UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

JONAS LAERTE ANSONI

METODOLOGIA PARA PROJETO DE BIORREATORES INDUSTRIAIS
VIA OTIMIZAÇÃO MULTIOBJETIVO COM BASE EM PARÂMETROS
DE DESEMPENHO CALCULADOS POR TÉCNICAS DE CFD

SÃO CARLOS

2015

JONAS LAERTE ANSONI

METODOLOGIA PARA PROJETO DE BIORREATORES INDUSTRIAIS
VIA OTIMIZAÇÃO MULTIOBJETIVO COM BASE EM PARÂMETROS
DE DESEMPENHO CALCULADOS POR TÉCNICAS DE CFD

Tese apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo para a obtenção do título de Doutor em Ciências.

Programa: Engenharia Mecânica

Área de concentração: Térmica e Fluidos

Orientador: Prof. Tit. Paulo Seleglim Júnior

ESTE EXEMPLAR TRATA-SE DA
VERSÃO CORRIGIDA.
A VERSÃO ORIGINAL ENCONTRA-
SE DISPONÍVEL JUNTO AO
DEPARTAMENTO DE
ENGENHARIA MECÂNICA DA
EESC-USP.

São Carlos

-2015-

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

A622m Ansoni, Jonas Laerte
Metodologia para projeto de biorreatores
industriais via otimização multiobjetivo com base em
parâmetros de desempenho calculados por técnicas de CFD
/ Jonas Laerte Ansoni; orientador Paulo Seleghim
Júnior. São Carlos, 2015.

Tese (Doutorado) - Programa de Pós-Graduação em
Engenharia Mecânica e Área de Concentração em Térmica e
Fluídos -- Escola de Engenharia de São Carlos da
Universidade de São Paulo, 2015.

1. Otimização multiobjetivo. 2. CFD. 3.
Biorreatores. 4. Distribuição tempo de residência. 5.
Tensão de cisalhamento. 6. Código aberto. I. Título.

FOLHA DE JULGAMENTO

Candidato: Licenciado JONAS LAERTE ANSONI.

Título da tese: "Metodologia para projeto de bioreatores industriais via otimização multiobjetivo com base em parâmetros de desempenho calculados por técnicas de CFD".

Data da defesa: 21/05/2015

Comissão Julgadora:

Prof. Titular **Paulo Seleglim Junior (Orientador)**
(Escola de Engenharia de São Carlos/EESC)

Prof. Dr. **Luben Cabezas Gómez**
(Escola de Engenharia de São Carlos/EESC)

Prof. Associado **Leandro Franco de Souza**
(Instituto de Ciências Matemáticas e de Computação/ICMC)

Prof. Titular **Julio Romano Meneghini**
(Escola Politécnica/EP-USP)

Prof. Dr. **José Antônio Silveira Gonçalves**
(Universidade Federal de São Carlos/UFSCar)

Resultado:

Aprovado

APROVADO

Aprovado

Aprovado

APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia Mecânica:
Prof. Associado **Gherhardt Ribatski**

Presidente da Comissão de Pós-Graduação:
Prof. Associado **Paulo Sergio Lima Segantine**

*"Aos meus pais,
Laerte (in memoriam) e Lourdes
com todo amor, admiração e gratidão."*

Agradecimentos

Ao meu pai Laerte (*in memoriam*) pelo exemplo de vida.

À minha mãe Lourdes, pela compreensão de minha ausência em muitos momentos, pelo apoio incondicional e por nunca medir esforços para prover aos filhos recursos necessários para que conseguissem estudar.

Às minhas queridas irmãs pelo amor, paciência e incentivo.

A Iara, minha companheira nesta longa jornada, pelo carinho inefável, incentivo e por tornar minha vida cada dia mais feliz.

Ao meu orientador prof. Paulo Seleghim Júnior, pela oportunidade e confiança a mim depositada para a realização desse trabalho.

À todos os meus amigos e colegas de laboratório que fizeram parte desta longa caminhada, em especial para Márcia, Evelise, Luis Enrique, Hugo, Adriana, Daniela, Glauber, Analice, Simone, Ernesto, Fernando e Mogaji. Obrigado pelos bons momentos.

Aos professores e funcionários do departamento de Engenharia Mecânica da Escola de Engenharia de São Carlos pela colaboração.

À CAPES pelo suporte financeiro.

Finalmente a todos que de alguma maneira contribuíram para a conclusão deste trabalho.

Resumo

ANSONI, J. L. **Metodologia para projeto de biorreatores industriais via otimização multiobjetivo com base em parâmetros de desempenho calculados por técnicas de CFD.** 2015. 203 f. Tese (Doutorado) – Departamento de Engenharia Mecânica, Escola de Engenharia de São Carlos da Universidade de São Paulo, São Carlos, 2015.

A crescente demanda por biocombustíveis e a concorrência dos combustíveis fósseis torna necessária a otimização dos meios já existentes e o desenvolvimento de novas tecnologias para produção de biocombustíveis, principalmente em projetos envolvendo biorreatores e fotobiorreatores (FBR) industriais. A dinâmica dos fluidos computacional (CFD) vem sendo utilizada em vários trabalhos para o estudo de parâmetros fluidodinâmicos que podem influenciar no rendimento dos processos químicos envolvidos, como tensão de cisalhamento, perfis de velocidade, tempo de residência e a influência da geometria sobre esses parâmetros. Contudo, não existe ainda um número abrangente de trabalhos que utilize técnicas de otimização acopladas com a resolução numérica do problema fluidodinâmico. Em alguns estudos, algoritmos de otimização são utilizados para determinar os melhores coeficientes das reações químicas. No entanto, não há estudos, até o momento, que reportem a otimização multiobjetivo simultânea dos parâmetros geométricos e do escoamento aplicados a equipamentos da indústria sucro-energética. Neste contexto, o presente trabalho de pesquisa tem como objetivo contribuir para o avanço científico e tecnológico através da implementação de um software aberto (PyCFD-O) que permita o acoplamento CFD-otimização e o desenvolvimento das bases de uma metodologia de projeto otimizado bem como de operação de biorreatores e FBRs de escala industrial. O PyCFD-O foi testado em dois estudos de caso que podem ser estendidos a um fermentador contínuo e um FBR. Os parâmetros geométricos de ambos os reatores foram otimizados de forma a minimizar simultaneamente a tensão de cisalhamento e a variância da distribuição do tempo de residência. O software PyCFD-O mostrou-se robusto, revelando que o processo global de otimização realiza de fato a busca pela fronteira de Pareto. Além da obtenção das geometrias otimizadas, também foram discutidos a influência dos parâmetros geométricos na hidrodinâmica do escoamento em ambos os casos.

Palavras-chave: Otimização multiobjetivo, CFD, Biorreatores, Distribuição tempo de residência, Tensão de cisalhamento, Código aberto.

Abstract

ANSONI, J. L. **Methodology for industrial bioreactor design via multiobjective optimization based on performance parameters calculated by CFD techniques.** 2015. 203 f. Tese (Doutorado) – Departamento de Engenharia Mecânica, Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2015.

The growing demand for biofuels and its competition with fossil fuels create the need to optimize the existing resources and development of new technologies for production of biofuels, particularly in projects involving industrial bioreactors and photobioreactors (PBR). Computational fluid dynamics (CFD) has been used in several studies for the study of fluid dynamics parameters that can influence the performance of the chemical process involved, such as shear stress, velocity profiles, residence time and the influence of these parameters on the reactor geometry. However, there are lacks of studies that utilize optimization techniques coupled with the numerical resolution of the fluid dynamic problem. The use of optimization algorithms has been reported in some cases, but there have not been reports on studies combining the optimization of flow parameters and multiobjective algorithms to choose ideal geometric parameters applied to the equipment of the sugar-energy industry. In this context, this research project aims to contribute to the advancement of scientific and technological knowledge through the implementation of open source software (PyCFD-O) for the CFD-optimization coupling and the development of the bases of a methodology for optimal design and operation of industrial scale bioreactors and PBR. The PyCFD-O software was tested in two case studies with characteristics that can be extended to a continuous fermenter and PBR. The geometric parameters of both reactors were simultaneously optimized in order to minimize the shear stress and the variance of residence time distribution. The PyCFD-O software showed robustness, revealing that overall optimization process actually performs the search of Pareto frontier. In addition to the geometry optimization, the influence of the geometrical parameters of the hydrodynamic of the flow was discussed in both case studies.

Keywords: Multiobjective optimization. CFD. Bioreactor. Residence time distribution. Shear stress. Open source code.

Lista de Figuras

Figura 1.1: Produtividade média de etanol por área para diferentes culturas. Fonte BNDES, 2008.	28
Figura 1.2: Evolução da produção de cana-de-açúcar, etanol e açúcar no Brasil.	30
Figura 1.3: Dornas de fermentação contínua. (Fonte: OSAKI, 2014).....	39
Figura 1.4: Diferentes sistemas de fotobiorreatores: A) Sistema aberto <i>raceway</i> nos EUA; B) Sistema aberto lagoa circular Taiwan; C) Sistema fechado tipo placas panas; D) Sistema fechado tubular.(Fonte: CHISTI, 1989, Wikipedia).	40
Figura 2.1: CFD complementa as análises teóricas e experimentais (fonte: FORTUNA, A. O., 2000).	49
Figura 2.2: Representação gráfica de robustez e precisão de um algoritmo de otimização (maximização).....	60
Figura 2.3: Representação qualitativa da fronteira de Pareto.....	62
Figura 2.4: Esquema do modelo NSGA-II (Adaptado de DEB et al., 2002).	68
Figura 2.5: Metodologia de otimização automática com CFD.....	69
Figura 3.1: Visão geral da estrutura do OpenFOAM (fonte: OpenFOAM <i>User's Guide</i>).....	73
Figura 3.2: Estrutura de diretórios e arquivos necessários para uma simulação com o OpenFOAM.....	74
Figura 3.3: Discretização do domínio da solução. (Adaptado de: OpenFOAM <i>Programmer's Guide</i>).....	81
Figura 3.4: Parâmetros em uma discretização por volumes finitos. (Adaptado de: OpenFOAM <i>Programmer's Guide</i>).....	82
Figura 4.1: Representação esquemática inicial do biorreator.....	88
Figura 4.2: Gráfico dos componentes locais de velocidade.	91
Figura 4.3: Definição do campo da partícula traçadora: <i>createField.H</i>	96
Figura 4.4: Cálculo da equação de transporte para a partícula traçadora: <i> pisoScalarFoamAdjTime</i>	96
Figura 4.5: Exemplo de malha computacional tetraédrica utilizada: Reator contínuo.....	97
Figura 4.6: Representação esquemática do reator <i>airlift</i> : (1) Seção de subida (<i>riser</i>); (2) seção de descida (<i>Downcomer</i>).....	99
Figura 4.7: A) Reator <i>airlift</i> ; B) Câmara para distribuição de ar: (1) Entrada de ar; (2) Válvula purgadora; C) Injetor.....	99
Figura 4.8: Representação esquemática: Bancada experimental. (1) Compressor; (2) tanque de armazenamento de ar; (3) Válvula de abertura de ar comprimido; (4) Válvula de regulação; (5) Rotâmetro.....	100
Figura 4.9: Diagrama esquemático sistema de aquisição pulso de pH: (1) Ponto de injeção traçador; (2) Eletrodo de pH; (3) Termopar; (4) Placas de aquisição; (5) Software.	102
Figura 4.10: Fotografia da bancada experimental pulso pH.....	102
Figura 4.11: Partícula traçadora.....	103
Figura 4.12: Desenho esquemático da bancada experimental: PIV <i>in-house</i> : (1) Câmera de alta velocidade; (2) Lâmpadas de LEDs; (3) Computador.....	105
Figura 4.13: Fotografias PIV <i>in-house</i> : A) Bancada experimental; B) Imagem para análise qualitativa.....	105
Figura 4.14: Desenho esquemático da bancada experimental: PIV LaVision. (1) Câmera CCD; (2) Lente óptica; (3) Laser pulsado duplo; (4) Computador.....	106

Figura 4.15: Fotografias PIV LaVision: A) Bancada experimental; B) Imagem para processamento.	106
Figura 4.16: Perfil de velocidade junto a uma superfície sólida. Fonte: SOUZA et al., 2011	115
Figura 4.17: Turbulência próxima a parede: A) Malha refinada próximo a parede ($y^+ \approx 1$). B) Malha para funções de parede.	116
Figura 4.18: Exemplo de malha computacional e injetores utilizados reator <i>airlift</i>	119
Figura 4.19: Metodologia de otimização automática: PyCFD-O.	121
Figura 4.20: Representação esquemática do paradigma mestre-escrava utilizado no PyCFD-O.	123
Figura 4.21: Processamento paralelo do método de otimização no PyCFD-O.	124
Figura 4.22: Código python para o cálculo da tensão de cisalhamento ponderada.	131
Figura 4.23: Representação esquemática do reator contínuo com as variáveis utilizadas no processo de otimização. A) Caso tangencial. B) Caso perpendicular.	132
Figura 4.24: Representação esquemática do reator <i>airlift</i> com as variáveis utilizadas no processo de otimização.	133
Figura 5.1: Perfis de velocidade U (500s) em um corte transversal na geometria de controle para as diferentes malhas analisadas.	136
Figura 5.2: Pontos de monitoramento e malha analisada (Geometria $H_{out} = 1.0$ e $\alpha = 0^\circ$).	140
Figura 5.3: Perfis de velocidade em x, y e z adquiridos dos pontos de monitoramento: A) Solver estacionário – simpleFoam; B) Solver transiente – pisoScalarFoamAdjTime.	141
Figura 5.4: Análise de desempenho obtido com a paralelização do solver CFD.	142
Figura 5.5: Curvas DTR dos reatores contínuos analisados: A) $H_{out} = 0.1\text{m}$ $\alpha = 0^\circ$ e $H_{out} = 1.3\text{m}$; B) $H_{out} = 0.8\text{m}$ $\alpha = 60^\circ$; C) $H_{out} = 0.1\text{m}$ $\alpha = 90^\circ$; D) $H_{out} = 1.3\text{m}$ $\alpha = 90^\circ$	144
Figura 5.6: Linhas de corrente: A) Geometria $H_{out} = 0.1$ e $\alpha = 0^\circ$; B) Geometria $H_{out} = 0.8$ e $\alpha = 60^\circ$; C) Geometria $H_{out} = 1.3$ e $\alpha = 90^\circ$	146
Figura 5.7: Distribuição do traçador no reator contínuo [500s]: A) Geometria $H_{out} = 0.1$ e $\alpha = 0^\circ$; B) Geometria $H_{out} = 0.8$ e $\alpha = 60^\circ$; C) Geometria $H_{out} = 1.3$ e $\alpha = 90^\circ$	146
Figura 5.8: Distribuição da tensão de cisalhamento ponderada no reator contínuo [500s]: A) Geometria $H_{out} = 0.1$ e $\alpha = 0^\circ$. B) Geometria $H_{out} = 0.8$ e $\alpha = 60^\circ$; C) Geometria $H_{out} = 1.3$ e $\alpha = 90^\circ$	147
Figura 5.9: Fronteiras não dominadas: A) Algoritmo EMOO; B) Algoritmo NSGA-II.	149
Figura 5.10: Comparativo entre as melhores fronteiras não dominadas dos métodos de otimização EMOO e NSGA-II.	150
Figura 5.11: Resultado das melhores avaliações após 10 gerações do algoritmo NSGA-II: Possível fronteira de Pareto e configurações factíveis e infactíveis.	151
Figura 5.12: Comparativo das geometrias otimizadas e as geometrias base no domínio de busca [200s].	151
Figura 5.13: Curvas DTR de três geometrias otimizadas.	152
Figura 5.14: Geometrias otimizadas e linhas de corrente: A) $\alpha = 43,5^\circ$, $H_{out} = 1,15\text{ m}$; B) $\alpha = 62,7^\circ$, $H_{out} = 1,28\text{ m}$; C) $\alpha = 87,3^\circ$, $H_{out} = 1,28\text{ m}$	153
Figura 5.15: Superfície de resposta: σ^2 - Reator contínuo.	155

Figura 5.16: Superfície de resposta: TCT ponderada – Reator contínuo.	155
Figura 5.17: Superfície de resposta: Tempo médio de residência ($\bar{t}$) – Reator contínuo.	155
Figura 5.18: Análise geometria $\alpha = 60^\circ$: A) Linhas de corrente e campo vetorial; B) Vorticidade.....	156
Figura 5.19: Pontos de monitoramento reator <i>airlift</i>	158
Figura 5.20: Perfis de velocidade em x, y e z adquiridos dos pontos de monitoramento em 60s: Reator <i>airlift</i>	159
Figura 5.21: Linhas de corrente reator <i>airlift</i> e região de aquisição do PIV no reator.	160
Figura 5.22: Comparativo PIV - LaVision e CFD do campo de velocidades num plano frontal - Reator <i>airlift</i>	160
Figura 5.23: Comparativo perfil de velocidade entre PIV e CFD - Reator <i>airlift</i>	161
Figura 5.24: Curvas com a concentração do fluido traçador normalizada - Experimental e CFD.....	162
Figura 5.25: Comparação entre CFD e experimental das curvas DTR nos pontos de monitoramento.	164
Figura 5.26: Influência da geometria nas linhas de corrente e localização dos pontos de monitoramento: Reator <i>airlift</i> [25s escoamento].	166
Figura 5.27: Influência da geometria na distribuição do fluido traçador: Reator <i>airlift</i> [25s escoamento].	166
Figura 5.28: Curvas DTR para as diferentes geometrias analisadas – Reator <i>airlift</i>	169
Figura 5.29: Fronteira de soluções não dominadas: Reator <i>airlift</i> (dy/Hz).	171
Figura 5.30: Linhas de correntes das geometrias otimizadas e localização dos pontos de monitoramento – Reator <i>airlift</i>	172
Figura 5.31: Curvas DTR das geometrias otimizadas: A) $dy = 0,120$ m, $Hz = 0,51$ m; B) $dy = 0,030$ m, $Hz = 0,40$ m; C) $dy = 0,054$ m, $Hz = 0,46$ m; D) $dy = 0,060$ m, $Hz = 0,45$ m;	173
Figura 5.32: Superfície de resposta $MA\sigma^2$ - Reator <i>airlift</i>	173
Figura 5.33: Superfície de resposta tensão de cisalhamento ponderada - Reator <i>airlift</i>	174
Figura 6.1: Representação da geometria do problema.	191
Figura 6.2: Fronteiras de Pareto e localização das menores F.A. do problema de otimização: A) EMOO; B) NSGA-II.....	193
Figura 6.3: Comparativo entre as geometrias com as menores F.A. obtidas com os métodos EMOO e NSGA-II.....	194

Lista de Tabelas

Tabela 1.1: Balanço de energia na produção de etanol, com diversas matérias-primas.....	29
Tabela 1.2: Indicadores da evolução tecnológica da indústria sucroalcooleira.....	31
Tabela 2.1: Lista de trabalhos analisados que utilizaram CFD para investigar biorreatores....	51
Tabela 2.2: Lista de trabalhos analisados que utilizaram CFD para investigar FBRs.....	55
Tabela 2.3: Comparação entre algoritmos baseados no gradiente e estocásticos.....	61
Tabela 3.1: Principais palavras chaves usadas no arquivo fvSchemes.....	81
Tabela 3.2: Unidades de medida do OpenFOAM (S.I.)	84
Tabela 4.1: Valores da geometria inicial adotada: Reator contínuo.....	88
Tabela 4.2: Pontos de monitoramento pulso pH.....	102
Tabela 4.3: Especificações da estação de trabalho Monalisa.	124
Tabela 5.1: Características da malha computacional - Reator contínuo.....	136
Tabela 5.2: Média dos erros relativos, em percentual, das malhas analisadas em relação à malha muito fina para as curvas dos perfis de velocidade.....	137
Tabela 5.3: Vazões para as diferentes malhas e erro comparado com valor real.	138
Tabela 5.4: Comparação entre os parâmetros de otimização para as malhas computacionais analisadas – Reator contínuo [500s].	138
Tabela 5.5: Custo computacional das malhas analisadas [500s] – Reator contínuo.	138
Tabela 5.6: Geometrias avaliadas – Reator contínuo.	143
Tabela 5.7: Parâmetros analisados [500s de simulação de escoamento].....	144
Tabela 5.8: Parâmetros geométricos para reator reduzido.	148
Tabela 5.9: Parâmetros dos algoritmos de otimização.	149
Tabela 5.10: Parâmetros de otimização das geometrias otimizadas [200s simulação de escoamento].	152
Tabela 5.11: Características da malha computacional - Reator <i>airlift</i>	157
Tabela 5.12: Custo computacional das malhas analisadas - Reator <i>airlift</i>	157
Tabela 5.13: Pontos de monitoramento reator <i>airlift</i>	158
Tabela 5.14: Tempos de circulação obtidos pelo método da esfera.	163
Tabela 5.15: Geometrias analisadas - Reator <i>airlift</i>	165
Tabela 5.16: Parâmetros analisados: Reator <i>airlift</i> [50s escoamento]	168
Tabela 5.17: Parâmetros de otimização das geometrias otimizadas[50s escoamento] – Reatora <i>airlift</i>	172
Tabela 6.1: Parâmetros dos algoritmos de otimização.	192
Tabela 6.2: Geometrias com as menores F.A.....	193

Lista de Símbolos

Símbolo	Descrição
$CD_{k\omega}$	Parte positiva do termo de difusão cruzada
$C(t)$	Função de concentração da partícula traçadora
D_{Bio}	Diâmetro do biorreator
d_{in}	Diâmetro de entrada do biorreator
d_{out}	Diâmetro de saída
dy	Distância placa em relação à parede
e	Energia específica total
e_M	Energia mecânica
$E(t)$	Função de distribuição de tempos de residência
$E(\theta)$	Função normalizada de distribuição de tempos de residência
F_1, F_2	Funções auxiliares no modelo de turbulência
f_σ	Termo fonte (CSF)
g	Aceleração gravitacional
H_{Bio}	Altura do biorreator
H_{out}	Altura do tubo de saída
H_z	Altura placa reator <i>airlift</i>
k	Energia cinemática turbulenta
k_c	Curvatura da superfície
L_{in}	Comprimento do tubo de entrada
$MA\sigma^2$	Média aritmética das variâncias do DTR
N	Quantidade de pontos da curva do perfil de velocidade
ND	Número total de nós da malha computacional
p	Pressão cinemática
p_d	Pressão modificada
p_x e p_y	Ponto no plano cartesiano xy

q	fluxo de calor
Q	Vazão
Q_e	Volume de fonte de energia
S_{ij}	Taxa de cisalhamento média
t	Tempo
$\bar{t}$	Média do tempo de residência
T	Temperatura
T_p	Tempo de uma partícula fluida
TCT_{pond}	Tensão de cisalhamento total ponderada
U	Vetor velocidade
u_g	Velocidade fase gasosa
u_l	Velocidade fase líquida
u_r	Velocidade relativa
V	Volume
x	Vetor de posição genérico / coordenada espacial
X_j^i	Pontos da curva do perfil de velocidades
X_j^{ref}	Pontos da curva de perfil de velocidades de referência
y^+	Distância adimensional à parede

Letras gregas

α	Variação do ângulo de entrada do biorreator
β_1, β^*	Constantes do modelo de turbulência
γ	Fração de vazio
Γ_k	Variável auxiliar no modelo de turbulência
ε	Taxa de dissipação de energia de turbulenta
μ	Viscosidade do fluido
η	Tensão superficial entre o ar e água
θ	Tempo adimensional
$\bar{\tau}$	Tempo médio de residência adimensional

