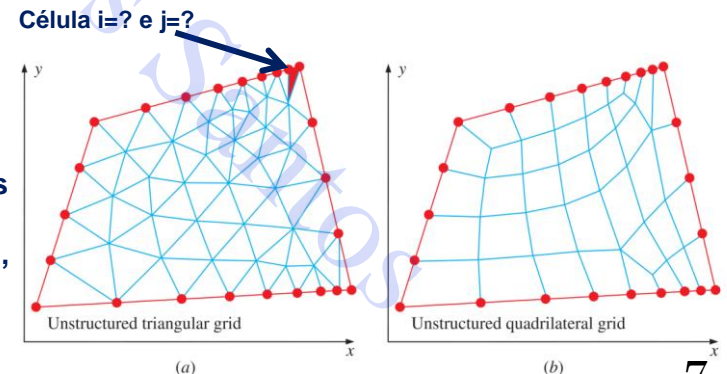


CLASSIFICAÇÃO DA MALHA COMPUTACIONAL:

a) **Malha Estruturada:** células com quatro lados (2D) ou seis lados (3D). Qualquer célula pode ser identificada, exclusivamente, pelos índices (i,j,k) .



b) **Malha Não-Estruturada:** em geral triângulos ou quadriláteros (2D) e tetraedros ou hexaedros (3D). As células não podem ser identificadas, exclusivamente, pelos índices (i,j,k) .



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

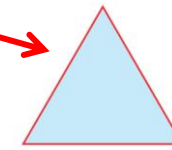
QUALIDADE DA MALHA COMPUTACIONAL

A qualidade da malha é um dos fatores mais críticos para produzir soluções confiáveis de CFD. Existem diferentes parâmetros utilizados para se medir a qualidade da malha, como por exemplo a “**assimetria equiangular**” ou “**equiangular skewness**” (Q_{EAS})

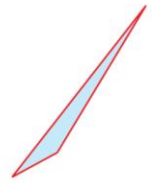
$$Q_{EAS} = \text{MAX} \left(\frac{\theta_{\max} - \theta_{\text{equal}}}{180^\circ - \theta_{\text{equal}}}, \frac{\theta_{\text{equal}} - \theta_{\min}}{\theta_{\text{equal}}} \right)$$

Células triangulares: $\theta_{\text{equal}} = 60^\circ$
 Células quadriláteras: $\theta_{\text{equal}} = 90^\circ$

(a) Triangular cells



Zero skewness



High skewness

(b) Quadrilateral cells



Zero skewness

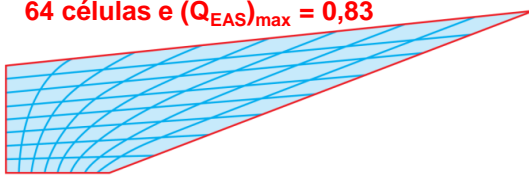


High skewness

Genggel e Cimbalá (2015)

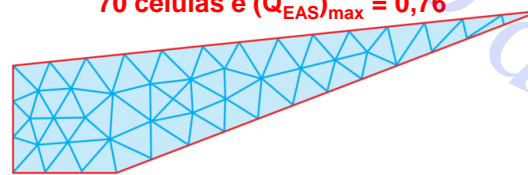
Comparação de 4 Malhas diferentes

64 células e $(Q_{EAS})_{\max} = 0,83$



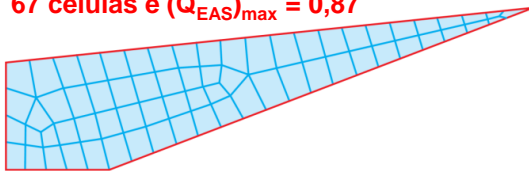
(a)

70 células e $(Q_{EAS})_{\max} = 0,76$



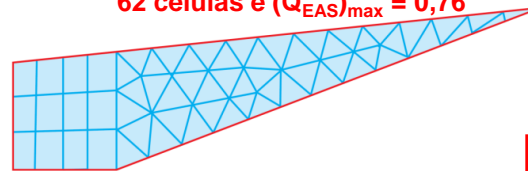
(b)

67 células e $(Q_{EAS})_{\max} = 0,87$



(c)

62 células e $(Q_{EAS})_{\max} = 0,76$



(d)

(a) Malha estruturada

(c) Malha quadrilateral não estruturada

(b) Malha triangular não estruturada

(d) Malha híbrida

Regra prática: uma malha não estruturada de alta qualidade é melhor do que uma malha estruturada de má qualidade

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

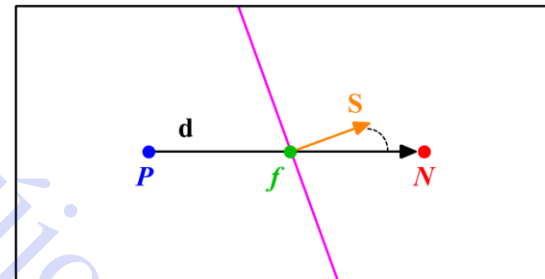
O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

QUALIDADE DA MALHA COMPUTACIONAL

Outros exemplos de parâmetros de qualidade de uma malha são a “**não ortogonalidade**” e a “**razão de aspecto**”

Exemplo: a “**não ortogonalidade**” da célula computacional é o desvio angular do vetor “**S**” (localizado no centro da face “**f**”) do vetor “**d**” que conecta os dois centros de células “**P**” e “**N**”.



Exemplo: a **razão de aspecto** da célula computacional (2D) é a razão entre o seu lado mais longo (Δx) e o seu lado mais curto (Δy).



Observação 3: Não há uma teoria escrita quando se trata da geração de malha computacional. Basicamente, todo o processo depende da experiência do usuário e de boas práticas padrão. Alguns códigos de CFD sugerem valores máximos permitidos para os parâmetros aqui mencionados a fim de se obter resultados satisfatórios.

Ex.: máximo skewness igual a 4 e máxima não ortogonalidade igual a 65° (OpenFOAM).

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

TESTE DE INDEPENDÊNCIA DE MALHAS

Qual o número ideal de células computacionais em uma malha computacional?