ρ	Densidade
σ	Tensor de tensões
$\sigma_{k1,2}, \sigma_{\omega1,2}$	Constantes do modelo de turbulência
σ^2	Variância DTR
τ	Tensão de cisalhamento
τ^e	Valor elementar médio da tensão de cisalhamento
τ_t	Tensão de cisalhamento turbulenta
τ_{Total}	Tensão de cisalhamento total
τ_v	Tensão de cisalhamento viscosa
ν	Viscosidade cinemática
ν_t	Viscosidade cinemática turbulenta
Ω	Domínio do problema
ω	Taxa de dissipação turbulenta específica

Lista de Siglas

AG	Algoritmo genético
C	Variável de campo partícula traçadora
CAD	Desenho assistido por computador
CAE	Engenharia assistida por computador
CAPEX	<i>Capital expenditures</i>
CDA	<i>Crowding Distance Assignment</i>
CFD	Dinâmica dos fluidos computacional
CFL	<i>Continuum Surface Force</i>
D	Tensor de tensões
DC	Difusividade partícula traçadora
DNS	<i>Direct numerical simulation</i>
DOE	Desenho experimental
DTR	Distribuição do tempo de residência
EMOO	<i>Evolutionary Multi-Objective Optimization</i>
ER	Erro relativo
FBR	Fotobiorreator
FDM	Método das diferenças finitas
FEM	Método dos elementos finitos
FNS	<i>Fast Non-dominated Sorting</i>
FVM	Método dos volumes finitos
GPU	Unidade de processamento gráfico
LES	<i>Large eddy simulation</i>
MDO	Otimização multidisciplinar de projetos
MER	Média do erro relativo
MOGA	<i>Multiobjective Genetic Algorithm</i>
MPI	<i>Message Passing Interface</i>
MULES	<i>Multidimensional Universal Limiter for Explicit Solution</i>
NETeF	Núcleo de Engenharia Térmica e Fluidos
NPGA	<i>Niched Pareto Genetic Algorithm</i>
NSGA	<i>Non-dominated Sorting Genetic Algorithm</i>
OpenFOAM	Open Field operation and manipulation

OPEX	<i>Operating expenditure</i>
PBS	<i>Portable Batch System</i>
PIV	<i>Particle Image Velocimetry</i>
PISO	Pressure-implicit split-operator
POF	<i>Pareto Optimal Frontier</i>
POO	Programação orientada a objeto
PyCFD-O	<i>Python CFD Optimization</i>
SPEA	<i>Strength Pareto Evolutionary Algorithm</i>
RANS	Reinolds Averaged Navier-Stokes equation
SGI	<i>Silicon Graphics</i>
SIMPLE	<i>Semi-implicit method for pressure-linked</i>
SST	<i>Shear Stress Transport</i>
TCT	Tensão de cisalhamento Total
TRT	Tempo de Residência Teórico
VEGA	<i>Vector Evaluated Genetic Algorithm</i>
VOF	<i>Volume of Fluid</i>

Sumário

CAPÍTULO 1	INTRODUÇÃO	27
1.1	Contexto e Motivação	27
1.2	Objetivos	34
1.3	Grupo de pesquisa	35
1.4	Biorreatores e o processo de fermentação	37
1.5	Microalgas e fotobiorreatores	39
1.6	Por que utilizar software livre?	41
1.7	Organização do texto.....	43
CAPÍTULO 2	CONCEITOS FUNDAMENTAIS E REVISÃO DA LITERATURA...45	
2.1	Mecânica dos fluidos computacional	45
2.2	Avanços do uso de CFD para caracterizar, projetar e otimizar sistemas de bioenergia	49
2.2.1	CFD e o estudo de biorreatores	50
2.2.2	CFD e o estudo de fotobiorreatores	54
2.3	CFD com otimização multidisciplinar de projetos (MDO).....	58
2.3.1	Definições básicas na otimização automática	59
2.3.2	Algoritmos genéticos aplicados à otimização multiobjetivo	62
2.3.2.1	Algoritmo NSGA-II	64
2.4	CFD no processo de otimização de projetos	68
2.5	Conclusão	69
CAPÍTULO 3	PACOTE CFD - OPENFOAM	71
3.1	O surgimento do OpenFOAM.....	71
3.2	O OpenFOAM a nível de usuário	72
3.2.1	Estrutura básica para efetuar uma simulação no OpenFOAM.....	73
3.2.2	Pré-processamento	74
3.2.3	Resolução Numérica – Solver	77
3.2.4	Pós-processamento	77
3.3	OpenFOAM a nível de usuário desenvolvedor	78
3.3.1	Orientação a objetos e C++	78
3.3.2	Desenvolvendo códigos no OpenFOAM	79
3.4	Unidades dimensionais das propriedades físicas no OpenFOAM	83
CAPÍTULO 4	METODOLOGIA	85
4.1	Reator contínuo	85
4.1.1	Escolha de uma geometria inicial representativa	86
4.1.2	Simulação numérica do escoamento interno	88

4.1.2.1	Discretização das equações governantes	88
4.1.2.2	Modelagem da turbulência	90
4.1.2.3	Discretização em volumes finitos	93
4.1.2.4	Solver desenvolvido no OpenFOAM para a simulação computacional do reator contínuo	94
4.1.2.5	Malha computacional – Reator contínuo	97
4.2	Reator <i>Airlift</i>	98
4.2.1	Bancada experimental	98
4.2.2	Estudos hidrodinâmicos realizados no reator <i>airlift</i>	100
4.2.2.1	Transferência de movimento: Tempo de mistura e circulação	100
4.2.2.2	Determinação dos campos vetoriais de velocidade	103
4.2.3	Simulação numérica do escoamento	107
4.2.3.1	Discretização das equações governantes	107
4.2.3.2	Modelagem da turbulência	112
4.2.3.3	Tratamento da região de parede	114
4.2.3.4	Solver desenvolvido no OpenFOAM para a simulação computacional do reator <i>airlift</i>	117
4.2.3.5	Malha computacional - reator <i>airlift</i>	117
4.3	Software PyCFD-O	119
4.3.1	Estrutura do programa	119
4.3.2	Métodos de otimização multiobjetivo implementados	122
4.3.3	Paralelização	122
4.4	Acoplamento CFD – Otimização software PyCFD-O	125
4.4.1	Processo de otimização do projeto	125
4.4.2	Cálculo do DTR	125
4.4.3	Cálculo da tensão de cisalhamento	128
4.4.4	Parâmetros geométricos de otimização e funções objetivos: Reator contínuo	131
4.4.5	Parâmetros geométricos de otimização e funções objetivos: Reator <i>airlift</i>	133

CAPÍTULO 5 RESULTADOS E ANÁLISE..... 135

5.1	Reator operado em regime contínuo	135
5.1.1	Estudo de convergência de malha	135
5.1.2	Tempo de simulação necessário para estabelecimento do regime permanente	139
5.1.3	Desempenho computacional obtido com as estratégias de paralelização ..	141
5.1.4	Análise da influência da geometria na hidrodinâmica do escoamento	143
5.1.5	Avaliação dos métodos de otimização NSGA-II e EMOO	147
5.1.6	Dominância de Pareto e geometrias otimizadas – Reator contínuo (1m ³)	150
5.2	Reator <i>airlift</i>	156
5.2.1	Análise da malha computacional	157
5.2.2	Tempo de simulação necessário para estabelecimento do regime permanente – Reator <i>airlift</i>	158
5.2.3	Perfil de velocidade – Comparativo PIV – CFD	159

5.2.4	Tempo de circulação e mistura	161
5.2.5	Análise da influência da geometria na hidrodinâmica do escoamento	164
5.2.6	Dominância de Pareto e geometrias otimizadas – Reator <i>airlift</i>	170
CAPÍTULO 6	CONCLUSÕES E TRABALHOS FUTUROS	175
6.1	Sugestões para trabalhos futuros	178
	REFERÊNCIAS BIBLIOGRÁFICAS	179
APÊNDICE A	Exemplo de otimização multiobjetivo	191
APÊNDICE B	Código fonte solver pisoScalarFoamAdjTime.....	195
APÊNDICE C	Código fonte solver interDyScalarFoam.....	201

CAPÍTULO 1 INTRODUÇÃO

1.1 Contexto e Motivação

A demanda global por biocombustíveis aumentou consideravelmente na última década, impulsionada inicialmente pelo aumento do preço do petróleo e a necessidade de maior segurança energética. Reconhecendo o potencial de desenvolvimento dos biocombustíveis para reduzir a dependência de combustíveis fósseis e gerar menos danos ambientais, vários países passaram a estimular o desenvolvimento de novas tecnologias para a geração de biocombustíveis. Nesse cenário, o Brasil vem se destacando com o programa de etanol de cana-de-açúcar, impulsionado pelas boas condições climáticas e às áreas agrícolas disponíveis. Dessa forma, o programa brasileiro de produção de etanol de cana-de-açúcar vem apresentando resultados interessantes, desde a pesquisa de variedades de cana de maior rendimento até a fabricação de motores que funcionam com qualquer mistura de gasolina e etanol (BNDES, 2008).

O Brasil e os EUA são os grandes produtores mundiais de etanol, porém a produção norte-americana se baseia no biocombustível originado do milho. Na Europa, o etanol também é produzido, no entanto, em escalas bem menores. O biocombustível produzido nesses países é originado principalmente do trigo e da beterraba.

Nos últimos anos, contudo, o ceticismo sobre o impacto positivo dos biocombustíveis tem enfrentado o que os especialistas chamam de *trade-off*¹ entre alimento e biocombustíveis devido ao seu impacto sobre os mercados agrícolas globais. Este fato foi bastante discutido na crise do preço dos alimentos entre os anos de 2007-2008 (AL-RIFFAI; DIMARANAN; LABORDE, 2010, HERTEL; TYNER; BIRUR, 2010). Além disso, vários estudos têm levantado sérias preocupações sobre o impacto ambiental negativo das consequências não intencionais da produção de biocombustíveis, particularmente devido ao impacto da mudança direta e indireta do uso do solo liberar grandes quantidades de carbono (SEARCHINGER et

¹ Trade-off (solução de compromisso): expressão que define uma situação de conflito de escolha. Ele se caracteriza em uma ação econômica que visa a resolução de um problema, mas acarreta outro.

al., 2008). As emissões de carbono são ocasionadas devido à conversão de florestas e terras virgens em lavouras motivadas pela expansão das fronteiras agrícolas.

Diante do paradoxo de se aumentar a utilização de biocombustíveis sem comprometer a produção de alimentos e ampliação de áreas cultivadas torna-se importante o aprimoramento e o desenvolvimento de novas tecnologias que tirem o melhor proveito da matéria-prima existente e da reutilização de seus subprodutos gerados em algumas etapas do processo. O desenvolvimento tecnológico nesta área também garantirá ao Brasil a sua permanência como um dos líderes mundiais no desenvolvimento de tecnologias no setor de biocombustíveis.

Uma das vantagens da produção de etanol a partir da cana-de-açúcar pode ser vista na Figura 1.1. Essa figura evidencia como podem ser diferentes os índices de produtividade por unidade de área cultivada. Os resultados apresentados na Figura 1.1, correspondem à lavouras de boa produtividade e as tecnologias para a conversão de açúcares e amido em etanol maduras e disponíveis (BNDES, 2008). Nesta figura observa-se também o ganho da produtividade do etanol com o aproveitamento da parte celulósica da cana a partir da hidrólise de materiais lignocelulósicos.

Figura 1.1: Produtividade média de etanol por área para diferentes culturas. Fonte BNDES, 2008.

Outros dois fatores que destacam o etanol de cana-de-açúcar são: o balanço energético e consequentemente o custo de produção. O balanço energético é dado pela razão entre a energia renovável contida no biocombustível e a energia fóssil utilizada para produzi-lo. O etanol de cana-de-açúcar apresenta um valor do balanço energético de 8,9 no Brasil (Tabela 1.1). Comparando-se com o etanol de milho (EUA), cujo valor é de 1,3, observa-se uma diferença a favor do etanol de cana-de-açúcar de aproximadamente sete vezes. Isto se reflete no custo de produção do etanol. O custo da produção de etanol, a partir de outras fontes como

o trigo e o milho chega a ser entre duas e quatro vezes mais elevadas. Segundo ANDRIETTA; STECKELBERG; ANDRIETTA, 2006, o custo de produção do etanol brasileiro é de aproximadamente US\$0,17/L contra US\$ 0,32/L para o combustível produzido pelos EUA, valor sob influência dos fortes subsídios do governo norte-americano.

Tabela 1.1: Balanço de energia na produção de etanol, com diversas matérias-primas.

Matéria-prima	Energia renovável/energia fóssil
Etanol de milho (EUA)	1,3
Etanol de cana-de-açúcar (Brasil)	8,9
Etanol de beterraba (Alemanha)	2,0
Etanol de sorgo sacarino (África)	4,0
Etanol de trigo (Europa)	2,0
Etanol de mandioca	1,0

(Fonte: MACEDO, 2007)

A posição de destaque na produção de etanol alcançado no Brasil se deve às melhorias alcançadas nos últimos 40 anos. Melhorias estas que ocorreram principalmente no aumento da produtividade da cana-de-açúcar, pela extensão do ciclo produtivo, otimização do setor de transportes, mecanização da colheita, aumento no rendimento do etanol por quantidade de cana-de-açúcar processada e melhor uso do bagaço nos sistemas de cogeração.

O setor sucroalcooleiro tem apresentado aumentos sucessivos de demanda de produção devidos, principalmente, a três fatores simultâneos: 1) aumento do consumo de etanol no mercado interno provocado pela crescente venda de veículos *flex-fuel*, 2) potencial aumento do mercado externo, devido às preocupações com o aquecimento global levando à utilização de energia renováveis e 3) aumento do consumo de açúcar pelo mercado externo (OSAKI, 2014).

A partir de 2003, com o advento dos carros *flex-fuel* formou-se um grande mercado consumidor para o etanol, abrindo-se novas perspectivas para a expansão da agroindústria da cana no Brasil. Desde então, a agroindústria canavieira brasileira tem se expandido a taxas elevadas, consolidando-se economicamente². Essa grande expansão da produção de etanol nas últimas décadas pode ser observada na Figura 1.2. No entanto, a produção de etanol, fica a mercê de especulações de mercados, isto é, se o preço do açúcar está alto a produção de etanol é diminuída. A produção de etanol também é extremamente dependente de condições

² Hoje esta em crise conjuntural, devendo melhorar com a recomposição do preço da gasolina.

climáticas, como pode ser observado na Figura 1.2. A safra 2011/12 apresentou uma grande queda devido à queda de produção de cana-de-açúcar ocasionada pela seca.

Figura 1.2: Evolução da produção de cana-de-açúcar, etanol e açúcar no Brasil.

É importante destacar que a expansão da produção de etanol e açúcar nas últimas décadas ocorreu não apenas com o aumento da área cultivada, mas também com expressivos ganhos de produtividade nas fases agrícola e industrial. Esses ganhos foram alcançados, exclusivamente, a partir de inovações incrementais, instalação de uma série de equipamentos periféricos e de novos procedimentos operativos (ABARCA, 2005), como por exemplo, a utilização de caminhões de uma, duas ou três caçambas para o transporte da cana-de-açúcar.

A Tabela 1.2 apresenta alguns dos indicadores da evolução tecnológica na área industrial do complexo sucroalcooleiro, entre 1975 e 2012. Observando os dados, vemos que, durante as duas últimas décadas praticamente não houve grande aumento na eficiência dos equipamentos. No entanto, devido às grandes escalas de processamento, normalmente centenas de toneladas por hora, uma pequena melhoria causaria impacto significativo no resultado final na produção de etanol e açúcar.

Mesmo com esse aumento na produção de etanol, segundo projeções da União da Indústria de Cana-de-açúcar (UNICA, 2012) em 2020 é prevista uma deficiência na produção de combustíveis (etanol, gasolina e GNV) para o mercado de combustíveis do ciclo Otto no Brasil de aproximadamente 20 bilhões de litros de combustíveis por ano, sendo que atualmente há um déficit de 6 bilhões de litros por ano. Esse déficit de 20 bilhões de litros

implicaria em gastos na importação de aproximadamente de 6 bilhões de reais se mantidos os preços da gasolina importada pelo Brasil (média de janeiro a julho de 2012) (UNICA, 2012).

Tabela 1.2: Indicadores da evolução tecnológica da indústria sucroalcooleira.

Variável	Tecnologia	1975	1994	2008	2012
Capacidade de moagem (TCD)	DH1/MCD01	5500	11000	13000	13000
Tempo de fermentação (h)	CODISTIL Ferm. Bat./Cont.	12	4/6	4/6	4/6
Teor alcoólico do vinho FOGLER,	CODISTIL Fermentação	7,5	10,5	10	10
Rendimento extração (% açúcar cana)	DH1/MCD01/Difusor	93	97	97	97
Redimento fermentativo (%)	CODISTIL Ferm. Bat./Cont.	80	91	91	91
Rendimento destilação (%)	Destiltech	98	99	99,5	99,5
Rendimento Total ($L_{álcool\ hid}/t_{cana}$)	Tecnologia CODISTIL	66	87	91	91
Bagaço excedente (%)	Tecnologia CODISTIL	Até 8	Até 78	Até 78	-

(Fonte: ABARCA, 2005 e CLEALCO, 2012)

Uma alternativa para expandir a produção de etanol sem o aumento da área plantada poderia ser o aproveitamento do bagaço e palha para a geração do chamado etanol de segunda geração (DIAS et al., 2011). Segundo SILVA, 2010 a utilização da biomassa da cana-de-açúcar é uma alternativa de combustíveis para o futuro do Brasil. Isso porque corresponderia a uma solução cuja base já é consolidada, técnica e economicamente, na qual o Brasil é pioneiro e líder mundial e também porque as outras alternativas de biocombustíveis não se comparam com o etanol em volume de produção atual ou possibilidades futuras.

Atualmente, a tecnologia para a produção do chamado etanol de segunda geração (2G), isto é, a partir do uso dos materiais lignocelulósicos e outros resíduos estão em fase de estudo e são produzidos em modelos experimentais. Através dos editais PAISS³ do Banco Nacional de Desenvolvimento Econômico e Social (BNDES) e a Financiadora de Estudos e Projetos (FINEP) o Brasil vem alocando cada vez mais recursos no desenvolvimento e aperfeiçoamento de novas tecnologias para tornar viável a fabricação do etanol 2G de cana-de-açúcar.

Um exemplo desses financiamentos foi o realizado pela GranBio. A GranBio é uma empresa de biotecnologia 100% brasileira que anunciou em 2013 a primeira fábrica de etanol celulósico em escala comercial do hemisfério Sul no município de São Miguel dos Campos

³ Plano Conjunto BNDES-Finep de Apoio à Inovação Tecnológica Industrial dos Setores Sucroenergético e Sucroquímico – [PAISS](#)

no estado de Alagoas. A unidade entrou em operação em setembro de 2014 e segundo a empresa tem capacidade para produzir 82 milhões de litros do biocombustível por ano (GRANBIO, 2014).

Outro subproduto originado da produção do etanol a partir da cana-de-açúcar que está em plena fase de estudo e que pode trazer grandes avanços tanto na produção de biocombustíveis e/ou compostos bioquímicos de interesse comercial é a vinhaça. A vinhaça é um resíduo líquido da destilação do caldo de cana-de-açúcar fermentado. Para cada litro de etanol produzido, 10 a 18 litros de vinhaça são deixados como resíduos (SEABRA, 2008 RAMIREZ; FARENZENA; TRIERWEILER, 2014). Dada sua natureza orgânica e baixa presença de contaminantes, metais ou outros compostos indesejáveis, a ideia de aplicá-la no solo é considerada. A vinhaça é rica em potássio e enxofre e apresenta quantidades consideráveis de fósforo, nitrogênio, cálcio, magnésio, além dos micronutrientes. Todavia, por apresentar pH muito ácido é necessário ter cautela na sua utilização como fertilizante. A recuperação desses nutrientes é de grande interesse para a sustentabilidade da indústria sucroenergética para a recuperação do solo. No entanto, estudos mostraram que a irrigação do solo com a vinhaça pode contaminar as águas subterrâneas, limitando o seu uso (BÜCHLER, 1987). Uma alternativa é o depósito da vinhaça em poços profundos, mas o seu armazenamento é limitado. Órgãos ambientais locais brasileiros, como CETESB estabeleceram normas para a descarga adequada de vinhaça (CETESB, 2014). Existem outros métodos, como a evaporação para produzir alimentos de origem animal e a incineração para a recuperação de potássio. No entanto, estes tratamentos requerem custos elevados (NAVARRO; DEL C. SEPÚLVEDA; RUBIO, 2000). Dada a importância do etanol para a economia nacional e as restrições ao uso da vinhaça, novas alternativas devem ser avaliadas. Neste contexto, a biorremediação utilizando microalgas pode ser uma alternativa promissora.

As microalgas têm sido utilizadas em uma escala global para diferentes fins. Além de seu alto teor de lipídios e carboidratos, as microalgas são consideradas excelente matéria-prima para a produção de biocombustíveis. As microalgas têm outras vantagens, tais como o rápido crescimento, a não concorrência com cultivos alimentares e a alta eficiência na fixação do CO₂. Devido à sua composição, as microalgas podem ser utilizadas para as mais variadas aplicações, tais como: obtenção de pigmentos, como suplemento nutricional (BRENNAN; OWENDE, 2010); tratamento de efluentes industriais e de águas residuais (DORIA et al., 2012; OLIVEIRA, 2013).

Muitos produtos derivados destes micro-organismos, tais como biocombustíveis, ainda apresentam custos elevados devido à necessidade de nutrientes para o crescimento da

biomassa, entre outras razões (RAMIREZ; FARENZENA; TRIERWEILER, 2014). Assim, a vinhaça surge como uma fonte de nutrientes, a fim de fornecer uma alternativa promissora para o cultivo de microalgas contribuindo para a resolução do descarte dos subprodutos de destilarias.

Usando-se vinhaça pré-tratada a partir de um processo anaeróbio, ESPAÑA-GAMBOA et al., 2011 determinaram a presença de condições favoráveis para o crescimento de microalgas. Vários trabalhos encontrados na literatura relacionam o tratamento de vinhaça para o cultivo de diferentes tipos de microalgas (BARROCAL et al., 2010; MARQUES et al., 2013 BICHARA, 2014; RAMIREZ; FARENZENA; TRIERWEILER, 2014). O cultivo é feito utilizando energia solar em fotobiorreatores e a microalga converte o dióxido de carbono e nutrientes em biomassa. Para isso é necessário o desenvolvimento de um fotobiorreator para a produção de altas concentrações de microalgas em escala industrial.

Deste modo, segundo Cortez et al., 2010, a indústria da cana-etanol não se limita ou se restringe à alternativa do etanol substituindo a gasolina nacional. É uma opção que abre um leque de oportunidades para o país, desde o mercado interno até o mercado externo. Em todos os casos, contudo, existe a necessidade de pesquisas para viabilizar um novo modelo agrícola e industrial, mais sustentável e eficiente, com uso integral dos recursos da cana-de-açúcar. Assim, o conceito de biorrefinaria surge como novo paradigma agroindustrial no qual a biomassa é cuidadosamente desconstruída para produzir tanto produtos de baixo-valor/alto-volume, como combustíveis líquidos (biodiesel e etanol) quanto produtos de baixo-volume/alto-valor, como os compostos farmacêuticos e nutricionais (OSAKI, 2014).