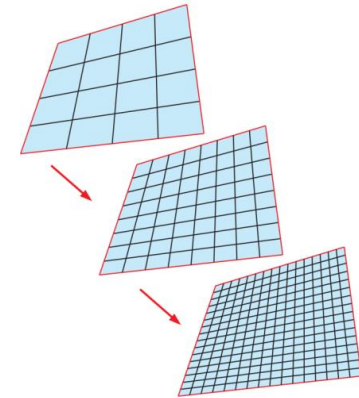
Analogia entre a Resolução de uma Foto e uma Malha Computacional

Quanto maior o número de pixels de uma foto (ou de células em uma malha computacional) maior será sua resolução (mais “coisas” iremos “enxergar”).

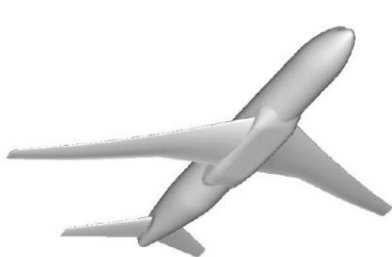


A malha é continuamente refinada até que não haja diferenças significativas nos resultados.

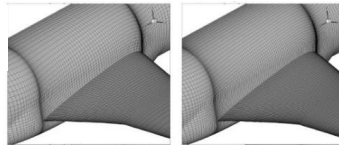
DICA (Çengel e Cimbala, 2015): refine-se a malha por um fator de 2 em todas as direções ou em 20% (devido às limitações computacionais) e repetir a simulação. Se os resultados não mudarem de forma apreciável, a malha original é utilizada



Ex.: Coeficiente de arraste (C_D) (AIAA CFD Drag Prediction Workshop)

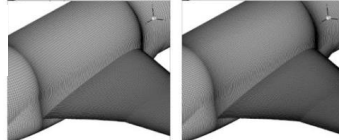


Malha “grossa”



Malha “média”

Malha “fina”



Malha “extra fina”

	Malha “grossa”	Malha “média”	Malha “fina”	Malha “extrafina”
células	≈ 3 500 000	≈ 11 000 000	≈ 36 000 000	≈ 105 000 000
C_D	0.0282	0.0270	0.0268	0.0269

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO

As condições de contorno são essenciais para definir como o fluido e a temperatura se comportam nas fronteiras do domínio de simulação. Essas condições garantem que o problema esteja bem formulado e que a solução seja fisicamente significativa.



1. Condições de Contorno para o Escoamento (Hidrodinâmica):

a) Velocidade Especificada (condição de Dirichlet)

- Define o valor da velocidade do fluido (\mathbf{u}) em uma entrada ou superfície.
- Exemplo: Em uma entrada de duto, $\mathbf{u} = (u_0, 0, 0)$, sendo u_0 a velocidade na direção x ; pode-se especificar, também, um perfil de velocidade conhecido, por exemplo, um perfil parabólico (escoamento laminar completamente desenvolvido na entrada).
- Aplicação: Entrada de um duto ou bocal.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO



b) Pressão Especificada (condição de Dirichlet)

- Define o valor da pressão (**P**) em uma superfície.
- Exemplo: Pressão atmosférica na saída de um duto ($P_{\text{manométrica}} = 0$).
- Aplicação: Saída de um sistema aberto para o ambiente.

c) Parede (condição de não deslizamento; “No-Slip Condition”)

- Define que a velocidade do fluido é zero na superfície da parede (aderência do fluido).
- Exemplo: Superfícies sólidas, como paredes de um duto ou superfície de um objeto; $\mathbf{u} = (0, 0, 0)$.
- Aplicação: Qualquer superfície sólida em contato com o fluido.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO

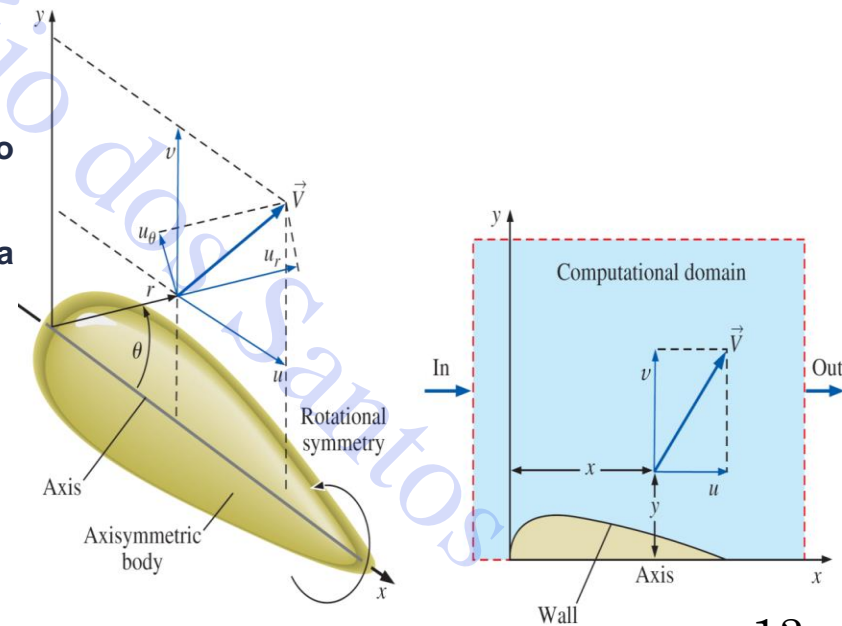


d) Simetria

- Define que o escoamento é simétrico em relação a um plano ou eixo.
- Exemplo: Redução do domínio de simulação para economizar recursos computacionais.
- Aplicação: Problemas com geometrias simétricas.

$$\frac{\partial u}{\partial n} = 0$$

sendo n a direção normal ao plano de simetria



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO

e) Condição de Saída Livre (“Outflow”)

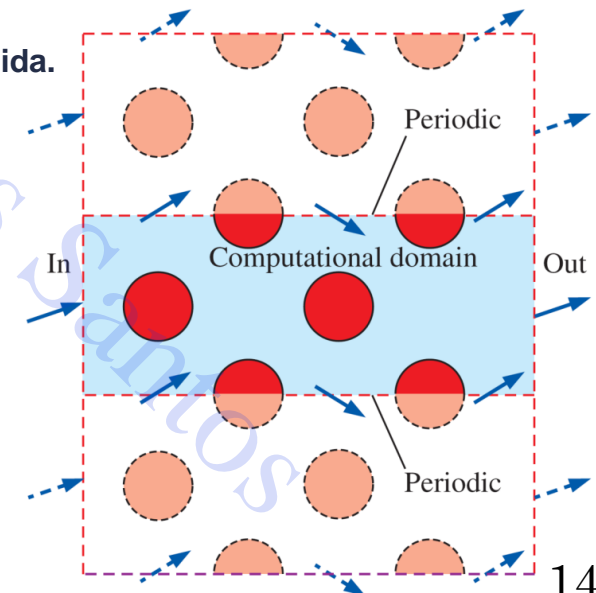
- ❑ Define que o fluido sai do domínio sem reflexão ou distúrbios.
- ❑ Exemplo: Saída de um duto onde o escoamento está totalmente desenvolvido.
- ❑ Aplicação: Saída de sistemas onde a pressão ou velocidade não é conhecida.