É neste contexto que o presente trabalho está inserido, busca-se contribuir para o desenvolvimento de novas tecnologias e otimização dos meios de produção do etanol de cana-de-açúcar e o reaproveitamento dos seus subprodutos. Durante os últimos anos várias pesquisas vêm sendo feitas para melhorar as reações químicas e biológicas que ocorrem nos biorreatores de uma usina. Entretanto, pesquisas que envolvam a otimização dos parâmetros geométricos associados com os aspectos fluidodinâmicos, no que diz respeito a produção de biocombustíveis a partir de cana-de-açúcar e no estudo do acoplamento bio-foto-fluidodinâmico em fotobiorreatores são praticamente inexistentes.

A otimização dos meios já existentes e aqueles ainda em desenvolvimento pode não somente aumentar a eficiência de produção, mas também reduzir os impactos ambientais causados pelo modelo de produção utilizado atualmente. A produção de biocombustíveis em larga escala e de “insumos verdes” (bioplásticos, biossolventes, etc.) requer o desenvolvimento de biorreatores de alta eficiência e produtividade específicos para as

condições do processo. Um dos focos de ineficiência na produção de biocombustíveis incide sobre o processo de conversão de açúcares em etanol, ou na transesterificação na produção de biodiesel. Especificamente na produção de etanol de primeira geração, os fermentadores industriais mais eficientes ainda estão muito abaixo dos limites estequiométricos, sobretudo quando os volumes envolvidos são da ordem de centenas de metros cúbicos.

O projeto de biorreatores também deve incluir ciclos de otimização nos quais sua geometria e demais parâmetros sejam alterados de acordo com um procedimento que permita estabelecer uma menor dispersão do tempo de residência, variação do campo de temperaturas, e assim por diante. Os parâmetros do escoamento influenciam diretamente no fenômeno químico e, conseqüentemente, na eficiência do biorreator. Sendo assim, é possível concluir que a otimização do processo de produção de etanol consiste em um trabalho multidisciplinar, onde diversos fatores devem ser analisados cuidadosamente e em conjunto.

Para conseguir melhorar a qualidade dos produtos, diminuir o desperdício e obter as propriedades desejadas optam-se cada vez mais pelas simulações computacionais (BITOG et al., 2011; PATWARDHAN et al., 2005; WU, 2012) antes da execução de um projeto ou para melhorias das instalações já existentes ao invés de se usar os antigos protótipos de bancada que consomem mais tempo e envolvem maiores custos. Assim, não restam dúvidas da forte tendência de se usar cada vez mais a mecânica dos fluidos computacional em aplicações industriais.

1.2 Objetivos

O presente trabalho tem como objetivo geral o desenvolvimento de um software de código aberto que permita o acoplamento de modelos CFD a modelos de otimização, a fim de possibilitar a escolha de projetos ótimos de biorreatores industriais e fotobiorreatores de grande escala. Como objetivo específico pretende-se contribuir para o avanço científico e tecnológico com o desenvolvimento de metodologias especializadas e mapeamento das características dos problemas de otimização associados ao projeto de biorreatores e fotobiorreatores de alto desempenho. Para atingir tal objetivo as seguintes metas específicas foram cumpridas:

- Estudo e revisão de biorreatores, fotobiorreatores, CFD e métodos de otimização multiobjetivo;
- Estudo de uma ferramenta de fluidodinâmica computacional em código aberto, visando à implementação e extensão dos códigos existentes;

- Revisão dos trabalhos de otimização com CFD aplicados em biorreatores e fotobiorreatores adotados na literatura;
- Desenvolvimento de uma metodologia computacional de otimização a partir da minimização de funções objetivo convenientemente definidas (desvio-padrão dos tempos de residência, tensão de cisalhamento, etc.);
- Estudo de técnicas e métodos de otimização;
- Realização de estudos baseados em testes extensivos para a determinação da topologia das funções de otimização;
- Aplicação de uma técnica de processamento paralelo em arquitetura de memória compartilhada e distribuída;
- Administração e gerenciamento de um cluster SGI Altix para a execução das simulações computacionais.
- Montagem de bancada experimental de um reator *airlift* para validação qualitativa das simulações CFD bifásicas;
- Avaliação e análise dos resultados.

O desenvolvimento deste trabalho deixa para a comunidade científica uma ferramenta acessível para otimização multidisciplinar, na qual as simulações do escoamento são realizadas através do software gratuito OpenFOAM e o estudo de estratégias de otimização. O fato desta ferramenta ter sido desenvolvida em código aberto permite a sua futura manipulação e possível aperfeiçoamento e ampliação do código por outros pesquisadores tanto na parte de modelagem do escoamento como nas metodologias de otimização.

O desenvolvimento de técnicas capazes de alcançar a otimização com precisão suficiente em tempos computacionais razoáveis terá um grande impacto no projeto e na operação de biorreatores de alto desempenho e, conseqüentemente, no aumento de sua eficiência operacional. Com isso pretende-se desenvolver as bases de uma metodologia de projeto otimizado bem como de operação de biorreatores de escala industrial.

1.3 Grupo de pesquisa

O Núcleo de Engenharia Térmica e Fluidos (NETeF) é um laboratório do programa de pós graduação em Engenharia Mecânica da EESC/USP. As linhas de pesquisas desenvolvidas no grupo incluem: escoamento multifásicos industriais, motores de combustão interna, máquinas de refrigeração, instrumentação e controle, mecânica dos fluidos computacional e bioenergia. As linhas de pesquisa desenvolvidas englobam estudos avançados que visam a

elaboração de teses de mestrado e doutorado e possibilitam o desenvolvimento de projetos científicos-tecnológicos.

Recursos obtidos de projetos de pesquisa permitiram nuclear uma nova área de pesquisa dedicada aos estudos de escoamentos industriais e suas aplicações na área de energias renováveis, em particular o desenvolvimento de biorreatores industriais. Os projetos são:

- “Otimização de biorreatores de estado semi-sólido contínuos através da modelagem do escoamento trifásico interno”, projeto regular FAPESP Proc. 2010/00442-5, vigência: 01/09/2010 a 31/08/2012, coordenador: Paulo Seleglim Jr. (EESC/USP);
- “Produção de biodiesel de microalgas em escala industrial”, consórcio FP7 Brasil/Europa, financiamento FAPESP/CNPq, vigência 01/05/2011 a 01/05/2015, coordenador: Armando Augusto H. Vieira e Paulo Seleglim Jr.
- “Desenvolvimento de um sistema de limpeza a seco, separação e transporte pneumático de cana-energia colhida de forma integral e de seus sistemas de pré e pós-processamento”, projeto selecionado no edital PAISS-AGRÍCOLA/BNDES, vigência 48 meses, coordenadores: Manuel Carnaúba (BioFlex/GranBio), Gonçalo A. G. Pereira (GranBio/BioCelere) e Paulo Seleglim Jr. (EESC/USP);

O trabalho mais recente desenvolvido pelo grupo de pesquisa envolvendo o uso de ferramentas de CFD para a otimização de biorreatores foi o de GÓIS, 2012. GÓIS, 2012 iniciou os estudos para a otimização mono-objetivo dos parâmetros geométricos ideais de fermentadores contínuos na produção de etanol, de forma a minimizar a tensão de cisalhamento ou a variância da distribuição dos tempos de residência. Neste trabalho foram utilizados os softwares computacionais Ansys CFX para a realização da simulação computacional do escoamento e o software modeFRONTIER para o acoplamento com o método de otimização. O presente trabalho de doutorado é proposto como sequencia natural à evolução da pesquisa do grupo referente ao campo de simulação e otimização computacional.

Como exemplo de aplicações futuras das ferramentas computacionais desenvolvidas neste trabalho (PyCFD-O) pode-se citar o trabalho de doutorado iniciado em 2015 no programa integrado de pós-graduação em bionergia (ESALQ/USP) intitulado "Aplicação de técnicas de otimização multiobjetivos ao projeto de reatores industriais de pré-tratamento com base em parâmetros de desempenho calculados por simulação numérica".

1.4 Biorreatores e o processo de fermentação

Segundo SCHMIDELL et al., 2001, denominam-se biorreatores, reatores químicos ou reatores biológicos, os vasos de reação nos quais ocorre uma série de transformações químicas intermediadas por microrganismos uni ou pluricelulares. Essas reações químicas são catalisadas por biocatalisadores, que podem ser enzimas ou células vivas.

A partir de 1950, os biorreatores passaram por um grande avanço e juntamente com o desenvolvimento da microbiologia foram responsáveis pelo sucesso de muitos processos fermentativos. Vários tipos de biorreatores têm sido desenvolvidos e utilizados nos mais diversos tipos de aplicações. Estes biorreatores podem ser classificados de acordo com o sistema operacional:

1. Processo em batelada: caracteriza-se por ser um processo descontínuo simples com a seguinte metodologia: prepara-se uma cultura, coloca-se este meio de cultura no biorreator (fermentador), adiciona-se o microrganismo responsável pelo processo biológico, retira-se o caldo fermentado e executam-se as operações para a recuperação do produto (SCHMIDELL et al., 2001). Este processo apresenta facilidades de manutenção e assepsia e também tem uma flexibilidade de operação, devido ao fato de poder utilizar os fermentadores para a fabricação de diferentes produtos. No entanto, a fermentação descontínua pode levar a baixos rendimentos e/ou produtividades, quando o substrato adicionado no início da fermentação exercer efeitos de inibição, repressão ou desviar o metabolismo celular para produtos que não interessam (PORTO, 2005). Este tipo de biorreator apresenta baixa eficiência e é comumente usado como base para as comparações de eficiências atingidas com relação a outros processos.
2. Processo em batelada alimentada: é caracterizado pela alimentação intermitente ou contínua sob condições controladas até atingir o volume do biorreator. Apesar de ser um processo antigo, este tipo de sistema é muito conveniente e satisfatório quanto à operação e eficiência de conversão de açúcares e etanol. O processo descontínuo alimentado é definido como uma técnica em processos microbianos, onde um ou mais nutrientes são adicionados ao fermentador durante o cultivo até o final da fermentação. A utilização de plantas de batelada alimentada ainda é amplamente utilizada

no Brasil. Acredita-se que cerca de 70% das destilarias ainda utilizam esse processo (ANDRIETTA; STECKELBERG; ANDRIETTA, 2006).

3. Processo contínuo: caracteriza-se por possuir uma alimentação contínua do meio de cultura a uma determinada vazão, sendo o volume da reação mantido constante pela retirada contínua do caldo de fermentação. O cultivo tem que ser mantido sob regime estacionário, ou seja, as propriedades do meio tem que permanecer constante com o tempo em cada ponto. O processo contínuo pode ser mais vantajoso que o batelada ou o batelada alimentada (FACCIOTTI, 2001). As principais vantagens do processo contínuo são: menores equipamentos, tempo de uso otimizado ou maior capacidade de fermentação com os mesmos equipamentos, rendimento médio alto, tempo de fermentação e paradas minimizadas, obtenção de caldo fermentado uniforme e menores gastos com mão de obra. Mas segundo FACCIOTTI, 2001, o processo contínuo também possui desvantagens, como: maior possibilidade de infecção, possibilidade de ocorrências de mutações genéticas espontâneas, possibilidade de seleção de mutantes menos produtivos e dificuldade de operação em estado estacionário. Mesmo com essas desvantagens, a utilização do processo contínuo esta apresentando uma crescente utilização. Não há estatísticas exatas, mas pesquisadores acreditam que o processo contínuo seja responsável pela produção de 25% a 30% do etanol produzido no Brasil. A escolha de qual processo utilizar (batelada ou contínuo) é um assunto que gera grandes discussões entre os especialistas e esta longe de ser esgotado, sendo exigido ainda muito estudo de Engenharia Cinética da Fermentação Alcoólica (PORTO, 2005). A Figura 1.3 apresenta uma foto com as dornas de fermentação contínua de uma usina de cana-de-açúcar.

Figura 1.3: Dornas de fermentação contínua. (Fonte: OSAKI, 2014)

Um dos fatores que vem limitando a instalação de processos contínuos para produção de etanol na indústria é a necessidade de sistemas de controle mais sofisticados. Além disso, a fermentação contínua é um processo que requer um maior conhecimento do microrganismo em relação ao meio que ele atua. Fatores como pH, temperatura, concentração de biomassa, entre outros, influenciam na produtividade do sistema, requerendo maior controle sobre o processo. Esses fatores dependem diretamente de alguns aspectos como o tamanho da superfície, das propriedades e dinâmica do fluido dentro do biorreator. A geometria do fermentador influencia diretamente esses fatores, quando o processo é contínuo. Sendo assim, é de suma importância a otimização da geometria do reator de forma a favorecer o processo fermentativo.

1.5 Microalgas e fotobiorreatores

A procura por fontes energéticas renováveis e viáveis economicamente cresce cada vez mais, e nesse contexto, as microalgas se destacam. As microalgas estão entre os organismos vivos que mais eficientemente utilizam energia luminosa para a produção de biomassa. A utilização de microalgas na produção de biocombustíveis e/ou compostos bioquímicos e o tratamento de efluentes em processos de grande escala se tornou um campo de estudo promissor e desafiador. O seu uso no tratamento da vinhaça, para a recuperação de nutrientes é de grande interesse para a sustentabilidade da indústria sucro-energética.

Os sistemas de produção de microalgas são geralmente classificados de acordo com as suas características hidráulicas e de engenharia como: sistemas abertos, que incluem lagoas, canais, unidades com circulação, etc.; e sistemas fechados, comumente chamados de

fotobiorreatores. Diferentes sistemas de fotobiorreatores têm sido testados e utilizados em laboratórios e em escala industrial (CARVALHO; MEIRELES; MALCATA, 2006; UGWU; AOYAGI; UCHIYAMA, 2008).

Os sistemas fechados ou fotobiorreatores possuem uma variada gama de configurações, sendo projetados para se obter uma maior produtividade de biomassa por unidade de volume. As configurações podem ser classificadas como: placas planas, alfa hélices, tubulares horizontais, concêntrico tipo *airlift*, *airlift* de placas planas, bissolares, semiesféricos e etc. Referências sobre a utilização desses tipos de fotobiorreatores podem ser encontrados em CORTÉS, 2009 e GRIMA et al., 1999. A Figura 1.4 ilustra a produção de microalgas em diferentes tipos de fotobiorreatores. Um reator tipo *airlift* de placas planas foi utilizado no presente estudo, o qual será descrito em detalhes na Seção 4.2.1.

Figura 1.4: Diferentes sistemas de fotobiorreatores: A) Sistema aberto *raceway* nos EUA; B) Sistema aberto lagoa circular Taiwan; C) Sistema fechado tipo placas planas; D) Sistema fechado tubular.(Fonte: CHISTI, 1989, [Wikipedia](#)).

Os parâmetros críticos dos fotobiorreatores incluem o regime de luz, material de construção, mecanismo de mistura, manutenção e estabilidade operacional em longo prazo com o objetivo neste caso de um rendimento máximo. Outro fator importante no uso de fotobiorreatores é a proteção contra contaminação de microrganismos estranhos. É necessária

a instalação de sistemas para controle de alguns parâmetros do processo, tais como: temperatura, vazão, pH, entre outros. Um exemplo de fotobiorreator que não apresentou resultados satisfatórios foi o construído em Cartagena na Espanha. Nesse caso, o acúmulo de oxigênio fotossintético e temperaturas excessivas foram as principais causas da falha (MIRÓN et al., 1999).

GRIMA et al., 1999 mencionam em seu trabalho que o projeto e metodologias de aumento de escala de fotobiorreatores são pouco desenvolvidos, o que ainda hoje pode ser constatado. Independentemente da configuração do fotobiorreator, varias questões de projeto devem ser abordadas para o seu pleno e ótimo funcionamento.

Segundo o projeto de pesquisa: Análise do acoplamento bio-foto-fluidodinâmico em fotobiorreatores industriais para o cultivo de microalgas a concentrações elevadas coordenado pelo Prof. Paulo Seleghim Jr., no qual este trabalho esta inserido, os fatores responsáveis pela degradação do desempenho de um fotobiorreator decorrem da interdependência entre três fenômenos. Esses fenômenos são: eficiência fotossintética e de crescimento celular, propagação, espalhamento e absorção luminosa e dinâmica do escoamento trifásico gás-liquido-sólido. Como estes efeitos são mutuamente dependentes, uma descrição através de modelos matemático-numérico acoplada será proposta, onde uma solução pode minimizar ou virtualmente eliminar todos os fatores responsáveis pela degradação do desempenho em escala industrial. Até o momento quase nada foi feito nessa área, exceto por alguns trabalhos que descrevem qualitativamente os chamados efeitos cruzados, como exemplo, a influência da fluidodinâmica sobre a mortandade celular (BITOG et al., 2011). Embora a relação entre a hidrodinâmica e transferência de massa tem sido amplamente investigada e correlacionada em biorreatores para culturas heterotróficas, poucos estudos sobre estes aspectos estão disponíveis em culturas fototróficas (UGWU; AOYAGI; UCHIYAMA, 2008).

1.6 Por que utilizar software livre?

A utilização de software livre ganha cada vez mais adeptos, uma vez que empresas e profissionais não necessitam despende elevadas somas de recursos na aquisição e manutenção de licenças, normalmente proporcionais ao número de processadores disponível ao usuário. Com a disseminação e a crescente utilização de clusters de computadores, onde centenas de processadores realizam milhares de operações de ponto flutuante em paralelo, fica ainda mais inviável a aquisição de software comercial. Para a aplicação eficaz desses

softwares, entretanto, é necessário o conhecimento de um conjunto de informações para o seu uso efetivo.

Os pacotes CFD são um exemplo dessa disputa entre pacotes comerciais e abertos. A importância dos pacotes CFD tem crescido muito nos últimos anos devido à confiabilidade e à rapidez dos resultados obtidos frente a testes experimentais. A escolha de um pacote CFD envolve várias considerações que devem ser avaliadas (SILVA, 2008). Alguns desses fatores são: a facilidade de uso, a capacidade para geração de geometrias e malhas, a eficiência e robustez das técnicas numéricas utilizadas pelo software e a amplitude dos problemas físicos que o pacote pode resolver. Atualmente os pacotes CFD que abrangem melhor essas características são os pacotes comerciais como o CFX e FLUENT (ANSYS) e PHOENICS (CHAM), principalmente devido aos grandes investimentos realizados por essas empresas e apelo comercial. Como consequência do apelo comercial, os preços desses pacotes são elevados, sendo nem sempre acessíveis ao meio acadêmico. Como exemplo, para um cluster com seis nós e 12 processadores cada, os custos com licenças de softwares alcançam os noventa mil reais.

Além do alto custo, os pacotes comerciais apresentam outras limitações, sendo que uma das maiores se refere à falta de detalhes na implementação de seus códigos. Essa falta de detalhes nos manuais é uma consequência do apelo comercial do software. Como pode ser constatado nestes manuais fornecidos pelos desenvolvedores do CFD comercial é feita uma breve descrição dos modelos e métodos numéricos utilizados. No entanto, essas descrições são insuficientes para explicar a implementação numérica em seu código, informações estas que poderiam fornecer ao usuário maior controle sobre as capacidades do pacote e evitar erros da sua utilização nas simulações.

A falta de acesso ao código fonte do pacote também pode ocasionar dificuldades para o usuário no ajuste ou adaptação de modelos do pacote CFD em seus problemas, além de impossibilitar que o pesquisador desenvolva novos modelos para uma aplicação específica. A possibilidade de verificar detalhadamente a modelagem, realizar modificações e poder implementar novos aplicativos fornece ao usuário possibilidades infinitas, sendo possível resolver problemas simples e mais complexos. Os softwares comerciais são denominados “caixas pretas”, onde o detalhamento interno do software é limitado a manuais.

Existem muitos códigos CFD gratuitos disponíveis, sendo a maioria desenvolvida no meio acadêmico e muitos destes possuem uma aplicação muito específica. Uma ferramenta CFD aberta deve fornecer ao usuário condições de analisar e implementar códigos e ao mesmo tempo ser geral o suficiente para ser aplicado em diferentes áreas. Com isso em

mente, nos últimos anos dois pacotes abertos vêm se destacando. Um deles é o MFI-X-Dem (Multi-phase Flow with Interface eXchanges) (Discrete Element Method) desenvolvido pelo National Energy Technology Laboratory (NETL) dos EUA. O outro pacote CFD que vem ganhando grande destaque é o OpenFOAM (Open Field Operation And Manipulation), o qual é utilizado neste trabalho.

A seguir são listados alguns fatores que levaram a escolha do OpenFOAM como ferramenta CFD para o desenvolvimento do presente trabalho:

- Código aberto e escrito em C++;
- Ferramenta de geração de malha e visualização de dados incorporados ao pacote;
- Possibilidade de interação direta com os desenvolvedores do software e com uma ampla comunidade de usuários;
- Ampla faixa de aplicações na engenharia;
- Facilidade de manipulação de dados e exportação de resultados;
- Possibilidade de desenvolvimento de códigos paralelos em GPU;

Por se tratar de um software aberto e sem uma interface gráfica que facilita a sua manipulação o OpenFOAM exige um tempo de aprendizagem bem maior se comparado aos pacotes comerciais. Para desenvolvimento de um código em OpenFOAM o usuário deve possuir conhecimento da linguagem de programação C++ e ler os manuais. Passado esse período de adaptação ao software e entendimento das diversas classes disponibilizadas, o desenvolvedor é capaz de simular escoamentos extremamente complexos. Esta facilidade de manipulação o torna uma ferramenta extremamente versátil para o pesquisador e de melhor proveito para o meio acadêmico.

Os pacotes CFD, pagos ou não, tipicamente requerem do usuário um grande esforço dedicado à compreensão e ao seu uso correto. Os códigos CFD irão fornecer uma resposta à maioria dos problemas quando estes forem apropriadamente colocados, porém é necessário paciência, prática e experiência para produzir resultados razoáveis.

1.7 Organização do texto

O presente trabalho está organizado da seguinte maneira:

- Capítulo 1: Introdução. Neste capítulo é feita uma abordagem do contexto geral na produção de bicomcombustíveis, mais especificamente de etanol, como as

suas vantagens e limitações de produção. Também é feita uma introdução a conceitos de biorreatores e fotobiorreatores e o porquê de se utilizar softwares de código aberto.

- Capítulo 2: Revisão da literatura. É apresentado um panorama geral dos trabalhos científicos onde trabalhos computacionais são aplicados na simulação de biorreatores para produção de bicom bustíveis, na indústria alimentícia e também na simulação de fotobiorreatores. Também é realizada uma revisão sobre trabalhos aplicando técnicas de otimização com CFD.
- Capítulo 3: Pacote CFD - OpenFOAM. É apresentado o pacote fluidodinâmico utilizado neste trabalho. Detalhes sobre a história e uso do software como ferramenta de fluidodinâmica são colocados. Uma breve introdução sobre a linguagem de programação C++ é colocada para que o leitor tenha uma base para acompanhar a programação interna do OpenFOAM e sua estrutura para manipulação e resolução de campos.
- Capítulo 4: Metodologia. Neste capítulo são abordados todos os detalhes da modelagem física e numérica do problema, inclusive a descrição do processo de otimização utilizado. É feita uma descrição da bancada experimental e são apresentadas as análises realizadas para caracterização da hidrodinâmica do reator *airlift*.
- Capítulo 5: Resultados e análises. Neste capítulo são apresentados os resultados e as referidas análises para ambos os reatores abordados.
- Capítulo 6: Conclusões. As conclusões são deduzidas à respeito das considerações finais e perspectivas futuras.

CAPÍTULO 2 CONCEITOS FUNDAMENTAIS E REVISÃO DA LITERATURA

Neste capítulo serão discutidos trabalhos sobre mecânica dos fluidos computacional aplicada à otimização de parâmetros geométricos e do escoamento em biorreatores e fotobiorreatores. Também serão discutidos trabalhos sobre a utilização de pacotes CFD com otimização multidisciplinar de projetos MDO juntamente com alguns métodos de otimização que serão foco de estudo deste trabalho.

2.1 Mecânica dos fluidos computacional

Dinâmica de fluidos computacional (do inglês *Computational Fluid Dynamics* – CFD) pode ser descrita como a área da computação científica que estuda métodos computacionais para a simulação de fenômenos que envolvem fluidos em movimentos com ou sem trocas de calor (FORTUNA, A. O., 2000).

O desenvolvimento do CFD moderno pode ser datado do início de 1950, com o advento dos computadores digitais. No entanto, o desenvolvimento dos métodos cruciais para o CFD, os métodos das diferenças finitas (do inglês *Finite difference method* - FDM) e métodos de elementos finitos (do inglês *Finite element method* - FEM) possuem origens diferentes. Em 1910, na *Royal Society of London*, Richardson (RICHARDSON, 1911) apresentou um trabalho onde resolvia numericamente a equação de Laplace utilizando FDM para a análise de tensões de uma barragem de alvenaria. Outro trabalho muito importante, considerado por muitos como o primeiro trabalho realizado utilizando-se FDM foi publicado em 1928 por Courant, Friedrichs e Lewy (COURANT; FRIEDRICHS; LEWY, 1928). Em contraste, o primeiro trabalho utilizando as técnicas do FEM foi publicado no *Aeronautical Science Journal* por Turner, Clough, Martin, e Topp em 1956 (TURNER et al., 1956) para a análise de tensões em aeronaves. Desde então, os dois métodos foram desenvolvidos e aplicados extensivamente em dinâmica de fluidos, transferência de calor e áreas afins.

Há uma crescente evidência dos benefícios provenientes do conhecimento combinado de ambos FDM e FEM. O método dos volumes finitos (do inglês Finite volume method - FVM) é um exemplo de método relacionado com o FDM e o FEM, devido a sua estrutura de dados simples, ele se torna cada vez mais popular. Outro exemplo é o método *flowfield-dependent variation* (FDV).

Historicamente, FDMs tem dominado a comunidade CFD. A simplicidade nas formulações e cálculos contribuiu para esta tendência. Por outro lado, FEMs são conhecidos por terem formulações mais complicadas e consumir mais tempo computacional. Entretanto, este não é mais o caso de desenvolvimentos recentes com aplicações utilizando FEM.

Nos últimos anos, como pode ser verificado em TANNEHILL; ANDERSON; PLETCHER, 1997, as técnicas em CFD passaram por um enorme progresso. A técnica se tornou muito poderosa e acompanhando o grande aumento do poder de processamento dos computadores o seu uso passou a abranger uma vasta gama de aplicações em diversas áreas. Alguns exemplos são:

- Aerodinâmica de aeronaves e veículos;
- Hidrodinâmica de navios;
- Usinas de energia: combustão em motores e turbinas à gás;
- Turbomáquinas: difusores e escoamento em passagens rotativas;
- Engenharia elétrica e eletrônica: resfriamento de equipamentos incluindo microcircuitos;
- Engenharia de processos químicos: mistura e separação;
- Ambiente interno e externo de edifícios: ventilação, aquecimento e carga de vento;
- Meteorologia;
- Engenharia biomédica: escoamento do sangue através das veias e artérias;

Cada vez mais, o uso de CFD está tornando-se um componente vital para o projeto de produtos e processos na indústria. A disponibilidade de hardware de computação de alto desempenho a preço acessível e a introdução de uma interface amigável ao usuário levaram a um aumento progressivo do interesse em softwares de CFD na comunidade industrial a partir da década 90.

O investimento necessário para o uso de CFD, incluindo licenças, treinamentos e hardwares não são baixos, mas o gasto total normalmente não é tão elevado quando comparado com a instalação de uma unidade experimental de alta qualidade. Algumas

vantagens do uso de CFD sobre abordagens experimentais para o projeto de sistemas de fluidos são (VERSTEEG; MALALASEKERA, 1995):

- Redução substancial de prazos e custos de novos projetos;
- Capacidade de estudar sistemas controlados onde é difícil ou impossível realizar experiências.
- Possibilidade de realizar estudos em condições perigosas além dos seus limites normais de desempenho;
- Nível praticamente ilimitado de detalhes dos resultados;

A variável de custo de um experimento, em termos de facilidade de construção e/ou custos hora-homem é proporcional ao número de pontos de dados e o número de configurações testadas experimentalmente. Em contraste códigos CFD podem produzir volumes extremamente grandes de resultados com praticamente nenhum custo adicional e é muito barato realizar estudos paramétricos, por exemplo, para otimizar o desempenho do equipamento.

Códigos CFD são estruturados ao redor de algoritmos numéricos que podem resolver os problemas de escoamento de fluidos. Os softwares CFD atuais podem ser divididos em três principais elementos: (i) pré-processamento, (ii) *solvers*⁴ e (iii) pós-processamento. A seguir é feita uma descrição cada item no contexto de um código CFD.

Pré-processamento

O pré-processamento consiste na definição do problema a ser resolvido no software CFD. Esta etapa pode ser dividida nos seguintes passos: definição da geometria da área de interesse (domínio computacional), subdivisão do domínio em um número de pequenos domínios (geração da malha ou *grid*), seleção das propriedades físicas e químicas do problema modelado, definição das propriedades do fluido e especificação de condições de contorno apropriadas.

A solução de um problema de mecânica dos fluidos (velocidade, pressão, temperatura, etc.) é definida nos nós dentro de cada célula. A precisão da solução do CFD é dominada pelo número de células na malha e pela precisão do método adotado. Em geral, uma grande quantidade de células melhora a precisão da solução. Precisão da solução, custo computacional e tempo de cálculo são ambos dependentes do refinamento da malha ou *grid*.

⁴ *Solvers* são arquivos executáveis construídos a partir do código fonte e de bibliotecas.

Solver

Os *solvers* são constituídos de métodos numéricos, os quais são responsáveis pela resolução numérica de um sistema de equações diferenciais que descrevem matematicamente o escoamento. Algumas dessas técnicas de solução numéricas são: diferenças finitas, elementos finitos, volumes finitos e métodos espectrais. A principal diferença entre esses métodos estão associados à maneira em que as variáveis do escoamento são aproximadas e como ocorre o processo de discretização.

Pós-processamento

Como no pré-processamento uma grande quantidade de trabalho está sendo gasto no desenvolvimento na área de pós-processamento. Devido ao aumento do poder de processamento alcançado nos últimos anos das unidades de processamento gráficos, muitos pacotes CFD possuem cada vez mais softwares equipados com ferramentas para visualização de dados. Essas ferramentas incluem: exibição do domínio e da malha, plotagem de vetores, superfícies 2D e 3D, rastreamento de partículas, manipulações de visualização (translação, rotação, escala, etc), criação de animações da dinâmica do escoamento, exportação dos dados e muitas outras opções.

Em todo processo de resolução numérica do escoamento de um fluido é necessário estar ciente de que os fenômenos físicos e químicos que ocorrem em um processo real são extremamente complexos. Deste modo os resultados gerados por um código CFD são aproximações que dependem de vários fatores para que tenham resultados plausíveis. Fatores estes como: identificação e modelagem do problema em termos do fenômeno físico e químico, dimensão necessária para o modelo 2D ou 3D, efeitos da temperatura ambiente ou variações de pressão, escolha do modelo de turbulência ou se estes efeitos são desprezíveis, entre outros. Portanto, são necessárias habilidades para se fazer a modelagem do problema de forma a reduzir a complexidade do mesmo a um nível admissível, preservando-se as suas principais características.

Algoritmos de solução numérica que possuem uma boa convergência, consistência e estabilidade também são cruciais para uma resolução adequada do CFD. O uso de métodos numéricos não implica que a mecânica dos fluidos experimental e as análises teóricas estejam sendo colocadas de lado. O ideal é que as três técnicas trabalhem juntas, se complementando durante um projeto que envolva escoamento de fluidos e no estudo de modelos teóricos para algum fenômeno particular, como turbulência.

Como citado em FORTUNA, 2000, um bom exemplo dessa combinação entre CFD e métodos experimentais (Figura 2.1) pode ser visto durante o projeto do aeroporto de Kansai, no Japão. Devido às grandes dimensões dos espaços internos da estrutura, o desenho do sistema de ar-condicionado foi realizado combinando simulações numéricas e testes experimentais (YAU; WHITTLE, 1991).

Figura 2.1: CFD complementa as análises teóricas e experimentais (fonte: FORTUNA, A. O., 2000).

2.2 Avanços do uso de CFD para caracterizar, projetar e otimizar sistemas de bioenergia

A energia pode ser classificada em renovável (proveniente de fontes que podem ser produzidas ou de meios infindáveis) e não renovável (proveniente de recursos limitados, como petróleo, por exemplo). Bioenergia é classificada como uma energia renovável que utiliza biotecnologia para converter biomassa em bicomcombustíveis e bioprodutos. As fontes de biomassa podem ser resíduos orgânicos da indústria e da agricultura ou culturas energéticas. A conversão da biomassa em bioenergia e bioprodutos pode ser obtida de uma ampla variedade de processos empregando uma série de tecnologias, como os biorreatores e fotobiorreatores.

Os avanços alcançados pelos softwares CFD juntamente com a evolução do poder de processamento dos computadores possibilitam hoje a realização de análises antes não possíveis. Pesquisadores, projetistas de equipamentos e engenheiros de processo estão utilizando cada vez mais CFD para o estudo de escoamentos e o desempenho de equipamentos, tais como reatores químicos, tanques agitados, leitos fluidizados, ciclones, sistemas de combustão, secadores de pulverização, matrizes de gasodutos, trocadores de calor, e outros equipamentos (BARRETT et al., 1994). O uso do CFD permite uma análise detalhada da mecânica dos fluidos e os efeitos locais destes tipos de equipamentos. Em muitos casos isso resulta em um melhor desempenho, maior confiabilidade, aumento de escalas (*scale-up*) mais confiantes, melhora da consistência do equipamento e aumento da produtividade da planta.

2.2.1 CFD e o estudo de biorreatores

A aplicação de CFD no estudo de biorreatores remonta do final dos anos 70, no entanto, foi a partir da década de 90 que vários trabalhos começaram a se difundir. Alguns exemplos são: BRUCATO et al., 1990; DELNOIJ et al., 1997; HARRIS; ROEKAERTS; F.J.J.ROSENDAL, 1996; KRESTA; WOOD, 1991; MARSCHALL; MLECZKO, 1999; RANADE; JOSHI, 1990.

Neste período, o trabalho de HARRIS; ROEKAERTS; F.J.J.ROSENDAL, 1996 merece destaque por realizar um trabalho abrangente em diferentes reatores utilizados em diversos setores da engenharia. Foram testados três tipos de escoamento em diferentes tipos de reatores. Diversos modelos foram implementados no software comercial Fluent e validados com dados de uma planta, permitindo que os cálculos do modelo CFD contribuíssem para o projeto de um reator melhorado. Seus resultados constituíram uma base para a identificação de tendências e potenciais obstáculos no projeto de reatores.

A última década assistiu a um rápido aumento no número de estudos sobre simulação numérica em biorreatores e fotobiorreatores. A evolução do CFD e da disponibilidade de computadores mais potentes abriu novas possibilidades para a modelagem e desenho deste tipo de equipamento. Como indicado na Tabela 2.1, as técnicas de CFD tem sido amplamente utilizadas na última década para simular os mais variados tipos de biorreatores e escoamentos.

O biorreator deve ser capaz de prover condições ótimas para o sucesso do processo biológico. A mistura não ideal em muitos processos fermentativos pode levar ao aumento da concentração de nutrientes, gases, pH, entre outros, afetando diretamente o processo. A frequência de exposição a esses fatores podem ser definidas com o cálculo da distribuição do tempo de circulação. DAVIDSON et al., 2003 efetuaram um estudo do escoamento em um tanque de mistura com um impelidor utilizando o software Ansys FLUENT. Os resultados qualitativos obtidos foram satisfatórios quando comparados a análises experimentais baseadas em processamento de sinais.

Buscando compreender melhor a dinâmica do fluido em tanques agitados por impelidor AUBIN; FLETCHER; XUEREB, 2004 utilizaram o software Ansys CFX. Neste trabalho foram investigados os efeitos turbulentos ocasionados pelas laminais do impelidor no fluido e a influência do modelo adotado para a caracterização do fluido, assim como o modelo turbulento e o esquema numérico. Os resultados foram validados utilizando-se dados experimentais obtidos através da técnica *laser doppler velocimetry* (LDV). Deste modo, verificou-se que a escolha do esquema numérico é extremamente importante, especialmente

para prever a energia cinética turbulenta. Os modelos turbulentos testados $k-\varepsilon$ e RNG tiveram pouca influência sobre a energia cinética turbulenta.

Tabela 2.1: Lista de trabalhos analisados que utilizaram CFD para investigar biorreatores.

Autores	Tipo de biorreator	Código CFD	Foco do estudo	Modelo de turbulência
DAVIDSON et al., 2003	Agitado por impelidor	Fluent	Estimação do tempo de circulação das partículas	$k-\varepsilon$
AUBIN; FLETCHER; XUEREB, 2004	Agitado por impelidor	CFX	Estudo da influência dos modelos turbulentos e esquemas numéricos sobre o escoamento	$k-\varepsilon$ / RNG
PATWARDHAN et al., 2005	Agitado por impelidor	CFX	Relação entre a geometria e as condições de funcionamento hidrodinâmica e desempenho do reator gás - líquido agitado	$k-\varepsilon$
XIA et al., 2008	Centrífugo com impelidor	Fluent	Estudo do perfil de velocidade, recirculação, tensão de cisalhamento e DTR	$k-\varepsilon$
UM; HANLEY, 2008	Agitado por impelidor	Fluent	Determinar a distribuição de viscosidade turbulenta durante o processo de fermentação de etanol	$k-\varepsilon$
SHAO et al., 2010	Agitado por impelidor	Fluent	Estudo numérico e experimental da influência da hélice do impelidor na produção de etanol	-
MOUSAVI et al., 2010	Horizontal tubular	Fluent	Otimização dos parâmetros que influenciam a produção de biomassa (bactérias metano-oxidantes)	$k-\varepsilon$
XUE; HEINDEL; FOX, 2011	Leito fluidizado	MFIX	Uso de CFD para estudo de pirólise rápida de celulose e bagaço	-
BANNARI et al., 2011	<i>Airlift</i>	OpenFOAM	Estudo da dispersão gás - líquido para análise da geometria do biorreator	$k-\varepsilon$
ZHANG; BAEYENS; TAN, 2012a ZHANG; BAEYENS; TAN, 2012b	Agitado por impelidor	Fluent	Estudar as características das bolhas de gás e o uso de impelidores no processo fermentativo para produção de etanol a partir de mandioca	-
SHAO et al., 2013	Agitado por impelidor	Fluent + EDEM	Estudar as interações de partículas sólidas com o líquido e o impulsor rotativo para verificar as influências das condições de funcionamento do reator e as propriedades das partículas	$k-\varepsilon$
ALOK; IMMANUEL, 2014	Agitado por impelidor	Fluent	Verificar a influência de diferentes tipos de impelidores na eficiência de um fermentador	$k-\varepsilon$ / $k-\omega$ / SST

Considerando que a estrutura do reator possui relações intrínsecas com as condições de operação e hidrodinâmica, PATWARDHAN et al., 2005 realizaram estudos a fim de otimizar a performance do reator a partir dos parâmetros estruturais. Foi realizado um estudo experimental e numérico do desempenho de um reator agitado por dois impelidores, com

escoamento gás-líquido. Os testes foram realizados em três geometrias de impelidores. A análise do perfil de velocidades e do tempo de residência do fluido foi analisada por terem relação direta com as reações químicas esperadas. Os resultados obtidos foram validados, fornecendo uma boa aproximação da distribuição do tempo de residência do reator em escala industrial.

XIA et al., 2008 observaram a necessidade de atentar-se para as variações que podem ocorrer em alguns parâmetros do escoamento para o bom desempenho do biorreator quando se aumenta a escala. Para o estudo, foi utilizado o software Fluent e parâmetros do escoamento como perfil de velocidade, recirculação, tensão de cisalhamento e distribuição do tempo de residência foram analisados em um biorreator com impelidor. Testes foram realizados com oito variações de impelidores em um reator tipo bancada apresentando resultados concordantes com os resultados experimentais.

Outro parâmetro importante para o projeto e desempenho de biorreatores é o entendimento da reologia de biomassa suspensa. No trabalho de UM; HANLEY, 2008, foi feito um trabalho experimental sobre o comportamento reológico de suspensões sólidas para ajustar um modelo apropriado aos dados experimentais e determinar a distribuição de viscosidade turbulenta durante o processo de fermentação do etanol. Estes dados experimentais foram utilizados para as simulações computacionais a fim de simular um aumento da concentração de biomassa. A análise mostrou um acentuado decréscimo da viscosidade em função do tempo no biorreator durante o processo de hidrólise enzimática e fermentação do etanol. Também se observou uma dependência direta entre a viscosidade aparente do caldo submetido ao processo de fermentação e a temperatura e concentração de biomassa.

SHAO et al., 2010 também realizaram um estudo sobre a influência de certos parâmetros no aumento de escala de um biorreator aplicado à produção de etanol celulósico. Dados como a densidade e a viscosidade do líquido, a densidade e as dimensões das partículas, fração volumétrica foram obtidos a partir de experimentos. A partir destes dados, o escoamento foi simulado computacionalmente testando várias velocidades de rotação da hélice.

Outro trabalho que buscou melhor compreender os parâmetros que influenciam a produção de biomassa em um biorreator é o de MOUSAVI et al., 2010. Foi utilizado CFD para simular e otimizar parâmetros como velocidade de entrada de ar, velocidade de entrada de líquido e viscosidade em um biorreator horizontal tubular. Para validar os resultados das simulações foram feitas comparações com dados experimentais adquiridos em um biorreator

piloto. As condições ótimas e os parâmetros mais significantes que afetam o comportamento hidrodinâmica foram determinados utilizando uma análise de modelo de variância. Os resultados mostraram que a velocidade de entrada do líquido exerceu a maior influência sobre a fração de volume do biorreator. No entanto, este tipo de aplicação não é compatível com a escala de produção requerida no processo industrial de produção de etanol em um fermentador.

Um trabalho interessante foi o realizado por XUE; HEINDEL; FOX, 2011. Este trabalho se destaca não somente pela sua aplicação, mas pelo software utilizado para o desenvolvimento do projeto, o MFiX. Um modelo CFD multifásico Euler-Euler é proposto para avaliar o processo de pirólise de biomassa em um reator tipo leito fluidizado. O modelo foi empregado para estudar a pirólise rápida de celulose e bagaço em um reator de escala de laboratório. A dinâmica e o aquecimento das partículas, reação da fase de biomassa, formação de carvão, elutriação e a distribuição do gás no interior do reator foram investigados.

Outro trabalho utilizando software CFD aberto encontrado na literatura seguindo as aplicações aqui mencionadas foi o realizado por BANNARI et al., 2011. Um modelo matemático utilizando técnicas CFD oferecidas pelo pacote OpenFOAM foi desenvolvido para estudar a dispersão gás-líquido em um reator tipo *airlift*. Esses tipos de reatores são interessantes para processos biológicos como a fermentação e produção de microalgas e cyanobacterias (LUO; AL-DAHMAN, 2004; POLLARD; A. P. ISON; SHAMLOU, 1998). O modelo desenvolvido foi utilizado para analisar uma geometria existente de um biorreator e validar a modificação do projeto inicial. O novo desenho do *airlift* obteve um ganho no desempenho, proporcionando um aumento da transferência global e local de massa e uma diminuição da tensão de cisalhamento.

Em ZHANG; BAEYENS; TAN, 2012a e ZHANG; BAEYENS; TAN, 2012b utilizou-se o CFD para o aperfeiçoamento de biorreatores para a produção de etanol a partir de mandioca. Nestes trabalhos foi aplicado o software Fluent na avaliação do uso de impelidores e misturador por bolhas induzidas na recirculação do fluido e na formação de bolhas de gás no processo fermentativo.

O trabalho de SHAO et al., 2013 destaca-se por utilizar a simulação CFD combinada com DEM (*Discret Element Method*) para simular a suspensão de partículas sólidas em um tanque agitado. Uma velocidade de rotação elevada causara uma tensão de cisalhamento extra sobre as partículas, de modo que a morfologia da particulada pode ser afetada, especialmente em processos relacionados a bio-/polímero- produtos. Assim, a abordagem acoplada CFD-

DEM mostrou-se eficaz para compreender a física do movimento das partículas sólidas em um tanque agitado.

Em ALOK; IMMANUEL, 2014 foi realizado um estudo do efeito de diferentes tipos de impelidores e defletores para determinar a correlação entre o tempo de mistura e o transporte de massa em um fermentador de tanque agitado. O trabalho envolveu a comparação de dados experimentais com simulações CFD onde a energia cinética turbulenta e a taxa de dissipação turbulenta foram calculadas para compreender a eficiência da mistura de cada impelidor.

2.2.2 CFD e o estudo de fotobiorreatores

Nas décadas de 70 e 80, a utilização de CFD para modelagem e projetos de fotobiorreatores (FBRs) foi praticamente abandonada devido ao fato de ser uma tarefa extremamente complexa (BITOG et al., 2011). A partir da década de 90, com os grandes avanços das técnicas computacionais e a disponibilidade de computadores mais potentes, tornou-se possível realizar uma análise mais detalhada do uso de fotobiorreatores para o crescimento de microalgas. A característica do escoamento no interior dos fotobiorreatores, os quais são muitos críticos, podem agora ser completamente analisados e visualizados.

No entanto, apesar destes avanços no desenvolvimento dos fotobiorreatores durante as duas últimas décadas, poucos projetos tiveram êxito na produção em larga escala (MIRÓN et al., 1999). Muitas limitações ainda dificultam a sua utilização, especialmente durante o aumento de escala (*scale-up*) (UGWU; AOYAGI; UCHIYAMA, 2008). O processo de aumento de escala é ainda pouco conhecido, devido sua complexidade e aos diferentes padrões de escoamento em distintas regiões, devido ao fato do projeto de um fotobiorreator envolver diferentes conjuntos de parâmetros tais como diâmetro, altura em relação ao diâmetro, tipo de gás injetado e as propriedades físicas do líquido (MOUZA; KAZAKIS; PARAS, 2004). O desenvolvimento de fotobiorreatores ainda é um dos principais passos que devem ser realizados para o cultivo de biomassa de algas em larga escala. Consequentemente, com os avanços das técnicas de CFD, os complexos padrões de escoamento, em relação aos modelos de fotobiorreator e sua operação podem agora ser investigados, obtendo-se assim soluções para a maioria dos problemas que foram encontrados.