$$\frac{\partial u}{\partial n} = 0, \quad \frac{\partial P}{\partial n} = 0 \quad \text{sendo } n \text{ a direção normal à saída do domínio}$$

f) Condição de Periodicidade

- ❑ Define que o escoamento se repete periodicamente em uma direção.
- ❑ Exemplo: Escoamento em um canal infinito ou com padrões repetitivos.
- ❑ Aplicação: Problemas com geometrias periódicas.

$$u(x, y, z) = u(x + L, y, z) \quad \text{sendo } L \text{ o comprimento do período}$$



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO



2. Condições de Contorno para Transferência de Calor (Térmica):

a) Temperatura Especificada (condição de Dirichlet)

- Define o valor da temperatura em uma superfície (T).
- Exemplo: Temperatura constante em uma parede ou entrada ($T_{\text{parede}} = 300 \text{ K}$).
- Aplicação: Superfícies com temperatura controlada; mudança de fase sobre uma superfície.

b) Fluxo de Calor Especificado (condição de Neumann)

- Define o fluxo de calor na superfície (q).
- Exemplo: Fluxo de calor constante ou variável em uma parede.
- Aplicação: Superfícies com aquecimento ou resfriamento conhecido.

$$-k \frac{\partial T}{\partial n} = q_{\text{especificado}} \quad \text{sendo } k \text{ a condutividade térmica e } n \text{ a direção normal à superfície}$$

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO

c) Convecção (condição de Robin)

- ❑ Define que o calor é transferido por convecção entre a superfície e o fluido.
- ❑ Exemplo: a energia térmica que chega à uma face por condução é transferida por convecção para o fluido adjacente. Isto é válido mesmo em estado não estacionário, visto que uma face é hipoteticamente bidimensional, logo não possui massa nem volume para armazenar ou gerar energia térmica.

$$-k \frac{\partial T}{\partial n} = h(T_{superfície} - T_{fluido}) \quad \text{sendo } h \text{ o coeficiente de transferência de calor e } n \text{ a direção normal à superfície}$$

- ❑ Aplicação: Superfícies em contato com um fluido externo.

d) Radiação

- ❑ Define que o calor é transferido por radiação térmica.
- ❑ Exemplo: a energia térmica que chega à uma face por condução é transferida por radiação para o meio adjacente. Isto é válido mesmo em estado não estacionário, visto que uma face é hipoteticamente bidimensional, logo não possui massa nem volume para armazenar ou gerar energia térmica.

$$-k \frac{\partial T}{\partial n} = \varepsilon \sigma (T_{superfície}^4 - T_{ambiente}^4) \quad \text{sendo } \varepsilon \text{ a emissividade e } \sigma \text{ a constante de Stefan-Boltzmann } (\sigma = 5,67 \times 10^{-8} \text{ W}/(\text{m}^2 \cdot \text{K}^4)).$$

- ❑ Aplicação: Superfícies expostas a ambientes com troca radiativa.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO



e) Parede Adiabática

Define que não há transferência de calor através da superfície.

Exemplo: $q = -k \frac{\partial T}{\partial n} = 0$ ou $\frac{\partial T}{\partial n} = 0$

Aplicação: Superfícies isoladas termicamente.

3. Condições de Contorno Específicas para a Turbulência:

Em escoamentos turbulentos, é necessário definir condições de contorno para as variáveis de turbulência.

Ex.: Para o modelo de turbulência **k-ε** (lê-se “ká-epsilon”), modelo de duas equações adicionais de transporte, é necessário definir as condições, tanto para o **k** – energia cinética turbulenta, quanto para o **ε** – taxa de dissipação da energia cinética turbulenta).

Observação 4: Para maiores detalhes sobre a modelagem do escoamento turbulento, ver (<https://www.scielo.br/j/rbg/a/HpS6mWtczZwCh49P3fDNpKq/>)

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS CONDIÇÕES DE CONTORNO

a) Intensidade Turbulenta e Comprimento de Turbulência

- Define a intensidade turbulenta (I) e o comprimento da turbulência (L) na entrada.
- Exemplo: $I = 5\%$, $L = 0,07 \times D$ (para escoamentos em dutos circulares de diâmetro D)
- Aplicação: Entrada de escoamentos turbulentos.

$$k = \frac{3}{2} (I \times u_{\text{médio}})^2, \quad \varepsilon = C_{\mu}^{3/4} \frac{k^{3/2}}{L}$$

sendo C_{μ} uma constante do modelo de turbulência ($C_{\mu} = 0,09$) e $u_{\text{médio}}$ a velocidade média

b) Parede (Funções de Parede)

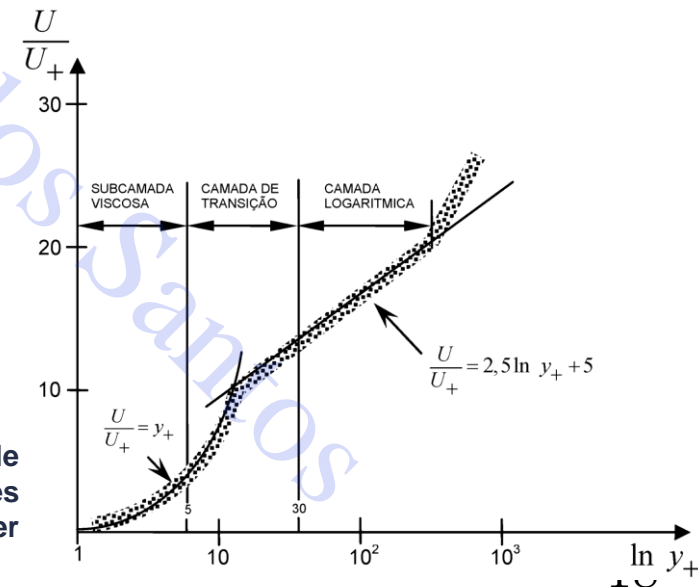
- Define como a turbulência pode ser modelada próximo às paredes.
- Exemplo: Lei de parede logarítmica.

$$u^+ = \frac{1}{\kappa} \ln(y^+) + B$$

Sendo u^+ a velocidade adimensional, y^+ a distância adimensional da parede, κ a constante de von Kármán ($\kappa \approx 0,41$) e B uma constante empírica ($B = 5$)

- Aplicação: uso de funções de parede para se estimar a velocidade na primeira célula computacional adjacente à parede sem precisar calcular toda a camada limite turbulenta, diminuindo, assim, o custo computacional.

Observação 5: O gráfico ao lado representa o perfil adimensional de velocidade junto a uma superfície sólida para um escoamento turbulento. Para maiores detalhes e definições dos dimensionais, ver <https://www.scielo.br/j/rbg/a/HpS6mWtczZwCh49P3fDNpKq>



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA A CONSTRUÇÃO DE MALHAS COMPUTACIONAIS

❑ SOFTWARES COMERCIAIS:

1) ANSYS Meshing

- ✓ Descrição: Parte do pacote ANSYS, é uma ferramenta poderosa e versátil para geração de malhas.
- ✓ Características:
 - Suporta malhas estruturadas, não estruturadas e híbridas;
 - Integração direta com ANSYS Fluent e CFX;
 - Ferramentas avançadas para refinamento de malha e controle de qualidade.
- ✓ Aplicações: CFD, FEA (“Finite Element Analysis”), simulações multifísicas.



2) ICEM CFD

- ✓ Descrição: Parte do pacote ANSYS, é conhecido por sua capacidade de gerar malhas hexaedrais estruturadas.
- ✓ Características:
 - Ferramentas avançadas para malhas estruturadas complexas;
 - Suporta scripts para automação.
- ✓ Aplicações: CFD em geometrias complexas.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA A CONSTRUÇÃO DE MALHAS COMPUTACIONAIS

❑ SOFTWARES COMERCIAIS:

3) STAR-CCM+ (Surface Wrapper)

- ✓ Descrição: Parte do software STAR-CCM+ da Siemens, possui ferramentas integradas para geração de malhas.
- ✓ Características:
 - Geração automática de malhas poligonais e poliedrais;
 - Ferramentas de reparo de geometria.
- ✓ Aplicações: CFD em indústrias automotiva, aeroespacial e energética.



4) HyperMesh

- ✓ Descrição: Desenvolvido pela Altair, é amplamente utilizado para pré-processamento em FEA e CFD.
- ✓ Características:
 - Interface poderosa para limpeza e preparação de geometrias;
 - Suporta malhas hexaedrais, tetraedrais e poligonais.
- ✓ Aplicações: Simulações estruturais e CFD.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

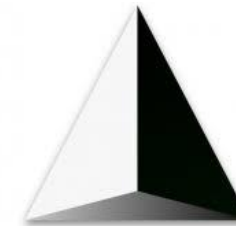
A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA A CONSTRUÇÃO DE MALHAS COMPUTACIONAIS

❑ SOFTWARES OPEN-SOURCE:

1) Gmsh

- ✓ Descrição: Software open-source para geração de malhas 2D e 3D.
- ✓ Características:
 - Suporta malhas estruturadas e não estruturadas;
 - Interface gráfica e scripting via linguagem própria;
 - Exporta para formatos como .msh, .stl, .vtk.
- ✓ Aplicações: FEA, CFD, simulações acadêmicas.



Gmsh

2) Salome (SMESH)

- ✓ Descrição: Plataforma open-source para simulação numérica, com módulo de geração de malhas (SMESH).
- ✓ Características:
 - Suporta malhas tetraedrais, hexaedrais e híbridas;
 - Integração com outros módulos de simulação.
- ✓ Aplicações: FEA, CFD, simulações multifísicas.

SALOME

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

A) Pré-Processamento: é a etapa de preparação do problema para a simulação. Aqui, o domínio do problema é definido e preparado para o cálculo.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA A CONSTRUÇÃO DE MALHAS COMPUTACIONAIS

❑ SOFTWARES OPEN-SOURCE:

3) snappyHexMesh

- ✓ Descrição: Ferramenta de geração de malhas incluída no OpenFOAM.
- ✓ Características:
 - Gera malhas hexaedrais cortadas (cut-cell) a partir de geometrias CAD;
 - Ideal para escoamentos complexos com geometrias detalhadas.
- ✓ Aplicações: CFD com OpenFOAM.

Open  FOAM

4) MeshLab

- ✓ Descrição: Software open-source focado em processamento de malhas 3D.
- ✓ Características:
 - Ferramentas para limpeza, reparo e otimização de malhas;
 - Ideal para malhas provenientes de escaneamento 3D.
- ✓ Aplicações: Visualização e preparação de malhas para simulações.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.



Principais tarefas:

- ❑ **Discretização das equações**: Transformar as equações diferenciais parciais (EDPs) em sistemas de equações algébricas que podem ser resolvidas numericamente;
- ❑ **Iteração e convergência**: O solver realiza cálculos iterativos até que a solução convirja (ou seja, até que os resultados parem de mudar significativamente);
- ❑ **Monitoramento**: Acompanhar o progresso da simulação e garantir que está convergindo corretamente.

MÉTODOS DE DISCRETIZAÇÃO DAS EQUAÇÕES DE TRANSPORTE

A seguir estão listados os principais métodos de discretização das equações de transporte (EDPs):

- ✓ Método das Diferenças Finitas (MDF);
- ✓ Método dos Volumes Finitos (MVF);
- ✓ Método dos Elementos Finitos (MEF).