Na literatura podem ser encontrados os mais variados tipos de FBR como, por exemplo: coluna de bolhas, *airlift*, tanque agitado, tubular, cônico, toro e etc. Os FBR são geralmente classificados de acordo com sua estrutura e pela produção de microalgas. O FBR

tubular é um dos tipos mais adequados para as culturas ao ar livre (BITOG et al., 2011). Em simulação computacional, os tipos mais comuns de FBRs que foram exaustivamente investigados estão o tipo coluna de bolhas e o *airlift* (BITOG et al., 2011).

Durante a última década, vários trabalhos de pesquisa utilizando CFD para o projeto de FBR foram publicados, alguns deles estão apresentados na Tabela 2.2.

Tabela 2.2: Lista de trabalhos analisados que utilizaram CFD para investigar FBRs

Autores	Tipo de FBR	Código CFD	Foco do estudo	Modelo de turbulência
KRISHNA; VAN BATEN; URSEANU, 2000	Coluna cilíndrica de bolhas	CFX	Simulação trifásica euleriana em regime agitado turbulento	Standart $k-\varepsilon$
MICHELE; HEMPEL, 2002	Coluna cilíndrica de bolhas	CFX	Influência da velocidade superficial do gás e a distribuição da velocidade do escoamento líquido	Standart $k-\varepsilon$
VAN BATEN; ELLENBERGER; KRISHNA, 2003	<i>Airlift</i>	CFX	Hidrodinâmica do escoamento	Standart $k-\varepsilon$
SZAFRAN; KMIIEC, 2004	<i>Jet-loop</i>	Fluent	Hidrodinâmica do escoamento em um escoamento trifásico (gás – líquido – sólido)	Standart $k-\varepsilon$
PRUVOST; POTTIER; LEGRAND, 2006	Toro	Fluent	Hidrodinâmica do escoamento, misturador com impelidor	$k-\omega$
DHOTRE; JOSHI, 2007	Coluna cilíndrica bolhas	Fluent	Projeto de um distribuidor de gás: simulação 3D de um sistema acoplado	$k-\varepsilon$
SIMONNET et al., 2008	Coluna retangular de bolhas	Fluent	Formulação da força de arrasto para descrever o regime de transições	$k-\varepsilon$
SU et al., 2010	Placa plana	Fluent	Estudo sobre misturadores desestabilizantes no FBR placa plana	$k-\varepsilon$
SEO et al., 2012	Coluna de bolhas	Fluent	Estudo de modelos multifásicos para interação bolha – escoamento	-
NAUHA; ALOPAEUS, 2013	Coluna cilíndrica de bolhas	Fluent	Desenvolvimento de um modelo abrangente para crescimento de algas em fotobiorreatores integrando a dinâmica dos fluidos	$k-\varepsilon$
WANG; TAO; MAO, 2014	Placa plana	CFX	Desenvolvimento de novo fotobiorreator de placa plana com defletores através de simulações CFD	$k-\varepsilon$
SOMAN; SHASTRI, 2015	<i>Airlift</i> placa plana	Fluent	Desenvolvimento de um fotobiorreator combinando características de um reator do tipo <i>airlift</i> e placa plana.	$k-\varepsilon$

KRISHNA; VAN BATEN; URSEANU, 2000 desenvolveram um modelo Euleriano-Euleriano para simulação de um escoamento gás-líquido-sólido de uma coluna de bolhas em regime turbulento se focando na investigação do processo de aumento de escala. Vale ressaltar que o trabalho realizado simulou computacionalmente um escoamento trifásico,

sendo KRISHNA; VAN BATEN; URSEANU, 2000 o primeiro a se atentar para isto no processo de simulação de FBRs (BITOG et al., 2011). Dois tipos de abordagens foram comparados, o primeiro foi um modelo assumindo simetria axial e o outro considerando um modelo tridimensional completo para as colunas cilíndricas. A comparação dos resultados experimentais com as simulações utilizando o modelo de Euler foi utilizada para concluir que o modelo bidimensional apresentou dados mais adequados para serem utilizados no processo de aumento de escala. Simulações para colunas com diâmetros variando de 1 a 6 metros foram realizadas para enfatizar a forte influência da escala hidrodinâmica.

A forte influência da escala hidrodinâmica no interior da coluna de bolhas é enfatizada em outros trabalhos subsequentes principalmente os de MICHELE; HEMPEL, 2002 e SZAFRAN; KMIEC, 2004. Os resultados das simulações CFD realizadas em MICHELE; HEMPEL, 2002 são muito significantes, uma vez que provaram a importância da correta modelagem das interações do gás disperso e fases sólidas em um escoamento trifásico.

A hidrodinâmica em reatores tipo *airlift* foi pesquisado por VAN BATEN; ELLENBERGER; KRISHNA, 2003, onde resultados experimentais e simulações CFD foram amplamente comparados. Os modelos bidimensionais axissimétricos se mostraram eficientes para este tipo de FBR. Assim esse modelo de CFD desenvolvido mostrou potencial de ser aplicado como uma ferramenta para o estudo de aumento de escala.

Uma investigação numérica da hidrodinâmica e as condições de mistura em um FBR, focando na influência da geometria do FBR sobre o crescimento de microalgas foi conduzido por PRUVOST; POTTIER; LEGRAND, 2006. Um FBR em formato de toro foi projetado de modo a permitir o controle de luz, enquanto a mistura é realizada através de um impelidor. Um alto grau de mistura foi observado, mostrando o potencial desta geometria inovadora para o cultivo de microrganismos fotossintéticos. No entanto, este processo foi validado em escala de laboratório sob condições bem controladas.

DHOTRE; JOSHI, 2007 utilizaram as simulações CFD em um FBR tipo coluna de bolhas. Um procedimento para ligar a câmara de gás no FBR foi proposto de forma a possibilitar a análise de diferentes padrões de injeção de gás. Foram feitas recomendações para o tamanho do bico de entrada e sua localização, área de abertura e o diâmetro do furo. A partir das informações quantitativas reportadas, pode ser possível selecionar os parâmetros dos projetos do distribuidor e a câmara de gás.

Outro trabalho que investiga a importância da formulação da força de arrasto para descrever o regime de transição em um FBR tipo coluna de bolhas foi feito por SIMONNET et al., 2008. O estudo mostra que a utilização da correlação arrasto permite reproduzir

algumas características típicas dos diferentes regimes como os perfis de velocidade, se tornando parabólicos no regime de transição, fenômenos transientes típicos, acumulação de bolhas grandes no centro da coluna na transição e regimes heterogêneos.

O trabalho de SU et al., 2010 utiliza uma estrutura desestabilizadora aplicada em um FBR tipo placa plana e simulou computacionalmente o campo de fluxo e a trajetória da partícula a fim de otimizar alguns parâmetros estruturais do FBR. Os resultados mostraram que o uso de desestabilizadores ajudou a atingir uma mistura mais homogênea necessária para o crescimento de microalgas.

Em SEO et al., 2012 técnicas de CFD foram utilizadas para realizar comparações quantitativas em projetos de FBR tipo coluna de bolhas, o qual é considerado pelos mesmos ser o melhor tipo de FBR para o cultivo de microalgas em larga escala. Quatro modelos multifásicos diferentes foram investigados para selecionar uma técnica de apropriada de modelagem e obter a forma de bolha e o escoamento do fluido ocasionado pela injeção da bolha. Os resultados foram validados com o auxílio de testes laboratoriais. Os resultados podem ser utilizados como base para o projeto de FBRs tipo coluna de bolhas para o cultivo de microalgas.

NAUHA; ALOPAEUS, 2013 utilizaram em seu trabalho um método abrangente para modelar o crescimento de algas em um fotobiorreator. Neste trabalho, o modelo integra a dinâmica do fluido, transferência de massa e a cinética de crescimento de algas. A disponibilidade de luz, nutrientes e carbono são consideradas. A abordagem de modelagem desenvolvida combinando CFD e o modelo comportamental demonstrou-se promissora podendo ser estendida para diferentes geometrias de fotobiorreatores. Um modelo abrangente, como este, é necessário para o projeto de fotobiorreatores e análise de aumento de escala, pois os fenômenos no reator são todos dependentes.

WANG; TAO; MAO, 2014 desenvolveram um novo fotobiorreator com defletores horizontais onde os efeitos da intensidade e localização da aeração foram investigados utilizando CFD. Os efeitos dos parâmetros estruturais dos defletores, incluindo as relações entre os defletores e as paredes do fotobiorreator e a distância entre os defletores foram avaliados. Em seguida uma análise experimental foi realizada no fotobiorreator desenvolvido comprovando o seu aumento de desempenho.

SOMAN; SHASTRI, 2015 também buscaram desenvolver um novo fotobiorreator combinando as vantagens dos reatores do tipo placa plana e *airlift*. O desenho utiliza o conceito da região de subida (*riser*) e região de descida (*downcomer*) de um reator *airlift*. No entanto, o *downcomer* tem uma superfície plana que se assemelha a um reator de placa plana.

O desempenho hidrodinâmico do fotobiorreator foi estudado utilizando técnicas de CFD e o desempenho foi comparado ao de um reator *airlift* convencional. As dimensões do novo projeto foram então otimizadas por meio de simulações CFD. O termo otimizado utilizado neste trabalho se refere à avaliação de uma quantidade fixa de cenários e, por conseguinte sua avaliação através da simulação CFD, não utilizando uma otimização multiobjetiva automatizada como este trabalho de doutorado propõe.

2.3 CFD com otimização multidisciplinar de projetos (MDO)

No projeto de muitos produtos e no desenvolvimento de aplicações, o CFD se tornou uma ferramenta de simulação analítica muito útil que está sendo aplicada nos mais diversos tipos de problemas. Para muitos projetos de produtos onde as simulações do escoamento dos fluidos são necessárias, as análises CFD provaram ser bastante úteis na previsão do padrão do escoamento para um dado conjunto de parâmetros do projeto. Considerando os esforços de modelagem e os recursos computacionais necessários, simulações CFD não são normalmente utilizados em sua plena capacidade de realmente otimizar o projeto (EL-SAYED; SUN; BERRY, 2005).

Tradicionalmente o processo de otimização de projetos com CFD é baseado em uma abordagem do tipo “tentativa e erro”. Partindo de uma geometria inicial, as mudanças no CAD são introduzidas manualmente com base em resultados de um número limitado de iterações de projetos e análises CFD. O processo é geralmente complexo, demorado e se baseia fortemente na experiência de engenharia, tornando o procedimento geral de concepção inconsistente, ou seja, diferentes soluções tidas como melhores são obtidas a partir de diferentes projetistas. Deste modo, não há garantia de que a solução final é, na verdade, a “melhor” solução. Esse tipo de metodologia foram aplicados nos processos de otimização utilizados nos trabalhos citados nas seções 2.3.1 e 2.3.2.

A utilização de análises CFD para otimização de projetos datam deste dos anos 70, em trabalhos focados em problemas aeroespaciais (HICKS; VANDERPLAATS, 1975, 1976, 1977). A partir disso a utilização de CFD com técnicas de otimização passaram a ser utilizadas principalmente nas áreas aeronáutica (MÄKINEN; PERIAUX; TOIVANEN, 1999; MELVIN et al., 1999), automobilística (CHOI; KIM; CHUNG, 1997; MUYL; DUMAS; HERBERT, 2004) e naval (CAMPANA et al., 2006; POLONI et al., 2000; THÉVENIN; JANIGA, 2008).

2.3.1 Definições básicas na otimização automática

Nos parágrafos seguintes são apresentadas as definições básicas e alguns conceitos sobre otimização automática, como apresentado na literatura (QUAGLIARELLA et al., 1998; RAO, 1996; THÉVENIN; JANIGA, 2008). O processo de otimização pode ser definido como sendo a busca do máximo absoluto (ou mínimo) de uma função, a qual depende de certas variáveis, respeitando certas equações de restrições. Métodos de otimização podem ser classificados de diversas maneiras, como mostrado em RAO, 1996. As classificações podem ser organizadas nas seguintes formas: na existência de restrições, na natureza das variáveis de projeto, na estrutura física do problema, na natureza das equações envolvidas, na natureza determinística das variáveis e no número de funções objetivos.

Algumas definições formais podem ser úteis para melhor entender o conceito de otimização automática. Primeiramente uma distinção entre otimização mono-objetiva e multi-objetiva deve ser feita. Na otimização mono-objetiva, a finalidade do problema pode ser matematicamente descrita como uma única função objetivo com as variáveis do problema. Em termos matemáticos o problema pode ser escrito como:

Encontre o conjunto de variáveis

$$X = \left\{ \begin{matrix} x_1 \\ x_1 \\ \dots \\ x_n \end{matrix} \right\}, \quad (2.1)$$

que determina o máximo (ou mínimo) valor da função objetivo $f(X)$ respeitando as condições:

$$\begin{aligned} g_j(X) &= 0, \quad j = 1, \dots, m \\ l_k(X) &< 0, \quad k = 1, \dots, p. \end{aligned} \quad (2.2)$$

Na otimização multiobjetivo, o problema é semelhante, mas em vez de uma única função objetivo, um dado número de funções tem de ser simultaneamente maximizada ou minimizada. Generalizando, o problema pode ser descrito matematicamente como no caso acima, de forma que encontre X , tal que maximize (ou minimize) $f_i(X)$, com $i = 1, \dots, q$, com ou sem equações condicionantes.

A pesquisa para o ótimo pode ser realizada utilizando uma variedade de algoritmos, onde a primeira distinção pode ser feita entre algoritmos baseados no gradiente e estocásticos.

Os algoritmos baseados no gradiente avaliam o gradiente da função objetivo e tentam achar o sentido crescente da função objetivo. O novo conjunto de variáveis é então selecionado de acordo com esta direção crescente dentro do domínio da função. Alternativamente, os algoritmos estocásticos pesquisam em todo o domínio da função através de múltiplas avaliações simultâneas no projeto e são especialmente adequados para problemas multiobjetivos. Um dos tipos mais utilizados de algoritmos estocásticos são os chamados algoritmos genéticos. Esta classe de algoritmos foi inicialmente proposta por John H. Holland no início da década de 70 (HOLLAND, 1975), baseada na evolução genética natural de um grupo de indivíduos.

Mais três conceitos adicionais devem ser considerados quando se comparam diferentes algoritmos, são eles: robustez, eficiência e precisão (NOCEDAL; WRIGHT, 1999). Robustez está relacionado à capacidade do algoritmo de encontrar o valor ótimo absoluto (máximo ou mínimo global) da função objetivo. Eficiência se refere à capacidade do algoritmo de não requerer muito esforço computacional ou armazenamento. Precisão, por outro lado, refere-se à capacidade de obtenção de uma solução o mais próximo possível do ótimo da função objetivo. Na Figura 2.2 é feita uma representação gráfica da robustez e precisão de um algoritmo de otimização.

Figura 2.2: Representação gráfica de robustez e precisão de um algoritmo de otimização (maximização).

Outra característica importante dos algoritmos de otimização é a velocidade de convergência, isto é, o número de iterações necessárias para alcançar a convergência (como por ex. encontrar o máximo da função objetivo). A seguir é apresentada a Tabela 2.3, na qual é feita uma comparação entre os algoritmos baseados no gradiente e os algoritmos estocásticos, com respeito às três características descritas acima.

Tabela 2.3: Comparação entre algoritmos baseados no gradiente e estocásticos.

	Algoritmos baseados no gradiente	Algoritmos estocásticos
Robustez	Baixa	Alta
Eficiência⁵	Alta	Baixa
Precisão	Alta	Baixa

Além dos algoritmos de otimização, outros conceitos da teoria da otimização são relevantes para o desenvolvimento de um projeto, são eles: Desenho experimental (em inglês: *Design of Experiments* – DOE ou *experimental design*), análise estatística e fronteira de Pareto.

DOE se refere ao processo de planejamento de um experimento, de modo que os dados adequados, quando analisados por métodos estatísticos resultem em conclusões válidas e objetivas. O DOE é utilizado no processo de otimização para selecionar as geometrias mais importantes a serem armazenadas em uma base de dados. A teoria do DOE é detalhada em vários livros como MONTGOMERY, 1997; THÉVENIN; JANIGA, 2008.

O último conceito importante na otimização multiobjetivo é a fronteira de Pareto. Em uma otimização mono-objetivo, a melhor solução é simplesmente o projeto que dá o máximo (ou mínimo) valor da função objetivo. Em uma otimização multiobjetivo, onde muitas funções objetivos estão envolvidas, a solução teórica seria o projeto que simultaneamente provém o valor máximo (ou mínimo) para todas as funções objetivos. Tal situação não acostuma acontecer, geralmente a melhor solução para uma função objetivo não corresponde ao melhor valor de outra, ocorrendo um conflito. Assim, o conceito de melhor projeto é substituído pelo conceito de projetos dominantes, que se concentram ao longo de uma linha geralmente conhecida como fronteira de Pareto. A Figura 2.3 mostra a representação da fronteira de Pareto para duas funções objetivos f_1 e f_2 considerando que elas têm que ser maximizadas. Os pontos no gráfico representam as possíveis escolhas, onde os valores maiores são os melhores. O ponto C não pertence à fronteira de Pareto, porque os seus valores $f_1(C)$ e $f_2(C)$ são menores. Em outras palavras os pontos A e B são dominantes sobre C porque os respectivos valores de ambas as funções objetivos são melhores. Por outro lado, comparando-se os pontos sobre a fronteira de Pareto, não há dominância estrita entre eles, pois não é possível melhorar a função objetivo sem que haja prejuízo para a outra. Considerando os pontos A e B, isto se traduz pela relação $f_1(A) < f_1(B) \Rightarrow f_2(A) > f_2(B)$. Em

⁵ Em referência a velocidade de convergência.

suma, a fronteira de Pareto pode ser definida como o lugar geométrico dos pontos não dominados, constituindo a solução ótima face a uma multiplicidade de objetivos. Um exemplo de otimização multiobjetivo com a determinação da fronteira de Pareto é apresentado no APÊNDICE A

Figura 2.3: Representação qualitativa da fronteira de Pareto

2.3.2 Algoritmos genéticos aplicados à otimização multiobjetivo

Uma das principais etapas de uma metodologia de otimização multiobjetivo é a identificação de um método adequado para a determinação das soluções não dominadas. Dentre os métodos de otimização multiobjetivo os algoritmos genéticos (AG) são os mais utilizados por terem a capacidade de produzir soluções de boa qualidade em problemas complexos e de grande porte. Desta forma, eles geralmente são utilizados para a resolução de problemas nos quais o espaço de busca é muito grande (COELLO; LAMONT; VELDHUIZEN, 2007).

A aplicação dos AGs para problemas multiobjetivo apresenta três grandes vantagens com relação a métodos clássicos de otimização:

- Não introduzem parâmetros adicionais no problema;
- Trabalham diretamente com várias funções utilizando o conceito de dominância de Pareto;
- Um conjunto diversificado de soluções pode ser encontrado apenas em uma execução do AG.

Os AGs multiobjetivo são comumente classificados de duas forma (DEB, 2001):

- Não elitistas: modelos que não utilizam alguma forma de elitismo nas suas iterações;
- Elitistas: modelos que empregam alguma forma de elitismo. O estudo realizado ZITZLER; DEB; THIELE, 2000 mostrou que o elitismo melhora as soluções encontradas por um AG multiobjetivo. A partir deste estudo, os novos trabalhos, de uma forma geral, passaram a incorporar alguma estratégia de elitismo.

Entre os inúmeros algoritmos genéticos multiobjetivos, destacam-se: VEGA, MOGA, NPGA, NSGA e SPEA. Uma breve descrição destes métodos será feito a seguir. Para mais detalhes aconselha-se COELLO; LAMONT; VELDHUIZEN, 2007; DEB, 2001.

O algoritmo VEGA (*Vector Evaluated Genetic Algorithm*) foi proposto por SCHAFFER, 1985. Foi um dos primeiros métodos de AG multiobjetivo implementado. Sua abordagem se difere de um algoritmo genético padrão na forma como a seleção é implementada. Esse operador foi modificado de forma que a cada geração um número de sub-populações é gerado aplicando-se uma seleção proporcional de acordo com cada função objetivo. As sub-populações podem ser cruzadas entre si e mutadas de forma usual. No entanto, segundo TICONA, 2003, o método proposto não obtém uma diversidade adequada de soluções ao longo da fronteira de Pareto. O método também não incorpora conceitos de otimalidade de Pareto (COELLO; LAMONT; VELDHUIZEN, 2007).

A partir do algoritmo VEGA um considerável número de AGs foram desenvolvidos basicamente com diferenças na forma de classificar as soluções baseadas no conceito de dominância e nos mecanismos de geração de novas populações para garantir convergência e diversidade no conjunto de soluções. A mais importante publicação relacionada a este período foi de GOLDBERG, 1989, o qual trazia o conceito de não dominância, uma ideia para a construção de uma eficiente técnica de otimização multiobjetivo que inspirou diversos trabalhos.

Desse grupo de algoritmos baseados na metodologia de não dominância pode-se citar, por exemplo, o MOGA. O MOGA (*Multiobjective Genetic Algorithm*), foi proposto por FONSECA; FLEMING, 1993. Esse algoritmo utiliza a noção do *ranking*, onde todos os indivíduos não dominados são colocados na primeira posição e os dominados são penalizados. Foi implementado também a ideia de nicho, para garantir a diversidade da população.

HORN; NAFLIOTIS; GOLDBERG, 1994 propuseram o algoritmo NPGA (*Niched Pareto Genetic Algorithm*). O NPGA utiliza um esquema de torneio de seleção baseado no

conceito de dominância de Pareto para seleção dos indivíduos. Dois indivíduos são selecionados e comparados com um subconjunto da população de solução, sendo selecionado para a próxima geração aquele que não for dominado. Quando os dois competidores são ao mesmo tempo dominados ou não dominados, o resultado do torneio é decidido através de comparação de *fitness*.

O algoritmo NSGA foi proposto por SRIVIVAS; DEB, 1995. Este algoritmo é baseado em várias camadas de classificação dos indivíduos. Antes da seleção toda a população é colocada em um ranking, onde todos os indivíduos não dominados são classificados em uma categoria e cada um recebe um valor de *fitness*. Esses indivíduos classificados são ignorados e o processo contínuo até toda a população ser classificada. Na seleção os indivíduos com maior valor de *fitness*, os primeiros a serem classificados, se reproduzirão mais que os outros.

No algoritmo SPEA (*Strength Pareto Evolutionary Algorithm*), proposto ZITZLER; THIELE, 1999 para avaliar a relação de dominância e classificar os indivíduos em níveis utiliza um conjunto adicional da população. Porém ao contrário dos algoritmos anteriores, os quais descartam os indivíduos não selecionados, ele emprega os indivíduos não dominados da população da geração anterior para determinar a aptidão dos indivíduos da população corrente.

Na Seção 2.3.2.1 é apresentado detalhadamente o algoritmo NSGA-II, um aperfeiçoamento do método NSGA (DEB et al., 2002), o qual foi utilizado neste trabalho. O NSGA-II é um dos principais modelos elitistas de otimização multiobjetivos encontrados na literatura (TICONA, 2003; ZITZLER; LAUMANN; THIELE, 2001).