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

Segue um resumo de alguns métodos de discretização das equações de transporte (EDPs):

Métodos de Discretização	Descrição	Vantagens	Limitações	Aplicações
<u>Método das Diferenças Finitas (MDF)</u>	<ul style="list-style-type: none"> Aproxima as derivadas das equações diferenciais usando diferenças entre valores da função em pontos discretos da malha; Utiliza uma malha estruturada (pontos alinhados em linhas e colunas). 	<ul style="list-style-type: none"> Simples de implementar; Eficiente para problemas com geometrias simples e malhas estruturadas. 	<ul style="list-style-type: none"> Menos flexível para geometrias complexas; Dificuldade em garantir conservação de grandezas (massa, momento, energia). 	<ul style="list-style-type: none"> Problemas com geometrias simples (ex.: escoamentos em dutos retos).
<u>Método dos Volumes Finitos (MVF)</u>	<ul style="list-style-type: none"> Divide o domínio em volumes de controle (células) e integra as equações de transporte sobre cada volume; Conserva grandezas como massa, momento e energia de forma natural. 	<ul style="list-style-type: none"> Conservação natural de grandezas; Flexível para malhas não estruturadas e geometrias complexas. 	<ul style="list-style-type: none"> Pode ser menos preciso em malhas irregulares; Requer cuidado na discretização dos fluxos nas interfaces. 	<ul style="list-style-type: none"> CFD (escoamentos turbulentos, compressíveis, multifásicos).
<u>Método dos Elementos Finitos (MEF)</u>	<ul style="list-style-type: none"> Divide o domínio em elementos (triângulos, tetraedros, quadriláteros, etc.) e aproxima a solução por funções de base (polinômios) dentro de cada elemento; Utilizado principalmente em problemas estruturais, mas também em CFD. 	<ul style="list-style-type: none"> Precisão em geometrias complexas; Ideal para problemas multifísicos (ex.: acoplamento fluido-estrutura). 	<ul style="list-style-type: none"> Menos eficiente para escoamentos turbulentos ou compressíveis; Dificuldade em garantir conservação de grandezas. 	<ul style="list-style-type: none"> Simulações estruturais, transferência de calor, microfluídica.

1. Visão Geral sobre CFD

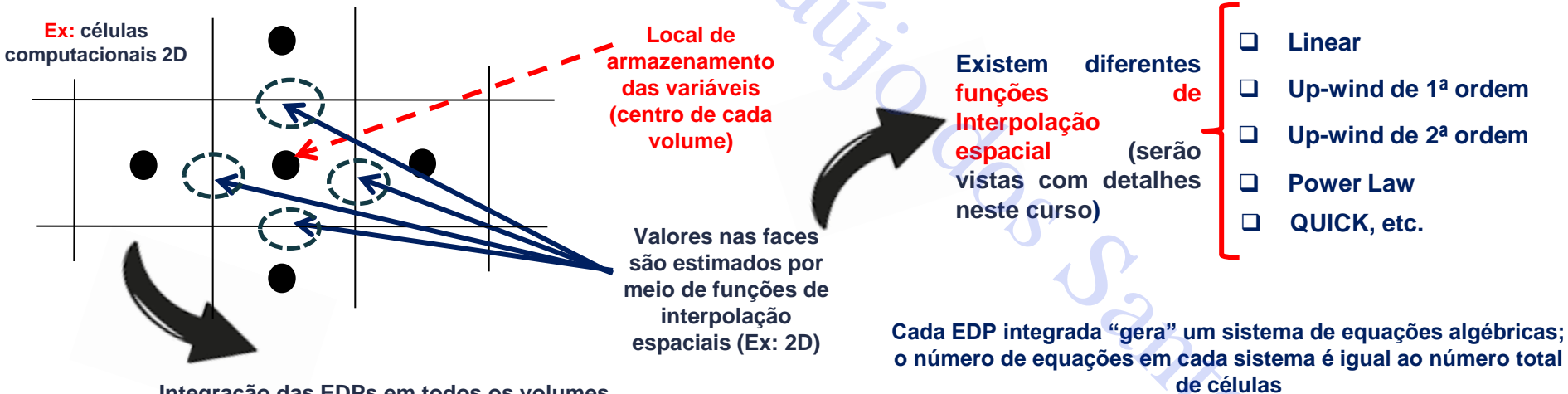
1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.

Observação 6: Para simulações CFD, o Método dos Volumes Finitos (MVF) é o mais utilizado em códigos comerciais e open-source devido à sua capacidade de conservação e flexibilidade. Neste curso, iremos focar com detalhes, nos próximos materiais, no MVF.

No Método dos Volumes Finitos os valores das propriedades transportadas (velocidade, temperatura, concentração etc.) são armazenadas no **centro de cada célula ou volume**. Como a integração das equações (EDPs) é feita através das faces dos volumes, é necessário conhecer estes valores nas faces, o que é feito através das **funções de interpolação espacial**, que serão detalhadas neste curso nos próximos materiais.



Integração das EDPs em todos os volumes.

Ex: Equação da continuidade em estado estacionário

$$\int_S \rho \vec{u} \cdot \vec{n} dS = 0$$

1 EDP \Rightarrow $\left\{ \begin{array}{l} \text{Eq. Algébrica}_1 \\ \text{Eq. Algébrica}_2 \\ \vdots \\ \text{Eq. Algébrica}_n \end{array} \right.$, sendo n o número de células

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.

Observação 7: Em problemas transientes (não estacionários), além da integração das EDPs no espaço, é necessário a integração no tempo. Logo, **funções de interpolação temporal** são usadas para aproximar como as variáveis do escoamento (como velocidade, pressão e temperatura) evoluem no tempo. Esses esquemas temporais determinam a precisão, estabilidade e eficiência da simulação.

Ex: Equação da continuidade em estado transiente

$$\int_{\Delta t} \int_V \frac{\partial \rho}{\partial t} dV dt + \int_{\Delta t} \int_S \rho \vec{u} \cdot \vec{n} dS dt = 0$$

Existem diferentes **funções de Interpolação temporal** (serão vistas com detalhes neste curso)

- Euler Explícito
- Euler Implícito
- Crank-Nicolson
- Adams-Bashforth
- etc.

Observação 8: Desta forma, um código de CFD resolve, de forma numérica, sistemas de equações algébricas. Estes sistemas surgem devido ao fato de que as propriedades armazenadas no centro de um volume em particular, dependerem dos valores destas mesmas propriedades nos centros dos “volumes vizinhos”, originando, assim, uma interdependência entre as equações.