2.3.2.1 Algoritmo NSGA-II

O algoritmo NSGA-II (*Non-dominated Sorting Genetic Algorithm II*) foi proposto por DEB et al., 2002 como uma evolução do algoritmo NSGA. O método NSGA-II é baseado em uma ordenação elitista por dominância (*Pareto ranking*). Esse procedimento consiste em classificar as soluções de um conjunto M em diversas fronteiras $F_1, F_2, F_3, \dots, F_k$ conforme o grau de dominância de tais soluções. Desta forma, a fronteira F_1 possui as soluções não dominadas de todo o conjunto M . A fronteira F_2 possui as soluções não dominadas de $M - F_1$, F_3 contém as soluções de $M - (F_1 + F_2)$ e assim sucessivamente.

Os principais mecanismos introduzidos no algoritmo NSGA-II são:

- Elitismo: garante a preservação de boas soluções no processo de busca;
- Procedimento *Fast Nondominated Sorting* (FNS): classifica a população em diferentes níveis segundo a dominância de Pareto;
- Procedimento *Crowding Distance Assignment* (CDA): busca garantir a diversidade da população.

O Algoritmo 1 exemplifica o procedimento de ordenação por dominância (*fast non dominated sorting*) proposto por DEB et al., 2002. A cada indivíduo (solução) i , do conjunto de soluções P , associam-se dois valores:

- nd_i : número de indivíduos que dominam o indivíduo i ;
- U_i : conjunto de indivíduos que são dominados pelo indivíduo i ;

Algoritmo 1: *Fast Non Dominated Sorting*

Entrada: M , conjunto de soluções

Saída: $F_1, F_2, \dots, F_k$, as fronteiras que classificam as soluções M

```

1  para cada indivíduo  $i \in M$  faça:
2       $nd_i = 0$ 
3       $U_i = \emptyset$ 
4      para cada indivíduo  $j \neq i$  e  $j \in M$  faça:
5          se  $i < j$  então  $U_p = U_p \cup \{j\}$ 
6          se  $j < i$  então  $nd_i = nd_i + 1$ 
5      fim
7      se  $nd_i = 0$  então  $F_1 = F_1 \cup \{i\}$ 
8  fim
9   $k = 1$ 
10 enquanto  $F_k \neq \emptyset$  faça:
11      $aux = \emptyset$ 
12     para cada  $i \in F_k$  faça:
13         para cada  $j \in U_i$  faça:
14              $n_j = n_j - 1$ 
15             se  $n_j = 0$  então  $aux = aux \cup \{j\}$ 
16         fim
17     fim
18      $k = k + 1$ 
19      $F_k = aux$ 
20 fim
```

As linhas 1-8 do Algoritmo 1 calculam os valores dos indivíduos e os conjuntos. Além disso, as soluções com $nd_i = 0$ (não dominados) estão contidas na fronteira F_1 . Em seguida, as linhas 9-20 percorrem o conjunto de indivíduos dominados U_i para cada indivíduo i de F_1 . O

contador nd_i de cada solução j em U_i é diminuído em 1. Se $nd_i = 0$, então a solução j pertence a seguinte fronteira (neste caso F_2). A iteração das linhas 9-20 é repetido até que todas as soluções estejam classificadas em uma fronteira.

O NSGA-II trabalha com duas populações, denotadas com P e Q de tamanho N_{ind} . Na primeira geração, os indivíduos iniciais da população P_1 geram as soluções em Q_1 através aplicação da seleção, recombinação e mutação. Essas populações P_1 e Q_1 são então combinadas, formando um conjunto R_1 com $2 \times N_{ind}$ indivíduos. A população expandida é classificada segundo a ordenação de dominância FNS e a próxima população P_2 já começa a ser composta pelos elementos de R_1 com menor índice de dominância a começar pelos indivíduos não dominados da fronteira 1 e assim por diante.

Para garantir a diversidade na fronteira calculada o NSGA-II utiliza uma estimativa da densidade das soluções que rodeiam cada indivíduo da população. Assim, calcula-se a média da distância das duas soluções adjacentes a cada indivíduo para todos os objetivos. Esse valor é denominado distância de multidão. O Algoritmo 2 apresenta os passos para calcular tal valor, onde $dist_S$ é a distância de multidão do indivíduo S do conjunto I , NS é o número de soluções contidas em I , e $f_{obj}(S)$ é o valor da obj -ésima função objetivo da solução S .

Algoritmo 2: *Crowding Distance*

```

1   $dist_S = \emptyset$ 
2  para cada  $obj$  faça:
3      classificar  $obj$  por  $f_{obj}$ 
4       $dist_0 = dist_{NS-1} = \infty$ 
5      para  $S=1$  até  $N-1$  faça:
6           $dist_S = dist_S + f_{obj}(S+1) - f_{obj}(S-1)$ 
7      fim para
8  fim para
9: Retorne  $dist_S$ 

```

O NSGA-II emprega o processo de seleção por torneio, considerando que a aptidão (*fitness*), $rank_S$, de cada indivíduo S , depende da fronteira Ni_d à qual pertence e da distância de multidão $dist_S$. Neste caso, um indivíduo S é comparado a um indivíduo p para escolher qual deles deve ser utilizado para gerar descendentes na nova população.

O indivíduo selecionado será aquele que possuir um menor valor de *rank*. Ou seja, o indivíduo S será escolhido se S possui um *ranking* menor que p ($rank_S < rank_p$). Se ambos os indivíduos possuem o mesmo *ranking*, será escolhido aquele que possuir um maior valor de distância de multidão ($rank_S = rank_p$ e $dist_S = dist_p$). O cálculo da distância de multidão permite que as soluções melhor espalhadas passem a ocupar as últimas vagas disponíveis de P_{i+1} garantindo a diversidade das soluções.

Se a última fronteira a ter seus indivíduos acrescentados na população P_{t+1} tiver uma quantidade de soluções que, se somada às aquelas já acrescentadas, exceder o tamanho de N de P_{t+1} , os indivíduos excedentes são desprezados segundo o critério de diversidade, ou seja, aqueles com maior valor do índice de distância são os escolhidos para compor P_{t+1} . Tal procedimento é ilustrado na Figura 2.4.

A sequência de passos seguido pelo NSGA-II é descrita no Algoritmo 3.

Algoritmo 3: NSGA-II

Entrada: Conjunto de parâmetros relevantes ao NSGA-II

Saída: Soluções na populações P_{final} e Q_{final} .

Inicialização

```

1   Criar uma população de soluções aleatórias  $P_1$  de  $N$  indivíduos
2   Ordenar  $P_1$  por dominância
3   Aplicar operadores genéticos em  $P_1$  para gerar uma nova população,  $Q_1$  de
   tamanho  $N$ 
4   para cada geração  $t = 2, \dots, N$  faça:
5       Aplicar o Algoritmo 1 em  $R_t = P_t \cup Q_t$ 
6        $k = 1$ 
7       enquanto  $|P_{t+1} + F_k| \leq N$  faça:
8           Aplicar o Algoritmo 2 em  $F_k$ 
9            $P_{t+1} = P_{t+1} \cup F_k$ 
10           $k = k + 1$ 
11      fim
12      Aplicar o Algoritmo 2 em  $F_k$ 
13      Classificar a  $F_k$  pelo ranking e a distância de multidão
14      Copiar as primeiras  $N - |P_{t+1}|$  soluções de  $F_k$  para  $P_{t+1}$ 
15      Gerar a nova população  $Q_{t+1}$  aplicando os operadores genéticos em  $P_{t+1}$ 
16  fim
17   $P_{final} = P_t$ 
18   $Q_{final} = Q_t$ 

```

Figura 2.4: Esquema do modelo NSGA-II (Adaptado de DEB et al., 2002).

2.4 CFD no processo de otimização de projetos

A introdução de uma estrutura matemática para encontrar uma concepção otimizada através da utilização das mais recentes técnicas de otimização, padroniza até certo ponto o processo de projeto e elimina a maioria dos inconvenientes encontrados na abordagem tradicional, como já descrito em seções anteriores. Os projetistas estão confiando cada vez mais neste tipo de processo automatizado, a fim de reduzir o tempo de desenvolvimento de produtos, satisfazer as exigências de projetos complexos e se manter em um mercado em pleno crescimento.

A fim de introduzir o processo de CFD para o ciclo de otimização, a metodologia tradicional manual para a definição de um caso CFD (ou seja, preparação CAD, geração da malha, configuração do *solver*, cálculo e pós-processamento) precisa ser totalmente automatizado por meio de *scripts*⁶ ou macros. Quando esses *scripts* são lidos e executados em sequencia usando um software de otimização avançado, o processo manual torna-se uma ferramenta de CFD automatizada. Algoritmos baseados no gradiente conjugado ou algoritmos estocásticos são então empregados para definir e realizar as variações em vários projetos a fim de encontrar uma solução ótima. O processo geral é representado na Figura 2.5.

⁶ *Scripts* podem ser definidos de maneira direta como “roteiros” seguidos por sistemas computacionais e trazem informações que são processadas e transformadas em ações efetuadas por um programa principal.

Figura 2.5: Metodologia de otimização automática com CFD.

Um exemplo de um processo de otimização automática com CFD amplamente difundido é o oferecido pela ANSYS através do acoplamento dos seus softwares CFX e Fluent com a empresa ESTECO (ESTECO, 2012). A empresa ESTECO (ESTECO, 2012) oferece uma ferramenta comercial para otimização multiobjetivo e um ambiente de projeto, acoplamento CAD/CAE, análise estrutural e CFD chamado ModeFRONTIER. O ModeFRONTIER é um software com interface gráfica baseado em Java que envolve uma ferramenta CAE, realizando a otimização modificando os valores atribuídos às variáveis de entrada e analisando os resultados nas funções objetivos. O software de otimização ModeFRONTIER é composto de modelos que vão desde métodos baseados no gradiente conjugado e algoritmos genéticos. A ferramenta é amplamente utilizada em projetos na área automobilística, a qual foi a motivação da sua criação. No entanto, é possível encontrar sua aplicação nas mais diversas áreas (PAPANIKOLAOU, 2010; SHI, 2011; SPOGIS; NUNHEZ, 2009).

Como todo software comercial a ferramenta apresenta algumas desvantagens, principalmente pelo fato de seus métodos de otimização ficarem em uma caixa preta, onde o usuário não tem acesso e por não permitir a alteração ou desenvolvimento de novos códigos.

2.5 Conclusão

O projeto de equipamentos na indústria sucroalcooleira segue uma abordagem extremamente experimentalista. Essa abordagem seguia uma linha de modelagem de equipamentos com base em aspectos empíricos ou semi-empíricos devido a uma longa história de análise dimensional na indústria de equipamentos. Com as recentes pesquisas e os esforços para o desenvolvimento de seus códigos na área da dinâmica dos fluidos computacional, vários pacotes CFD provaram ser úteis no desenvolvimento de muitos projetos. Neste capítulo foi feita uma revisão de vários trabalhos que utilizam técnicas CFD

para o melhoramento estrutural de vários tipos de biorreatores e fotobiorreatores. Mas como observado é raro encontrar trabalhos nesta área de aplicação que utilizam técnicas de otimização automáticas, como os propostos por SPOGIS; NUNHEZ, 2009 e GÓIS, 2012 que utilizam o ModeFRONTIER. Assim esta área de aplicação carece de trabalhos que explorem o desenvolvimento de metodologias especializadas de otimização multiobjetivos automatizadas e no mapeamento das patologias características dos problemas de otimização associados ao projeto de biorreatores e fotobiorreatores de alto desempenho.

CAPÍTULO 3 PACOTE CFD - OPENFOAM

Neste capítulo é feita uma breve descrição do pacote CFD OpenFOAM e são discutidos alguns aspectos relativos ao desenvolvimento e adaptação de novos *solvers*. Uma descrição mais detalhada da metodologia empregada no OpenFOAM pode ser encontrada em consultas na documentação oficial disponibilizada no site⁷, fóruns de discussões⁸ e em JASAK, 1996.

3.1 O surgimento do OpenFOAM

Em 1993 dois doutorandos do *Imperial College*, Henry Weller e Hrvoje Jasak, uniram forças para desenvolver um código para operar e manipular campos tensoriais visando sua aplicação à fluidodinâmica computacional. Surgiu assim o termo FOAM (*Field Operation And Manipulation*). A tecnologia FOAM é baseada em um conjunto eficiente e flexível de módulos em C++, de forma a facilitar o seu uso para desenvolvimento de códigos que possam resolver problemas específicos da engenharia que envolvam campos vetoriais. A ferramenta buscava também unir utilitários para realizar tarefas de pré e pós-processamento, como manipulação de dados e construção e processamento de malhas, além de permitir bibliotecas de expansão usadas pelos *solvers*, como uma biblioteca de modelos físicos e utilitários.

Durante alguns anos, o FOAM passou a ser desenvolvido por uma empresa inglesa chamada Nabla seguindo uma linha comercial. O intuito da parceria com esta empresa era de desenvolver uma ferramenta para simulação CFD de processos fisicamente complexos e oferecendo serviços de consultoria sobre o uso do software e implementação de novos *solvers* com modelagem complexa. Essa parceria durou até 2004, quando os desenvolvedores concluíram que a abordagem comercial não era a ideal. Desta forma, o FOAM passou a ter domínio público sobre a GPL (*Gnu Public License*) e o programa passou a se chamar OpenFOAM, com referência ao código aberto.

⁷ <http://www.openfoam.org/docs/>

⁸ <http://www.cfd-online.com/>

A ideia de tornar o código aberto do OpenFOAM é devido ao fato de que ele se torne uma ferramenta de pesquisa e de que o futuro do código possa ser garantido através de colaborações e interações com o meio acadêmico. Deste então, a comunidade científica têm tratado o OpenFOAM com muito interesse, testando-o para casos de *benchmark*⁹, utilizando em projetos de pesquisa e propondo inúmeras colaborações para o desenvolvimento do código (RUSCHE, HENRIK, 2002; SILVA, 2008).

Hoje em dia é possível encontrar uma vasta comunidade na internet de usuários e desenvolvedores, onde são discutidos aspectos de uso, implementação de código e análise numérica do software. Como citado em SILVA, 2008 um comentário feito por Jasak reflete a situação atual do OpenFOAM: “É incrível a popularidade que o OpenFOAM está recebendo da comunidade científica. O código está se espalhando como um poderoso vírus na internet”. O objeto de se usar o código em um ambiente mundial de colaboração está sendo alcançado, fazendo com que o software seja considerado uma das maiores plataformas de CFD em código aberto. Grandes empresas multinacionais já visualizaram seu potencial e utilizam o OpenFOAM no desenvolvimento de seus projetos de alta tecnologia. Alguns exemplos dessas empresas são: AUDI, Bayer, Shell, Honda, Airbus.

Atualmente o OpenFOAM é mantido pela ESL-OpenCFD e distribuído por meio da OpenFOAM Foundation através do site <http://www.openfoam.com/>. A versão atual do OpenFOAM é a 2.3.1 (Data do *release*: 10 dezembro de 2014).

3.2 O OpenFOAM a nível de usuário

Em nível de usuário comum, ou seja, aqueles que usam os solvers que já estão implementados no software a sua utilização é bem simples e pode ser dividida nas seguintes etapas:

1. Gerar a estrutura necessária para efetuar a simulação;
2. Pré-processamento: geração da geometria e da malha e definições de parâmetros da simulação;
3. Solução numérica do problema: nesta etapa o modelo utilizado é resolvido de acordo com as condições impostas;
4. Pós-processamento: visualização dos resultados.

⁹ *Benchmark* é o ato de executar um programa de computador, ou um conjunto deles, a fim de avaliar a performance relativa a um objeto, normalmente executando uma série de testes padrões.

O OpenFOAM é fornecido com ambientes de pré e pós-processamento, onde as interfaces são próprias do software assegurando assim uma manipulação consistente entre todos os ambientes. A estrutura geral do OpenFOAM é mostrada na Figura 3.1.

Figura 3.1: Visão geral da estrutura do OpenFOAM (fonte: OpenFOAM User's Guide)

3.2.1 Estrutura básica para efetuar uma simulação no OpenFOAM

O OpenFOAM utiliza um conjunto de arquivos para o armazenamento das informações necessárias para se resolver um caso. Assim, cada caso segue uma estrutura de diretórios que contém os arquivos que armazenam as informações necessárias para a execução do *solver*. Estes arquivos possuem as informações como a descrição da geometria, detalhes da malha, condições de contorno, parâmetros para os métodos numéricos e as propriedades físicas do problema. Na Figura 3.2 pode ser observada uma representação de um caso genérico da estrutura de diretórios.

O diretório principal <Nome do caso> é também denominado de diretório raiz. Dentro deste diretório estão todos os outros subdiretórios e arquivos de configuração. A seguir é feita uma breve descrição sobre o conteúdo destes diretórios.

- <Dir. do tempo>: neste diretório estão contidos os arquivos individuais de dados para os campos das variáveis tratadas no caso, como o campo de velocidades, pressão, tensão, etc. Cada instante simulado é criado um diretório o qual se refere ao tempo simulado e dentro dele ficam armazenados estes arquivos. É neste diretório que são definidas as condições iniciais do problema.
- <system>: dentro deste diretório se encontram os arquivos associados aos procedimentos de solução do caso. Neste diretório é obrigatório existir pelo menos 3 arquivos: controlDict, fvSolution e o fvSchemes, que serão analisados a seguir.

- <constant>: neste diretório estão contidos todos os arquivos de propriedades físicas pertinentes ao caso, por exemplo, transportProperties. No subdiretório polyMesh ficam localizados os arquivos referentes a descrição completa da geometria e da malha, arquivos estes denominados blockMeshDict e boundary.

Figura 3.2: Estrutura de diretórios e arquivos necessários para uma simulação com o OpenFOAM

Esses arquivos podem ser alterados diretamente pelo usuário usando um editor de texto. É proposta de trabalho futuro do presente projeto o desenvolvimento de uma interface gráfica para o software PyCFD-O, de modo a permitir que o usuário possa alterar os dados pertinentes ao caso.

3.2.2 Pré-processamento

A criação da geometria e principalmente a geração da malha é uma etapa muito importante para simulações em CFD. Uma malha computacional é constituída por linhas e pontos, os pontos ou nós são considerados onde essas linhas se interceptam e servem de orientação para o cálculo de propriedades físicas com base num modelo matemático. Uma malha computacional nada mais é que uma representação ou a “discretização” do plano físico utilizado na simulação numérica. A solução de um sistema de equações diferenciais (modelo matemático) pode ser geralmente simples quando empregado uma malha bem construída.

Uma malha mais fina, ou seja, com uma maior quantidade de pontos do domínio, produz resultados com maior precisão. Contudo, quanto maior a quantidade de nós maior o número de operações a ser resolvido e consequentemente maior o custo computacional. Desta forma, deve-se optar por malhas que forneçam resultados com precisão adequada e que exijam um tempo de simulação aceitável.

Diferentemente dos pacotes comerciais o OpenFOAM não possui um editor CAD¹⁰ para construção da geometria do problema. Para malhas estruturadas, em alguns casos é possível editar o arquivo `blockMeshDict` diretamente e criar a geometria. Após a edição é utilizado o comando `blockMesh` para gerar a malha a partir do arquivo `blockMeshDict`. O OpenFOAM foi desenvolvido para trabalhar com malhas tridimensionais, assim sendo necessário prestar atenção quando for trabalhar com malhas de diferentes dimensões.

No entanto, na grande maioria dos casos não é possível a criação da geometria sem o auxílio de softwares. Assim, pode-se optar por softwares comerciais como o ICEM e GAMBIT ou livres como o NETEGEN, TETGEN, GMSH e SALOME para a criação da geometria e geração de malha. O OpenFOAM aceita diferentes tipos de geometrias para as células, como hexaédricas, tetraédricas, prismática, piramidal, etc. O OpenFOAM também conta com o auxílio de várias funções para a importação das malhas como pode ser verificado nos seus manuais (*User's Guide* e *Programmer's Guide*).

Após a criação da geometria e a geração da malha é necessário definir as condições de controle da simulação nos arquivos localizados no diretório `<system>` e as propriedades físicas e modelos adicionais do problema no diretório `<constant>`. Uma descrição dos arquivos de controle é feita a seguir:

- `controlDict`: controla o tempo de simulação, passo de tempo, intervalo de escrita de dados, forma de compressão de dados, etc.
- `fvSchemes`: especifica os métodos para discretização dos termos derivativos das equações. O método de discretização padrão adotado pelo OpenFOAM é a integração de Gauss para volumes finitos.
- `fvSolution`: neste arquivo são definidos os métodos de solução do sistema de equações lineares e suas respectivas tolerâncias, assim como alguns parâmetros para o algoritmo de solução do campo de escoamento (correção pressão-velocidade e ortogonalidade da malha).

¹⁰ Computer-Aided Design ou em português Desenho auxiliado por computador (DAC) é o nome genérico de sistemas computacionais utilizados para facilitar o projeto e o desenho técnico.

Arquivos adicionais para controle de simulação podem ser adicionados no diretório <system>, como por exemplo um arquivo para controle das simulações em paralelo (decomposeParDict). O OpenFOAM utiliza a biblioteca de domínio público MPI (*Message Passage Interface*) para a comunicação entre os computadores e a decomposição do domínio pode ser feita por quatro métodos: simple, hierarchical, scotch e o manual. Mais detalhes sobre esses métodos podem ser encontrados no *User Guide*.

Os métodos numéricos para a discretização dos termos derivativos das equações são definidos no arquivo de configuração fvSchemes como mencionado acima. É imperativo que cada termo da equação esteja vinculado a uma aproximação numérica. O OpenFOAM tem como padrão a integração de Gauss para volumes finitos para o método de discretização. A integração de Gauss é baseada na soma dos fluxos das variáveis nas faces dos volumes, que são interpolados a partir do centro dos volumes. O usuário tem liberdade de escolher o método de interpolação a ser usado. Alguns dos métodos de interpolação disponíveis são: interpolação linear (diferenças centrais), upwind, QUICK, SFCD, esquemas TVD (Total Variation Diminishing) e esquemas NVD. Os métodos disponíveis para a integração no tempo são: Euler (primeira ordem, restrito, implícito), Crank-Nicholson (Segunda ordem, restrito, implícito), backward (segunda ordem, implícito) e steadyState (não resolve a derivada temporal).

Os métodos de solução de equações lineares e suas respectivas tolerâncias assim como alguns parâmetros para o algoritmo de solução do campo de velocidades do escoamento, como correção pressão-velocidade e ortogonalidade da malha são definidos no fvSolution. Alguns dos métodos iterativos utilizados no OpenFOAM são: método de Gauss-Seidel, multigrid algébrico e variantes do gradiente conjugado (PCG e PBiCG). Como os métodos são iterativos é preciso definir um parâmetro de parada para o método, um deles é quando o resíduo se torna menor que a tolerância especificada (`tol`) e a razão entre os resíduos da iteração atual e inicial for menor que a tolerância relativa (`relTol`). Mais detalhes sobre os métodos numéricos podem ser encontrados em GOLUB; VAN LOAN, 1996; PRESS et al., 1992.

Os métodos para o acoplamento pressão velocidade utilizados no OpenFOAM são: PISO (*Pressure-implicit split-operator*) e SIMPLE (*Semi-implicit method for pressure-linked equations*). Os dois algoritmos são baseados em procedimentos iterativos, avaliando a solução em um dado instante de tempo e então a corrigindo. O OpenFOAM utiliza o SIMPLE para simular casos estacionários e o PISO para casos transientes. Detalhes sobre os métodos

podem ser encontrados em livros de mecânica dos fluidos computacional como FERZIGER; PERIC, 2001; MALISKA, 1995 ou na tese do Prof. Jasak (JASAK, 1996).