Ex.: Imagine uma malha computacional contendo 1 milhão de células computacionais utilizada para calcular o escoamento turbulento de um fluido newtoniano, não isotérmico, tridimensional (3D) e transiente. Neste caso, se considerarmos as equações de transporte (EDPs) da continuidade, de momentum (sendo três equações, uma para cada direção espacial), da energia e o modelo de turbulência k-ε (modelo de duas equações de transporte), o código teria que resolver, de uma forma geral, sete sistemas de equações algébricas contendo, cada um, 1 milhão de equações, para cada passo de tempo. No total, teríamos **7 milhões de equações**, além de equações constitutivas (equações de estado etc.)

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

Seguem um resumo de alguns métodos utilizados na solução de sistemas de equações algébricas:

Método	Vantagens	Desvantagens	Aplicabilidade
<u>Método de Jacobi</u>	<ul style="list-style-type: none"> • Simples de implementar; • Fácil de paralelizar. 	<ul style="list-style-type: none"> • Convergência lenta; • Requer armazenamento dos valores da iteração anterior. 	<ul style="list-style-type: none"> • Pode ser aplicado a matrizes simétricas e não simétricas; • Usado em problemas onde a simplicidade é mais importante que a eficiência.
<u>Método de Gauss-Seidel</u>	<ul style="list-style-type: none"> • Convergência mais rápida que o método de Jacobi; • Não requer armazenamento adicional dos valores da iteração anterior. 	<ul style="list-style-type: none"> • Difícil de paralelizar; • Ainda pode ser lento para sistemas mal condicionados. 	<ul style="list-style-type: none"> • Pode ser aplicado a matrizes simétricas e não simétricas; • Amplamente utilizado em CFD para sistemas de pressão e velocidade.
<u>TDMA (Método da Matriz Tridiagonal)</u>	<ul style="list-style-type: none"> • Extremamente eficiente para sistemas tridiagonais; • Simples de implementar e direto para sistemas 1D. 	<ul style="list-style-type: none"> • Aplicável apenas a sistemas tridiagonais; • Iterativo para sistemas 2D e 3D. 	<ul style="list-style-type: none"> • Usado em problemas 1D ou em métodos de solução linha-a-linha para problemas 2D/3D; • Aplicável a matrizes simétricas e não simétricas (desde que tridiagonais).
<u>PCG (Método do Gradiente Conjugado Pré-Condicionado)</u>	<ul style="list-style-type: none"> • Eficiente para sistemas grandes e esparsos; • Uso de um pré-condicionador para tratar com sistemas mal condicionados. 	<ul style="list-style-type: none"> • Pode ser custoso computacionalmente; • Escolha do pré-condicionador é crítica e pode ser difícil. 	<ul style="list-style-type: none"> • Aplicável apenas a matrizes simétricas (campo de pressão).
<u>Multigrid</u>	<ul style="list-style-type: none"> • Utiliza múltiplas malhas de diferentes resoluções para acelerar a convergência; • Extremamente eficiente para sistemas grandes. 	<ul style="list-style-type: none"> • Complexidade de implementação; • Requer ajuste cuidadoso dos parâmetros (número de malhas, ciclos, etc.). 	<ul style="list-style-type: none"> • Pode ser aplicado a matrizes simétricas e não simétricas; • Amplamente utilizado em CFD para sistemas de pressão e velocidade.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

Observação 9: Como será visto neste curso, as equações de Navier-Stokes envolvem, tanto a velocidade, quanto a pressão. Em um escoamento compressível, a pressão pode ser relacionado à densidade por uma equação de estado (Ex.: gás ideal). Porém, em um escoamento incompressível, a pressão não está diretamente relacionada a outras variáveis termodinâmicas, então ela deve ser determinada de forma a garantir que a equação da continuidade seja satisfeita. Para resolver o acoplamento pressão-velocidade em escoamentos incompressíveis, são utilizados algoritmos que resolvem as equações de Navier-Stokes de forma iterativa e garantem que o campo de velocidade seja livre de divergência (lembrar que em um escoamento incompressível, o divergente do campo de velocidade é nulo).

Segue um resumo de alguns algoritmos de acoplamento pressão-velocidade:

Método	Descrição	Vantagens	Desvantagens
<u>SIMPLE</u> (Semi-Implicit Method for Pressure-Linked Equations)	<ul style="list-style-type: none"> Desenvolvido por Patankar e Spalding (1972), é o método mais popular para acoplamento pressão-velocidade; Resolve iterativamente as equações de momento e a equação de Poisson para a pressão. 	<ul style="list-style-type: none"> Simples de implementar; Amplamente utilizado e validado. 	<ul style="list-style-type: none"> Convergência lenta para escoamentos complexos; Requer sub-relaxação para garantir estabilidade.
<u>PISO</u> (Pressure Implicit with Splitting of Operators)	<ul style="list-style-type: none"> Desenvolvido por Issa (1986), é um método de correção de pressão em dois passos; Ideal para escoamentos transientes. 	<ul style="list-style-type: none"> Mais rápido que o SIMPLE para escoamentos transientes; Menos dependente da sub-relaxação. 	<ul style="list-style-type: none"> Mais complexo de implementar.
<u>PIMPLE</u>	<ul style="list-style-type: none"> É uma combinação dos métodos PISO e SIMPLE. 	<ul style="list-style-type: none"> Pode ser aplicado a escoamentos transientes, malhas móveis e problemas com grandes gradientes de pressão. 	<ul style="list-style-type: none"> Implementação mais complexa que métodos como SIMPLE ou PISO puros; Requer ajuste cuidadoso dos parâmetros para cada problema.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA SIMULAÇÕES CFD

❑ SOFTWARES COMERCIAIS:

1) ANSYS Fluent

- ✓ **Descrição:** Um dos softwares de CFD mais utilizados no mundo, parte do pacote ANSYS.
- ✓ **Características:**
 - Interface amigável e robusta; Utiliza o Método dos Volumes Finitos;
 - Amplo leque de modelos físicos (escoamento laminar, turbulento, compressível, incompressível, transferência de calor, reações químicas, etc.);
 - Ferramentas avançadas de pós-processamento.
- ✓ **Aplicações:** Automotiva, aeroespacial, energia, HVAC, etc.