O OpenFOAM apresenta implementadas as principais condições de contorno, como entrada de massa (inlet), saída de massa (outlet), parede fixa ou móvel (wall), condições atmosféricas, simetria, entre outras. O usuário tem a liberdade de desenvolver a sua própria condição de contorno, caso seja necessário. Para completar os principais parâmetros a serem definidos no OpenFOAM temos o arquivo que possui as propriedades físicas do fluido.

3.2.3 Resolução Numérica – Solver

Voltando a Figura 3.1, temos a próxima etapa correspondente à resolução numérica das equações do modelo. O OpenFOAM utiliza os arquivos executáveis denominados de *solvers* para realizar tal propósito. Esses arquivos possuem os modelos para resolver o caso pretendido e as informações de todas as rotinas de cálculo a serem executados. Deste modo, quando os *solvers* são invocados é feita a leitura de todos os parâmetros da simulação definidos nos arquivos do caso especificados na etapa de pré-processamento. Com estas informações o *solver* pode partir para resolução numérica do problema.

O OpenFOAM apresenta uma grande diversidade de *solvers* disponível no pacote original. No entanto, se houver a necessidade do usuário usar um modelo que não se encontra na versão original é permitido criar um novo solver para o caso específico. Essa é uma das grandes vantagens em se trabalhar com softwares *open source*.

3.2.4 Pós-processamento

O OpenFOAM possui uma versão do software Paraview (PARAVIEW, 2012) para o pós-processamento dos resultados denominada paraFoam. O Paraview é um software de código aberto, multi-plataforma e projetado para a visualização de dados de tamanho variado, desde arquivos pequenos até arquivos muito grandes.

O paraFoam é um acoplamento feito entre o OpenFOAM e o Paraview, onde as ferramentas básicas para visualização de resultados CFD estão incluídas. Algumas dessas ferramentas são: criação de gráficos de contorno, vetores e linhas de fluxo, linhas de corrente, etc. O OpenFOAM também permite a exportação de seus dados para que possam ser utilizados outros softwares de visualização com recursos mais avançados. É possível converter os resultados fornecidos pelo OpenFOAM para formatos lidos por softwares como

Ansys, Feildview, Ensignt e Tecplot. Também existe uma ferramenta no OpenFOAM para a conversão dos resultados para o formato VTK possibilitando a leitura dos dados em qualquer visualizador com este padrão.

3.3 OpenFOAM a nível de usuário desenvolvedor

Como já mencionado, uma das grandes vantagens em usar softwares com códigos abertos é a possibilidade do usuário alterar e criar novas rotinas. No entanto, o usuário desenvolvedor deve possuir conhecimentos prévios para conseguir realizar esta tarefa. No caso do OpenFOAM, os desenvolvedores necessitam ter um conhecimento em linguagem C++ e um conhecimento dos operadores, classes e funções já existentes no software. Além desses conhecimentos é indispensável um conhecimento em CFD.

As principais fontes sobre programação no OpenFOAM são seus manuais (*User's Guide* e *Programmer's Guide*) assim como a documentação disponível no site <http://www.openfoam.org/docs/cpp/> (*OpenFOAM C++ Documentation*). Também existem vários fóruns e comunidades que podem auxiliar muito na aprendizagem, além de cursos oferecidos pela OpenCFD e algumas instituições.

É altamente recomendado para o usuário desenvolvedor ler a tese de Jasak (JASAK, 1996) que apresenta detalhadamente vários aspectos sobre a formulação numérica, incluindo a metodologia de discretização, condições de contorno, etc., e a teoria de algoritmos implementados no OpenFOAM.

3.3.1 Orientação a objetos e C++

Uma das principais funções da Programação Orientada a Objeto (POO) é a possibilidade de desenvolvimento de códigos que podem ser reaproveitados. Na abordagem orientada a objetos, os dados são tratados como elementos críticos do programa e não é permitido alterá-los livremente. Os dados são associados a funções que acessam e operam, protegendo-os de modificações ocasionadas por funções externas. A orientação a objetos permite a decomposição do problema em entidades chamados objetos, a qual somente funções previamente criadas podem operar sobre esses objetos (YANG, 2001).

Para o entendimento da linguagem orientada a objetos, o desenvolvedor deve ter em mente os conceitos de objetos, classes, abstração de dados, encapsulamento, herança, e polimorfismo. Estes conceitos são considerados entidades básicas de um sistema orientado a objeto. Detalhes sobre esses conceitos podem ser encontrados em qualquer livro de

programação orientada a objetos. Uma boa referência para C++ com POO é o trabalho de YANG, 2001. A linguagem C++ foi desenvolvida por Bjarne Stroustrup em 1980, sendo chamada nesta época como C com classes. Atualmente ela é uma das principais linguagens de programação demonstrando-se ser altamente versátil e amplamente utilizada por uma grande parte de empresas desenvolvedoras de softwares (STROUSTRUP, 2000).

A seguir é descrito algumas vantagens da programação orientada a objetos em relação às abordagens convencionais:

- Fornecer uma estrutura modular para programas, facilitando a definição de tipos de dados abstratos onde detalhes de implementação estão ocultos e a unidade possui uma interface claramente definida;
- Tornar mais fácil a manutenção e a modificação de códigos, assim novos objetos podem ser criados a partir dos já existentes;
- Fornecer uma boa estrutura para bibliotecas de códigos onde os componentes de um software podem ser facilmente adaptados e modificados pelo programador.

Contudo, o desenvolvimento de aplicativos utilizando a linguagem C++ proporciona uma perda de 10% na velocidade de processamento em relação à linguagem C, devido principalmente as propriedades inerentes à orientação a objetos. Esta perda na eficiência pode variar dependendo da conscientização do programador. Alguns trabalhos (MALAN; LEWIS, 2004) foram realizados de forma a avaliar a aplicabilidade de códigos escritos em C++ na construção de algoritmos eficazes em cálculos de problemas CFD, aplicando algoritmos que reduzem o tráfego de dados e balanceando o polimorfismo do código com a eficiência computacional.

3.3.2 Desenvolvendo códigos no OpenFOAM

Uma das grandes vantagens do uso da linguagem matemática é a eficiência em expressar conceitos abstratos. Por exemplo, no escoamento de um fluido, o termo campo de velocidade possui um significado mesmo sem qualquer menção à natureza do escoamento ou qualquer dado específico de velocidade. O termo encapsula a ideia de movimento com direção e magnitude juntamente com outras propriedades físicas. Matematicamente podemos representar o campo de velocidades por um único símbolo, por exemplo, U e expressar certos conceitos usando símbolos, por exemplo, o campo de magnitude de velocidade como

$|U|$. Utilizar essa notação matemática possibilita expressar conceitos complexos com extrema clareza.

Com isso em mente os desenvolvedores do OpenFOAM buscaram utilizar as ferramentas disponíveis na linguagem C++ e técnicas de sobrecarregamento de operadores para criar uma simbologia matemática usual em seus códigos permitindo que fossem executadas operações sobre elas. Desta forma, foi possível utilizar uma sintaxe parecida com a notação matemática usual para representar as equações da mecânica do contínuo. As equações diferenciais parciais e os conceitos de escalares, vetores, tensores e seus respectivos campos, assim como a álgebra tensorial e sistemas de unidades podem ser invocados pelo OpenFOAM, facilitando em muito a implementação de novos solvers e também tornando muito mais ágil o desenvolvimento e a interpretação dos códigos pelo usuário.

As classes implementadas no OpenFOAM declaram tipos e operações que fazem parte da linguagem matemática utilizada na engenharia e no meio científico. O campo de velocidades apresentado anteriormente pode ser representado no código de programação do OpenFOAM através do símbolo U e a magnitude desse campo de velocidades é representado por $\text{mag}(U)$. Como já sabemos a velocidade é um campo vetorial e deve existir um código orientado a objetos com uma classe denominada `vectorField`, onde o campo de velocidade é visto como um objeto dessa classe.

A clareza no uso de objetos na programação para representar objetos físicos e entidades abstratas devem ser utilizadas com atenção. A própria estrutura das classes restringe o desenvolvimento do código dentro das próprias classes. Novas classes podem herdar propriedades de outras classes, como por exemplo, a classe `vectorField` que pode ser derivada de uma classe `vector` e/ou uma classe `Field`. O C++ fornece um mecanismo chamado de classes `template`, de forma que a classe `Field<Type>` pode representar um campo de qualquer `<type>`, como `scalar`, e `vector` e `tensor`. Os *templates* juntamente com a herança reduzem a duplicação de código e criam hierarquias de classe que impõe uma estrutura ao código.

O método de discretização utilizado no OpenFOAM é dos volumes finitos e as bibliotecas responsáveis pelo processo de aproximação dos termos derivativos das variáveis tensoriais são `fvm.H` e `fvc.H`. Cada termo nas equações diferenciais parciais é representado individualmente no OpenFOAM utilizando duas classes denominadas `finiteVolumeMethod` (`fvm.H`) e `finiteVolumeCalculus` (`fvc.H`). Ambas as classes `fvm.H` e `fvc.H` contêm funções estáticas que representam os operadores diferenciais, como gradiente, divergente e

laplaciano. Apesar de ambas as bibliotecas possuírem o mesmo propósito, suas aplicações são totalmente diferentes.

A Tabela 3.1 apresenta alguns exemplos de operadores importantes em CFD e suas respectivas representações no OpenFOAM.

Tabela 3.1: Principais palavras chaves usadas no arquivo fvSchemes.

Palavra Chave	Categoria do termo matemático
snGradSchemes	Componente do gradiente normal a face da célula
gradSchemes	Gradiente ∇
divSchemes	Divergente $\nabla \bullet$
laplacianSchemes	Laplaciano ∇^2
timeSchemes	Primeira e segunda derivada temporal $\partial/\partial t$ e $\partial^2/\partial^2 t$

Uma representação da discretização do domínio da solução é mostrada na Figura 3.3. O domínio do espaço é discretizado em uma malha computacional onde as equações diferenciais parciais são subsequentemente discretizadas. A discretização no tempo, se necessária, é simples: ela é dividida em um conjunto de passos Δt , que podem mudar durante uma simulação numérica, dependendo de alguma condição calculada durante o processo previamente simulado.

Figura 3.3: Discretização do domínio da solução. (Adaptado de: OpenFOAM *Programmer's Guide*)

Em um nível mais detalhado, a discretização no espaço requer a subdivisão do domínio em um número de células, ou volumes de controle. Essas células devem ser

contíguas, isto é, elas não devem sobrepor umas as outras e completar todo o domínio. Uma típica célula resultante da discretização do domínio é mostrada na Figura 3.4.

Figura 3.4: Parâmetros em uma discretização por volumes finitos. (Adaptado de: OpenFOAM *Programmer's Guide*)

O ponto P corresponde à célula de interesse, onde as variáveis dependentes e outras propriedades são principalmente armazenadas, embora também possam ser armazenadas nas faces ou vértices. O ponto N corresponde a célula vizinha, f é a face de comunicação entre as duas células, d é o vetor de P até N e S_f é um vetor que contém a área da face.

No OpenFOAM não há limitação quanto ao número de faces que limitam cada célula e nem qualquer restrição ao alinhamento de cada face. Este tipo de malha muitas vezes é denominada como "arbitrariamente desestruturada" para diferenciá-las de malhas no qual as faces das células tem um alinhamento prescrita, tipicamente com os eixos das coordenadas.

Na figura é mostrada uma célula típica resultante da discretização do domínio no OpenFOAM através do método dos volumes finitos. Códigos com malhas arbitrariamente não estruturadas oferecem uma maior liberdade na geração e manipulação da malha, em particular, quando a geometria do domínio é complexa ou sofrem alterações ao longo do tempo.

Como já mencionado, a utilização das classes no OpenFOAM permite que a sintaxe de escrita dos *solvers* se assemelhem à solução de equações diferenciais parciais. Por exemplo, considere a equação abaixo:

$$\frac{\partial \rho U}{\partial t} + \nabla \cdot \phi U - \nabla \cdot \mu \nabla U = -\nabla p. \quad (3.1)$$

Esta equação é representada em C++ no OpenFOAM como:

```
solve
(
    fvm::ddt(rho, U)
    + fvm::div(phi, U)
    - fvm::laplacian(mu, U)
    ==
    - fvc::grad(p)
);
```

Como pode ser observado, o código é simples de ser interpretado pelo usuário e sua representação se assemelha muito com a sintaxe matemática. Os códigos dos solvers são sequenciais já que representam um algoritmo de solução onde suas equações são sequenciais.

Os usuários não necessitam de um grande conhecimento de programação orientada a objetos e C++ para desenvolver um solver, mas devem conhecer bem os princípios por trás da orientação a objetos e ter um conhecimento da sintaxe de C++. O conhecimento das equações, modelos, métodos de discretização, solução e algoritmos é definitivamente mais importante.

Toda a implementação do código aplicado no OpenFOAM é baseado na teoria apresentada na tese de Jasak (JASAK, 1996). Mais dois materiais são recomendados para o aprofundamento no assunto, contendo detalhes sobre o código e aplicações práticas WELLER et al., 1998 e JASAK; WELLER; NORDIN, 2004.

3.4 Unidades dimensionais das propriedades físicas no OpenFOAM

Na mecânica do contínuo, as propriedades físicas são representadas por alguma unidade de referência, como por exemplo, massa é representada em quilogramas (kg), volume em metros cúbicos (m^3), pressão em Pascal ($Pa = kg.m.s^{-2}$). Assim as operações algébricas realizadas sobre essas propriedades utilizando unidades de medidas consistentes; em particular as operações de adição, subtração e igualdade são apenas fisicamente significativas para as propriedades de mesmas unidades dimensionais. Dessa forma o OpenFOAM trabalha com unidades no sistema internacional de medidas S.I. Um outro fator muito importante a notar quando se esta usando o OpenFOAM é que ele trabalha com unidades cinemáticas. A Tabela 3.2 apresenta o sistema de unidades que o OpenFOAM utiliza em seus códigos.

Tabela 3.2: Unidades de medida do OpenFOAM (S.I.)

No.	Propriedade	Unidade	Símbolo
1	Massa	Quilograma	kg
2	Comprimento	Metro	m
3	Tempo	segundo	s
4	Temperatura	Kelvin	K
5	Quantidade	Moles	mol
6	Corrente	Ampere	A
7	Intensidade luminosa	Candela	cd

CAPÍTULO 4 METODOLOGIA

Neste capítulo é apresentada a metodologia utilizada para o desenvolvimento do projeto. Inicialmente será descrito todo o desenvolvimento necessário para o processo de simulação do reator contínuo, desde a escolha de uma geometria representativa, passando pelo desenvolvimento das equações governantes do escoamento e a geração da malha computacional. Em seguida, é apresentada a metodologia relacionada ao reator *airlift*, apresentando a bancada experimental e os procedimentos e métodos utilizados para a aquisição dos dados experimentais. Expostos os procedimentos experimentais, são desenvolvidos os processos necessários para a simulação computacional do escoamento bifásico como a discretização das equações governantes, modelagem da turbulência e malha computacional. O software PyCFD-O, desenvolvido no presente trabalho, para o processo de acoplamento CFD - otimização multiobjetivo é descrito. Finalizando o capítulo, o problema de otimização multiobjetivo aplicado aos reatores contínuo e *airlift* é formulado bem como são definidas as variáveis de otimização e os processos matemáticos e computacionais envolvidos para a simulação com o software PyCFD-O.

4.1 Reator contínuo

Atualmente, o projeto de equipamentos na indústria de reatores segue duas abordagens distintas. A primeira abordagem segue uma linha de modelagem com base em aspectos empíricos ou semi-empíricos devido a uma longa história de análise dimensional na indústria de equipamentos. Por outro lado, a fluidodinâmica computacional (CFD), que é bastante utilizada em outros campos da ciência (como engenharia aeroespacial e automotiva), está se tornando cada vez mais interessante para os projetistas. Ambas as abordagens para projeto de equipamentos ainda coexistem e se complementam. Porém, com os recursos computacionais se tornando cada vez mais baratos e disponíveis, logo estas duas linhas se fundirão em uma única linha de simulação de projeto de reatores.

Como discutido, os parâmetros do escoamento em um reator têm influência direta sobre as reações químicas envolvidas no processo. Estes parâmetros podem ser o perfil de

velocidade, tempo de residência, tensão de cisalhamento, vorticidade, entre outros. Assim, uma análise adequada da influência da geometria do reator no escoamento pode aumentar consideravelmente a sua eficiência. Como citado em GÓIS, 2012, para o caso dos fermentadores aplicados para a produção de etanol a partir de cana-de-açúcar, devido as suas dimensões e as altas taxas de processamento, um pequeno aumento no rendimento do fermentador pode causar um elevado aumento na eficiência do processo químico. Um exemplo mencionado no mesmo trabalho considera que uma safra de dois milhões de toneladas de cana-de-açúcar pode ser convertida em 180 milhões de litros de etanol. Neste caso, um aumento de 1% no rendimento do fermentador representa 1,8 milhões de litros, resultando cerca de 20% e 30% a mais no lucro no resultado final (GÓIS, 2012).

Esta seção traz uma abordagem teórica dos fenômenos físicos e metodologia computacional aplicados no problema do reator contínuo.

4.1.1 Escolha de uma geometria inicial representativa

A escolha da geometria inicial para os estudos foi feita de forma a possibilitar uma análise direta da influência de alguns parâmetros geométricos na dinâmica do escoamento. A geometria adotada também possibilita que os resultados obtidos possam ser aproveitados para o caso de um fermentador contínuo para produção de etanol.

A fermentação contínua difere dos métodos tradicionais de fermentação por batelada, pois as matérias primas são continuamente alimentadas no reator, enquanto os produtos são continuamente retirados do sistema. Esta configuração apresenta diversas vantagens práticas como, por exemplo, a virtual eliminação dos "tempos mortos", i.e. período em que o equipamento não trabalha devido à necessidade de limpeza, esterilização, enchimento/esvaziamento, etc. As variáveis de operação, desde que sejam mantidas dentro de condições nominais, nessa configuração têm várias vantagens, como praticamente eliminando o tempo de inatividade inerente para limpeza e esterilização, e as defasagens para que os organismos entrem no período de alta produtividade. Além disso, tanto o CAPEX¹¹ quanto o OPEX¹² tendem a serem menores devido às simplicidades intrínsecas do equipamento. No entanto, torna-se necessário elaborar métodos para o desenvolvimento do projeto que possam garantir um desempenho adequado do reator, principalmente para casos de grande escala.

¹¹ **CAPEX**: Sigla em inglês de *capital expenditure* (despesas de capital ou investimento em bens de capital), montante de investimentos realizados em equipamentos e instalações.

¹² **OPEX**: Sigla em inglês de *Operational Expenditure*, que significa o capital utilizado para manter ou melhorar os bens físicos de uma empresa.

Entre as muitas causas da deterioração do desempenho para grandes escalas, provavelmente a dispersão do tempo de residência é uma das mais importantes. A razão para isto está relacionada ao fato de que em reatores contínuos de pequena escala a dissipação tende a prevalecer sobre os efeitos dinâmicos e, conseqüentemente, o campo de fluxo tende a ser mais organizado, ou até mesmo laminar. Assim, os tempos de residência associados a cada linha de corrente que liga a entrada e a saída apresentam variações de acordo com o projeto, relativamente ao tempo ideal de reação. Em um fermentador industrial, tipicamente dentro de 100 a 1000 metros cúbicos, os efeitos dinâmicos são preponderantes causando intensa turbulência, formação de grandes vórtices, etc., e as linhas de corrente tendem a se distribuir pelo volume do reator, aumentando significamente a dispersão dos tempos de residência. Assim, a quantidade de fluido que permanece o período adequado de tempo confinado no reator, isto é, o tempo de residência ideal para uma reação química eficaz, é reduzido significamente causando a redução do desempenho da conversão. Este aspecto particular será examinado a seguir no caso de teste em escala piloto (1m^3), bem como outros efeitos importantes relacionados à geometria do reator com o desempenho das reações envolvidas.

A configuração geométrica prospectiva para este estudo foi inspirada pelo tipo de reator industrial utilizado para a fermentação contínua da sacarose em etanol em algumas usinas de cana-de-açúcar no Brasil. O reator compreende um reservatório cilíndrico alimentado tangencialmente pela parte inferior e drenada a partir do topo através de um tubo de saída para garantir uma adequada exaustão de CO_2 . A Figura 4.1 ilustra uma representação esquemática da geometria inicial utilizada e na Tabela 4.1 é apresentada a parametrização correspondente e os valores iniciais utilizados para a definição de um volume interno de aproximadamente 1m^3 ou 1000 litros. Para restringir a otimização em um tempo de residência teórico pré-definido (volume do reator / vazão volumétrica), a altura (H_{Bio}) e o diâmetro (D_{Bio}) do reator permanecem constantes. Os diâmetros de entrada e saída (d_{in} , d_{out}) juntamente com o comprimento da tubulação de entrada (L_{in}) são fixados. Para o processo de otimização, as únicas variáveis de projeto alteradas são o ângulo de entrada (α) e a altura de saída (H_{out}).

As estimativas iniciais servem como aproximação inicial para as primeiras simulações. Segundo WILLIAMS; SAINI; WICK, 2002, fermentadores com formato cilíndrico permitem um crescimento homogêneo dos seres vivos utilizados como catalisadores da reação, pois garantem uma melhor tensão de cisalhamento.

Tabela 4.1: Valores da geometria inicial adotada: Reator contínuo.

Parâmetros	Valores
H_{Bio}	1,5 m
H_{out}	1 m
L_{in}	1 m
D_{Bio}	1 m
$d_{in} = d_{out}$	0,1 m
α	0°

Figura 4.1: Representação esquemática inicial do biorreator.

4.1.2 Simulação numérica do escoamento interno

Para o desenvolvimento da presente pesquisa é utilizado o pacote aberto de CFD OpenFOAM. Alguns detalhes dos métodos e a metodologia utilizados na resolução numérica do problema serão omitidos, podendo ser encontrados em detalhes nos manuais do OpenFOAM e na tese de JASAK, 1996.

4.1.2.1 Discretização das equações governantes

Os escoamentos em geral podem ser descritos matematicamente por um conjunto de equações de conservação da massa, quantidade de movimento e energia. Utilizando-se a notação vetorial elas podem ser escritas como:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0 \quad (4.1)$$

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u u) = \rho g + \nabla \cdot \sigma \quad (4.2)$$

$$\frac{\partial(\rho e)}{\partial t} + \nabla \cdot (\rho e u) = \rho g u + \nabla \cdot (\sigma u) - \nabla \cdot q + \rho Q_e \quad (4.3)$$

onde ρ é a densidade do fluido, u é o vetor velocidade, σ é o tensor das tensões, e é a energia específica total, Q_e o termo fonte de energia específica, q é o fluxo de calor e t é o tempo.