2) ANSYS CFX

- ✓ **Descrição:** Outro software da ANSYS, focado em simulações de alta precisão.
- ✓ **Características:**
 - Solver baseado em volumes finitos;
 - Excelente para escoamentos turbulentos e multifásicos;
 - Integração com outros módulos do ANSYS (estrutural, térmico, etc.).
- ✓ **Aplicações:** Turbomáquinas, energia, indústria química.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA SIMULAÇÕES CFD

❑ SOFTWARES COMERCIAIS:

3) STAR-CCM+

- ✓ Descrição: Desenvolvido pela Siemens, é uma ferramenta poderosa e integrada para CFD.
- ✓ Características:
 - Geração de malha, simulação e pós-processamento em uma única plataforma;
 - Suporta simulações multifísicas (CFD, FEA, termodinâmica);
 - Malhas poligonais e poliedrais; Utiliza o Método dos Volumes Finitos.
- ✓ Aplicações: Automotiva, aeroespacial, energia, biomedicina.



4) COMSOL Multiphysics

- ✓ Descrição: Software de simulação multifísica que inclui módulos de CFD.
- ✓ Características:
 - Integração de CFD com outras físicas (estrutural, térmica, elétrica, etc.);
 - Interface baseada em modelos físicos pré-definidos; Utiliza o Método dos Elementos finitos (MEF).
- ✓ Aplicações: Simulações acopladas, microfluídica, transferência de calor.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA SIMULAÇÕES CFD

❑ SOFTWARES OPEN-SOURCE:

1) OpenFOAM

- ✓ Descrição: O principal software open-source para CFD, amplamente utilizado em pesquisa e indústria.
- ✓ Características:
 - Baseado no método dos volumes finitos;
 - Altamente personalizável, com suporte a scripts em C++;
 - Grande variedade de solvers e modelos físicos.
- ✓ Aplicações: escoamentos turbulentos, multifásicos, combustão, etc.

The logo for OpenFOAM, featuring the text "OpenFOAM" in a blue sans-serif font. The letter "O" is replaced by a blue downward-pointing triangle.

2) SU2

- ✓ Descrição: Software open-source focado em simulações aerodinâmicas e otimização.
- ✓ Características:
 - Desenvolvido para problemas de aerodinâmica e propulsão;
 - Suporta malhas estruturadas e não estruturadas;
 - Integração com ferramentas de otimização; Utiliza o Volumes finitos (MVF) e Elementos Finitos (MEF) em alguns módulos.
- ✓ Aplicações: Projetos aeroespaciais, otimização de formas.

The logo for SU2 code, featuring the text "SU2" in a large, bold, red sans-serif font. Below it, the word "code" is written in a smaller, blue sans-serif font.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

B) Processamento: É a etapa em que as equações governantes (ex.: equações de Navier-Stokes) são resolvidas numericamente em cada elemento da malha.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA SIMULAÇÕES CFD

❑ SOFTWARES OPEN-SOURCE:

3) Code Saturne

✓ **Descrição:** Desenvolvido pela EDF (França), é um software open-source para CFD geral.

✓ **Características:**

- Focado em escoamentos incompressíveis e compressíveis;
- Suporta malhas não estruturadas;
- Interface gráfica e scripting disponíveis; Utiliza o Método dos Volumes Finitos.



✓ **Aplicações:** Energia, HVAC, engenharia mecânica.

4) ElmerFEM

✓ **Descrição:** Software open-source multifísico que inclui módulos de CFD.

✓ **Características:**

- Integração de CFD com outras físicas (estrutural, térmica, etc.);
- Interface gráfica e suporte a scripts; Utiliza o Método dos Elementos Finitos (MEF).



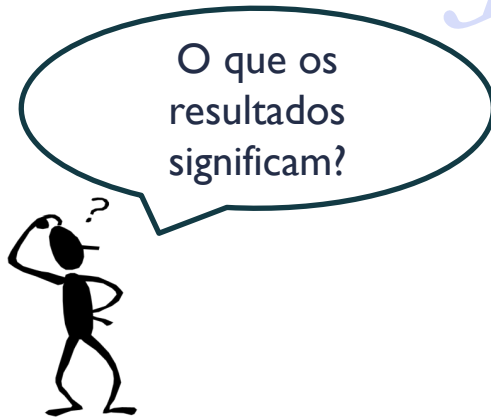
✓ **Aplicações:** Simulações acopladas, microfluídica, transferência de calor.

1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

C) Pós-Processamento: É a etapa de análise e visualização dos resultados obtidos na simulação.

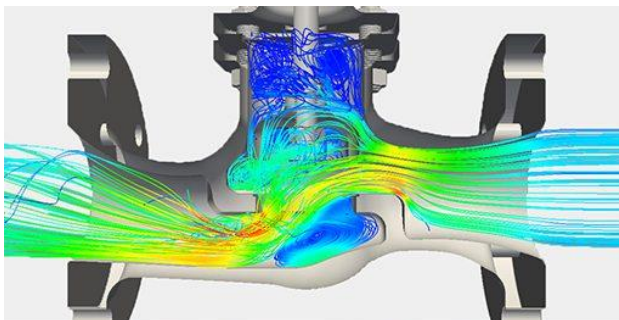


Principais tarefas:

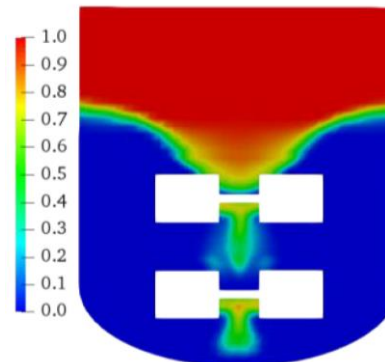
- ❑ **Visualização dos dados:** Gerar gráficos, contornos, vetores, superfícies ou animações para representar o campo de escoamento;
- ❑ **Análise quantitativa:** Calcular grandezas de interesse (ex.: forças, coeficientes de arrasto, transferência de calor);
- ❑ **Validação:** Comparar os resultados com dados experimentais ou analíticos para verificar a precisão da simulação;
- ❑ **Interpretação:** Extrair insights e conclusões sobre o comportamento do fluido.

Seguem alguns exemplos de pós-tratamento de simulações CFD:

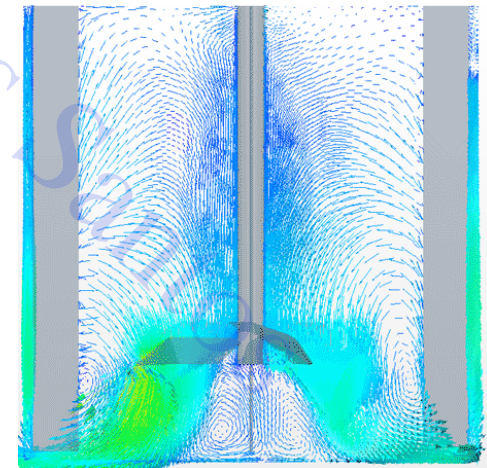
Trajetórias (pathlines)



Mapas de Cores (color map)



Vetores (vectors)



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

C) Pós-Processamento: É a etapa de análise e visualização dos resultados obtidos na simulação.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA PÓS-PROCESSAMENTO

❑ SOFTWARES COMERCIAIS:

1) Tecplot

- ✓ Descrição: Um dos softwares mais populares para pós-processamento de CFD.
- ✓ Características:
 - Visualização 2D e 3D de campos escalares e vetoriais;
 - Geração de gráficos, mapas de contorno, linhas de corrente e isosuperfícies etc.;
 - Suporte a múltiplos formatos de dados (ANSYS, OpenFOAM, STAR-CCM+, etc.).
- ✓ Aplicações: Amplamente utilizado em aeronáutica, automotiva, energia e pesquisa acadêmica.



2) ANSYS CFD-Post

- ✓ Descrição: Ferramenta de pós-processamento integrada ao pacote ANSYS.
- ✓ Características:
 - Visualização de campos escalares e vetoriais;
 - Geração de gráficos, animações e relatórios etc.;
 - Integração direta com os solvers ANSYS Fluent e CFX.
- ✓ Aplicações: Indústria automotiva, aeroespacial, HVAC e energia.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

C) Pós-Processamento: É a etapa de análise e visualização dos resultados obtidos na simulação.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA PÓS-PROCESSAMENTO

❑ SOFTWARES COMERCIAIS:

3) STAR-CCM+ (Simcenter)

- ✓ Descrição: Software de CFD da Siemens que inclui ferramentas de pós-processamento integradas.
- ✓ Características:
 - Visualização 3D de campos escalares e vetoriais;
 - Geração de gráficos, cortes, linhas de corrente e animações etc.;
 - Suporte a simulações multifísicas.
- ✓ Aplicações: Automotiva, aeroespacial, energia, biomedicina.



4) COMSOL Multiphysics (COMSOL Post)

- ✓ Descrição: Software de simulação multifísica com módulo de pós-processamento integrado.
- ✓ Características:
 - Visualização de campos escalares e vetoriais;
 - Geração de gráficos, animações e relatórios etc.
- ✓ Aplicações: Microfluídica, transferência de calor, acústica e eletromagnetismo.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

C) Pós-Processamento: É a etapa de análise e visualização dos resultados obtidos na simulação.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA PÓS-PROCESSAMENTO

❑ SOFTWARES OPEN-SOURCE:

1) ParaView

✓ **Descrição:** Software open-source desenvolvido pela Kitware, amplamente utilizado para visualização científica.

✓ **Características:**

- Visualização 2D e 3D de campos escalares e vetoriais;
- Geração de gráficos, mapas de contorno, etc.
- Suporte a múltiplos formatos de dados (OpenFOAM, VTK, etc.).

✓ **Aplicações:** Pesquisa acadêmica, simulações de CFD e FEA.



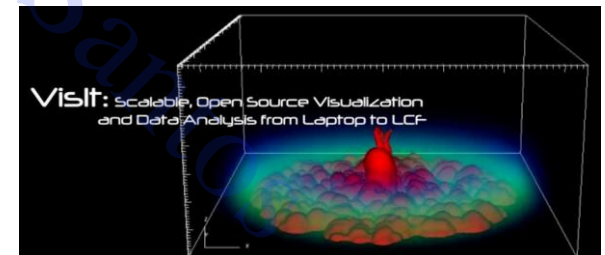
2) VisIt

✓ **Descrição:** Software open-source desenvolvido pelo Lawrence Livermore National Laboratory.

✓ **Características:**

- Visualização 2D e 3D de grandes volumes de dados;
- Geração de gráficos, cortes, isosuperfícies e animações etc.;
- Suporte a múltiplos formatos de dados.

✓ **Aplicações:** Simulações científicas, CFD e análise de dados.



1. Visão Geral sobre CFD

1.4 Estruturação de um código de Fluidodinâmica Computacional

O tratamento de um problema físico por meio de CFD abrange três principais etapas: **A) pré-processamento**; **B) processamento**; e **C) pós-processamento**, resumidas à seguir.

C) Pós-Processamento: É a etapa de análise e visualização dos resultados obtidos na simulação.

PRINCIPAIS SOFTWARES/CÓDIGOS USADOS PARA PÓS-PROCESSAMENTO

❑ SOFTWARES OPEN-SOURCE:

3) Matplotlib (Python)

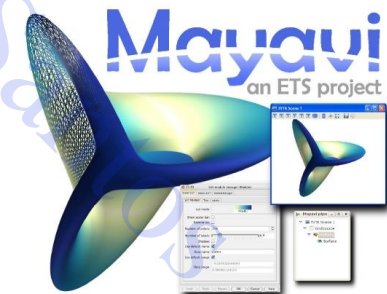
- ✓ **Descrição:** Biblioteca de visualização em Python, amplamente utilizada para criar gráficos e visualizações personalizadas.
- ✓ **Características:**
 - Geração de gráficos 2D e 3D;
 - Visualização de dados escalares e vetoriais;
 - Integração com outras bibliotecas científicas (NumPy, SciPy, etc.).
- ✓ **Aplicações:** Análise de dados, pesquisa acadêmica e prototipagem.



matplotlib

4) Mayavi (Python)

- ✓ **Descrição:** Biblioteca de visualização 3D em Python, baseada no VTK.
- ✓ **Características:**
 - Visualização 3D de campos escalares e vetoriais;
 - Geração de gráficos, isosuperfícies e animações.
- ✓ **Aplicações:** Simulações científicas e CFD.



Bibliografia

VERSTEEG, H.K.; MALALASEKERA, W. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method, 2nd ed. Pearson Education Ltd.: Harlow, UK, 2007.*

MALISKA, C.R. *Transferência de Calor e Mecânica dos Fluidos Computacional, Livros Técnicos e Científicos Editora S/A, 2ª Edição Revista e Ampliada, 2004.*

PATANKAR, S.V. *Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, Taylor & Francis Group. New York, 1980.*