As leis de conservação acima são válidas para qualquer contínuo. O número de variáveis desconhecidas é maior que o número de equações do sistema, tornando o sistema indeterminado. Com o intuito de fechar o sistema, se faz necessário introduzir informações adicionais através das relações constitutivas. Para fluidos Newtonianos, o conjunto de equações a seguir é utilizado. A equação do estado relaciona a densidade com a pressão e temperatura

$$\rho = \rho(p, T). \quad (4.4)$$

O tensor de tensões σ é relacionado com a taxa de deformações do campo de velocidades

$$\sigma = -pI + \mu[\nabla u + (\nabla u)^T] + \lambda(\nabla \cdot u)I. \quad (4.5)$$

A Lei de Fourier da condução de calor relaciona o fluxo de calor com o gradiente do campo de temperatura

$$q = -k \nabla T. \quad (4.6)$$

A energia específica total é a soma da energia mecânica e_M (ou energia cinética) e a energia interna u , e pode ser expressa como

$$e = e_M + u(p, T) = \frac{1}{2} u u + u(p, T). \quad (4.7)$$

Inserindo a Eq. (4.5) na Eq. (4.2), e assumindo que $\lambda = \frac{2}{3}\mu$ obtem-se a equação de Navier-Stokes para escoamentos compressíveis.

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u u) = \rho g - \nabla \left(p + \frac{2}{3} \mu \nabla \cdot u \right) + \nabla \cdot [\mu (\nabla u + (\nabla u)^T)]. \quad (4.8)$$

Inserindo as Eqs. (4.5) e (4.6) na Eq. (4.3) obtem-se a equação de conservação da energia para fluidos compressíveis

$$\frac{\partial(\rho e)}{\partial t} + \nabla \cdot (\rho e u) = \rho g u - \nabla \cdot (p u) - \nabla \cdot \left(\frac{2}{3} \mu (\nabla \cdot u) u \right) + \nabla \cdot [\mu (\nabla u + (\nabla u)^T)] + \nabla \cdot (k \nabla T) + \rho Q. \quad (4.9)$$

Os coeficientes de transporte da viscosidade e da condução são funções das variáveis de estado termodinâmicas, ou seja, $\mu = \mu(p, T)$ e $k = k(p, T)$.

Para problemas isotérmicos e onde a densidade é considerada constante (incompressível), as equações governantes podem ser simplificadas para:

$$\nabla \cdot u = 0 \quad (4.10)$$

$$\frac{\partial u}{\partial t} + \nabla \cdot (u u) = -\nabla p + g + \nabla \cdot (\nu \nabla u), \quad (4.11)$$

onde ν é a viscosidade cinemática e p é a pressão cinemática.

4.1.2.2 Modelagem da turbulência

A maioria dos escoamentos de fluidos que ocorrem na natureza é turbulenta. A turbulência pode ser descrita como um estado contínuo de instabilidade no escoamento e é caracterizada pela irregularidade do escoamento, aumento da difusividade e dissipação de energia (JASAK, 1996; TENNEKES; LUMLEY, 1972)

Existem várias possibilidades de simular escoamentos turbulentos. A primeira é através do método *Direct Numerical Simulation* (DNS) que integra numericamente as equações governantes sobre toda a gama de escalas de turbulência. Os requisitos para que isso seja possível são malhas extremamente refinadas e passos no tempo muito pequenos, requerendo muitos recursos computacionais, tornando-o inadequado para muitas aplicações na engenharia. As outras técnicas tratam a turbulência através da proposição de modelos de

turbulência como fechamento das equações médias espaciais e temporais de Navier-Stokes, denominadas RANS (*Reynolds Averaged Navier-Stokes Equations*). Entre os modelos mais difundidos nesta classe, e que usam duas equações diferenciais parciais adicionais, estão o $k - \varepsilon$ e suas variantes, o $k - \omega$ e o *Shear Stress Transport*.

Outro modelo bastante utilizado que se situa entre o DNS e o RANS é o *Large eddy simulation* (LES). No LES é utilizado um filtro que separa as grandes estruturas da turbulência, ou seja, os grandes vórtices que transportam energia e quantidade de movimento, das pequenas estruturas, representados pelos pequenos vórtices, onde prevalece a isotropia (SOUZA et al., 2011).

Será dada ênfase nos modelos RANS que são utilizados nos testes no presente trabalho. O RANS assume que as variáveis do escoamento podem ser decompostas entre componentes de uma média em um certo período de tempo e flutuações, por exemplo, para o campo de velocidades

$$u(x,t) = \bar{u}(x) + u'(x,t) \quad (4.12)$$

onde $\mathbf{x} = (x, y, z)$ é o vetor posição, $u(x,t)$ corresponde à velocidade local instantânea, $\bar{u}(x)$ é a velocidade média local e $u'(x,t)$ são as flutuações de velocidade devido a turbulência (vide Figura 4.2).

Figura 4.2: Gráfico dos componentes locais de velocidade.

Substituindo-se a Eq. (4.12) nas Eqs. (4.10) e (4.11), e tendo-se os tempos médios das equações resultantes obtém-se as equações RANS

$$\nabla \cdot \bar{\mathbf{u}} = 0 \quad (4.13)$$

$$\frac{\partial \bar{\mathbf{u}}}{\partial t} + \nabla \cdot (\bar{\mathbf{u}} \bar{\mathbf{u}}) = -\frac{1}{\rho} \nabla p + \mathbf{g} + \nabla \cdot (\nu \nabla \bar{\mathbf{u}}) - \nabla \cdot \overline{\mathbf{u}' \mathbf{u}'} \quad (4.14)$$

O termo $\overline{u'u'}$ é chamado de tensor de tensões de Reynolds. Para o fechamento do sistema faz-se necessária a modelagem do tensor das tensões de Reynolds. O modelo mais popular utiliza a aproximação de Boussinesq, que descreve uma relação linear da seguinte forma:

$$-\overline{u'u'} = \nu_t (\nabla \bar{u} + (\nabla \bar{u})^T) + \frac{2}{3} k I, \quad (4.15)$$

onde a energia cinemática turbulenta, k , é definida como:

$$k = \frac{1}{2} \overline{u'u'}. \quad (4.16)$$

A viscosidade cinemática turbulenta (ν_t) pode ser calculada de várias maneiras. O OpenFOAM possui alguns modelos implementados para esse cálculo como Spalart Allmaras, $k - \varepsilon$, $k - \omega$ e $k - \omega SST$. A seguir, uma breve descrição de cada modelo é apresentada:

- Spalart Allmaras (Modelo de uma equação): Modelo relativamente simples que resolve uma equação diferencial de transporte para a viscosidade turbilhonar.
- $k - \varepsilon$ (Modelo de duas equações): Um dos modelos mais utilizados, onde o transporte de dois escalares é descrito por duas equações (k energia cinética turbulenta (m^2/s^2) e a taxa de dissipação de energia cinética turbulenta (m^2/s^3)). O tensor de tensões de Reynolds é calculado através de uma hipótese que relaciona o tensor com o gradiente da velocidade média e a viscosidade turbulenta.
- $k - \omega$ (Modelo de duas equações): O modelo também utiliza duas equações, uma para o k , calculada da mesma maneira que para o modelo $k - \varepsilon$, e uma para o ω , que é descrito como uma frequência turbulenta.
- $k - \omega SST$ (MENTER; ESCH, 2001): O modelo busca conciliar as vantagens do modelo $k - \varepsilon$, que apresenta um melhor desempenho em regiões completamente turbulentas afastadas da superfície do perfil com as do modelo $k - \omega$, que exibe um melhor desempenho nas regiões próximas a parede. Cada modelo é aplicado onde seus resultados são mais precisos. Para isso, uma função de transição que realiza a mudança entre os dois modelos é inserida, evitando instabilidades.

A complexidade, a capacidade de modelagem e o custo computacional aumentam nos modelos de turbulência descritos acima, na mesma ordem em que foram listados. Com os avanços computacionais, torna-se cada vez mais possível priorizar os modelos mais precisos.

O modelo $k - \varepsilon$, apesar de somente tratar turbulência isotrópica, é um dos modelos mais populares e tem apresentado resultados satisfatórios para diferentes situações (VERSTEEG; MALALASEKERA, 2007). O modelo é válido para escoamentos completamente turbulentos. As escalas de turbulência são diretamente influenciadas pelo valor da energia cinética turbulenta, k , e da taxa de dissipação, ε . Dessa forma, existe um acoplamento para o cálculo das variáveis que permite calcular a viscosidade turbulenta local, Equação (4.17):

$$\nu_t = C_\mu \frac{k^2}{\varepsilon}, \quad (4.17)$$

$$\varepsilon = \overline{\nu u' u' : \nabla u'}. \quad (4.18)$$

4.1.2.3 Discretização em volumes finitos

As equações de conservação de massa, quantidade de movimento e transporte de escalares podem ser representados pela equação de transporte genérica:

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho u \phi) - \nabla \cdot (\rho \Gamma_\phi \nabla \phi) = S_\phi(\phi) \quad (4.19)$$

que é composta por quatro termos: aceleração temporal, aceleração convectiva, difusão e um termo fonte (da esquerda para a direita). O método dos volumes finitos requer que cada termo na Eq. (4.19) seja satisfeito sobre um volume de controle V_p , ao redor de um ponto P em uma integral da forma:

$$\int_t^{t+\Delta t} \left[\frac{\partial}{\partial t} \int_{V_p} \rho \phi dV + \int_{V_p} \nabla \cdot (\rho u \phi) dV - \int_{V_p} \nabla \cdot (\rho \Gamma_\phi \nabla \phi) dV \right] dt = \int_t^{t+\Delta t} \left[\int_{V_p} S_\phi(\phi) dv \right] dt. \quad (4.20)$$

Assim, o método de volumes finitos consiste em se obter a aproximação numérica da equação diferencial dada pela Eq. (4.19) a partir de sua integração em um volume de controle elementar. A discretização espacial é feita integrando todos os termos da equação no espaço para cada volume de controle no domínio. O resultado final é a equação discretizada para um grupo de pontos de uma malha.

Como descrito no CAPÍTULO 3 o OpenFOAM utiliza o método dos volumes finitos em seu código. Para maiores detalhes sobre a discretização utilizando-se o método dos volumes finitos, as funções de interpolações, métodos acoplamento pressão-velocidade e os métodos de resolução dos sistemas lineares implementados no pacote CFD OpenFOAM recomenda-se consultar JASAK, 1996 e RUSCHE, H., 2002.

4.1.2.4 Solver desenvolvido no OpenFOAM para a simulação computacional do reator contínuo

Para as simulações do reator contínuo foram adotadas algumas hipóteses simplificadoras. Dentre essas hipóteses podemos descrever: escoamento incompressível, isotérmico e transiente, juntamente com fluido newtoniano. A partir dessas hipóteses optou-se por utilizar o solver *pisoFoam* contido no pacote CFD OpenFOAM. O solver *pisoFoam* utiliza o algoritmo PISO (*Pressure-Implicit with Splitting of Operator*) como método de acoplamento pressão-velocidade. O algoritmo pode ser utilizado no OpenFOAM para problemas transientes e está baseado em procedimentos iterativos, capazes de avaliar a solução num dado instante de tempo e corrigi-la.

Como critério de estabilidade é utilizado o adimensional denominado número de Courant (Co) que caracteriza a extensão relativa das oscilações numéricas. O número de Courant pode ser definido como uma variável local do domínio fluido que representa o fluxo numérico advectivo adimensionalizado pelas malhas espacial e temporal em cada volume. Por definição o número de Courant é dado pela Equação (4.21):

$$Co = \frac{|u|\Delta t}{\Delta x} \quad (4.21)$$

onde $|u|$ é a magnitude da velocidade local, Δt o passo no tempo e Δx o comprimento característico de um elemento de malha. O número de Courant pode ser interpretado como a razão entre a distância percorrida por uma onda de perturbação em um passo de tempo e o tamanho local dos volumes de malha.

Um dos critérios de convergência mais comum para obtenção de precisão física e estabilidade do cálculo é o parâmetro CFL (Courant-Friedrichs-Lewy). Este parâmetro é a condição de limitação do número de Courant, ou seja, ele é um multiplicador do passo de tempo que fornece um número de Courant igual à unidade. Por exemplo, um cálculo transiente explícito (sem iterações no espaço) exige que o número de Courant de cada volume

da malha seja inferior à unidade em todo o domínio. Então este cálculo está submetido à condição de um CFL inferior a um, definindo a Condição CFL (FORTUNA, 2000). Assim, uma das maneiras de garantir que um modelo seja estável, é garantindo que a distância percorrida durante o passo de cálculo do modelo seja inferior à dimensão das células do mesmo, ou seja, inferior à solução com que o espaço é resolvido no modelo. Portanto, Δt tem que ser tão menor quanto for o Δx . No entanto, se o número de Courant é demasiado pequeno, muito tempo computacional será usado. Sendo assim, o número de Courant é um parâmetro muito importante que afeta o custo computacional e a estabilidade da simulação.

Pode-se utilizar o número de Courant para controlar o intervalo de tempo na simulação transiente em códigos CFD. Assim, visando garantir a qualidade das soluções e o custo computacional do modelo numérico, o solver base *pisoFoam* foi adaptado de modo a permitir ajustar o intervalo de tempo durante a simulação de acordo com o número máximo de Courant. As modificações consistem inicialmente em modificar as opções no arquivo de controle *controlDict* habilitando a função *adjustTimeStep*, definindo-se o número máximo de Courant (*maxCo*) e limitando-se o tamanho máximo do passo no tempo (*maxDeltaT*). No entanto, estas são somente as modificações na configuração do caso. O código fonte principal do solver *pisoFoam* deve ser modificado para permitir o passo do tempo ajustável. O código fonte do solver com as modificações realizadas são apresentados no APÊNDICE B.

O próximo passo para o desenvolvimento do solver aplicado ao problema do reator contínuo foi possibilitar o cálculo da partícula traçadora. A partícula traçadora computacional consiste em uma partícula passiva fluida, ou seja, ela apresenta as mesmas características do fluido e sem difusividade. Para realizar isto no OpenFOAM foi adicionado no solver uma variável de campo denominada *C*. O cabeçalho *createField.H* é onde as informações específicas referentes as variáveis que serão resolvidas se encontram. Os primeiros itens carregados em uma simulação são as propriedades do fluido, como por exemplo, a viscosidade cinemática que se encontra no dicionário *transportProperties*. Neste dicionário adiciona-se a nova propriedade de transporte relacionada com a difusividade da partícula que será indicada com *DC*, como observado na Figura 4.3-A. Pela definição de partícula passiva, a difusividade é nula, no entanto, esta variável deve ser criada e o seu valor é definido no dicionário *transportProperties* nas configurações do problema.

Posteriormente, é necessário editar o arquivo de dicionário para criar o campo *C* correspondente à partícula traçadora, como ilustrado na Figura 4.3-B.

```

A)
15 // --- Scalar Diffusion
16 dimensionedScalar DC
17 (
18     transportProperties.lookup("DC")
19 );

B)
51 // --- Scalar
52 Info<< "Reading field C\n" << endl;
53 volScalarField C
54 (
55     IOobject
56     (
57         "C",
58         runTime.timeName(),
59         mesh,
60         IOobject::MUST_READ,
61         IOobject::AUTO_WRITE
62     ),
63     mesh
64 );

```

Figura 4.3: Definição do campo da partícula traçadora: *createField.H*.

O próximo passo é a inclusão de uma nova equação que descreva o transporte da partícula traçadora. Para isso, é necessário editar o arquivo-fonte “.C”. Uma vez que o transporte da partícula depende do campo de velocidade, o cálculo da equação de transporte deve ser realizado a posteriore, i.e. após a solução da equação de movimento (depois do laço PISO), mas antes que a integração no tempo seja executada. A Figura 4.4 apresenta o trecho do código-fonte do solver correspondente a implementação da equação de transporte.

```

134 // --- Scalar Transport
135 Info<< "\nCalculating scalar transport\n" << endl;
136 C.storePrevIter();
137 for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
138 {
139     volScalarField DCEff = DC + turbulence->nut()/0.7;
140     fvScalarMatrix TEqn
141     (
142         fvm::ddt(C) + fvm::div(phi, C) -
143         fvm::laplacian(DCEff, C)
144     );
145     TEqn.relax();
146     TEqn.solve();
147 }
148
149 runTime.write();

```

Figura 4.4: Cálculo da equação de transporte para a partícula traçadora: *pisoScalarFoamAdjTime*.

Uma vez desenvolvido o solver *pisoScalarFoamAdjTime*, é necessário compilar o código-fonte com o *script wmake*¹³ que permite acessar componentes dependentes da biblioteca do OpenFOAM em C++ e criar os *links* dinâmicos. Ao compilar o código-fonte o solver está pronto para ser executado.

¹³ Para maiores detalhes do processo de compilação das aplicações e bibliotecas no openFOAM verifique o manual no seguinte link: <http://www.openfoam.org/docs/user/compiling-applications.php>

4.1.2.5 Malha computacional – Reator contínuo

Para a solução de um problema de fluidodinâmica computacional uma das etapas básicas é o processo da geração da geometria e da malha. Nesta fase são definidos os parâmetros geométricos do problema. A escolha da malha computacional deve ser feita de forma que permita a solução numérica com a resolução necessária com o menor custo computacional.

Todas as malhas referentes ao reator contínuo foram geradas pelo software SALOME através da importação da geometria em um formato CAD “.step”. No software SALOME foram definidas as faces para a aplicação das condições de contorno. Dentre os algoritmos de geração de malha, o que mais se adequou às necessidades de refinamento e as propriedades da geometria foi a malha tetraédrica. O algoritmo de geração de malha utilizado foi o NETGEN (SCHOBRL, 1997). A Figura 4.5 apresenta um exemplo de malha tetraédrica gerada para as simulações do reator. As especificações das malhas geradas, tais como o número de volumes tetraédricos são apresentadas na seção de estudo de convergência de malha (Seção 5.1.1).

Após a geração da malha, a exportação se dá através do formato “.unv”, sendo as propriedades da malha verificadas no OpenFOAM. Todo o processo de criação da geometria e geração da malha computacional, assim como a verificação da qualidade da malha é feito automaticamente pelo software PyCFD-O, como será descrito na Seção 4.3.

Figura 4.5: Exemplo de malha computacional tetraédrica utilizada: Reator contínuo.

4.2 Reator *Airlift*

4.2.1 Bancada experimental

Os fotobiorreatores do tipo placa têm recebido muita atenção para o cultivo de micro-organismos fotossintéticos (UGWU; AOYAGI; UCHIYAMA, 2008; ZHANG; KURANO; MIYACHI, 2002), devido a certas características tais como: grande superfície de iluminação, adequação para culturas ao ar livre, bons trajetos de iluminação, boa produtividade de biomassa, processo de produção relativamente barato, fácil limpeza. No entanto, esses tipos de reatores também possuem certas limitações, como: difícil projeto de aumento de escala (*scale-up*), dificuldade no controle da temperatura da cultura, algum grau de crescimento de algas na parede, tensões hidrodinâmicas prejudiciais para algumas cepas de algas.

Dadas essas características, um reator *airlift* de circulação interna do tipo placa foi escolhido como um dos objetos de estudo do presente trabalho. A aplicação de análises CFD combinadas com a otimização da geometria possibilita uma melhoria do rendimento deste tipo de reator. Deste modo, os experimentos foram efetuados utilizando um reator *airlift* de circulação interna do tipo placa com um volume total de 61,5 litros. Um desenho esquemático do reator é ilustrado na Figura 4.6.

Modelos de fotobiorreatores *airlift* de circulação interna similares ao utilizado neste trabalho foram aplicados para o cultivo de microalgas em vários trabalhos (GRIS, 2011; RAMIREZ; FARENZENA; TRIERWEILER, 2014; TREDICI, 2003 YU et al., 2009). No entanto, nesses trabalhos são utilizados fotobiorreatores em escala reduzida. GRIS, 2011 e RAMIREZ; FARENZENA; TRIERWEILER, 2014 aplicaram este tipo de fotobiorreator para o cultivo de microalgas para o tratamento de vinhaça.

O reator utilizado no presente trabalho foi confeccionado inteiramente em material transparente (acrílico) e os injetores em aço inoxidável. O reator é composto por onze injetores que foram introduzidos em furos no acrílico localizados na parte inferior. Os orifícios de cada injetor apresentam 0,8 mm de diâmetro e são igualmente espaçados entre si ao longo do comprimento do reator com uma distância de 15 mm da parede do reator. A saída de ar dos injetores ocorre na parte inferior do reator. A Figura 4.7 apresenta uma fotografia do reator *airlift*, mostrando a câmara para distribuição de ar nos injetores.

Câmara de ar

O suprimento de ar à linha é feito por um compressor do tipo parafuso da Schulz Modelo 3030 com vazão efetiva de 3511 l/min e pressão de saída de 7,5 bar. O ar é direcionado a tanques que asseguram a manutenção da pressão e vazão de ar nas linhas de teste. Para a medição de vazão de ar foi utilizado um rotâmetro, modelo RMA-5-SSV da

Dwyer com precisão de $\pm 4\%$ da leitura de fundo de escala. A vazão total de ar na entrada da câmara de ar do reator para os todos os experimentos foi 5 l/min. Na Figura 4.8 é apresentada uma representação esquemática da bancada experimental.

Água a temperatura ambiente e ar comprimido são os fluidos utilizados para os experimentos e as simulações. Assumiu-se que a água seria capaz de imitar uma cultura líquida de algas. Essa suposição foi feita devido à pequena diferença da densidade e viscosidade entre a água e uma cultura de algas (DEL NINNO, 2012, POSTEN; WALTER, 2012). Isto é possível devido a densidade específica da alga ser próxima de 1 e a proporção em peso seco de algas em água ser próximo de 0,1%. Portanto, isto mostra que a cultura de algas é, de fato, principalmente água e tem propriedades de fluidos semelhantes. Assim torna-se uma aproximação confortável assumir que os fluidos vão agir sobre o mesmo carregamento dinâmico.

Figura 4.8: Representação esquemática: Bancada experimental. (1) Compressor; (2) tanque de armazenamento de ar; (3) Válvula de abertura de ar comprimido; (4) Válvula de regulação; (5) Rotâmetro.

4.2.2 Estudos hidrodinâmicos realizados no reator *airlift*

A seguir, é apresentada a metodologia utilizada na determinação de alguns dos principais parâmetros hidrodinâmicos, tais como: tempo de mistura e circulação e os campos vetoriais.

4.2.2.1 Transferência de movimento: Tempo de mistura e circulação

A obtenção dos tempos de mistura e circulação de um reator *airlift* é importante, pois caracterizam o reator do ponto de vista dinâmico e de homogeneização. O tempo de

circulação é definido pelo tempo necessário para que ocorra a circulação completa de um elemento de volume da fase líquida contida no interior do reator, sendo utilizado no cálculo da velocidade superficial do líquido (CHISTI, 1989). O tempo de mistura, diferentemente do tempo de circulação, é o tempo necessário para que uma substância se misture igualmente no interior do biorreator. O tempo de mistura é definido como o tempo necessário para que uma variável atinja 95% do valor final.

O tempo de mistura é um parâmetro muito importante na concepção de fotobioreactores para vários processos biológicos. Uma mistura adequada do meio de cultura pode garantir uma alta concentração de células, manter as células de algas em suspensão, eliminar estratificação térmica, ajudar distribuição de nutrientes, melhorar as trocas gasosas, bem como reduzir o grau de sombreamento mútuo e diminuir a probabilidade de fotoinibição (UGWU; AOYAGI; UCHIYAMA, 2008 JAVANMARDIAN; PALSSON, 1991).

Para determinar esses parâmetros foi utilizada a técnica do pulso de pH. Esta técnica consiste na injeção de um ácido-traço e o acompanhamento da medida da variação do pH em um ponto específico do reator, determinando-se assim os tempos de mistura e de circulação (CERRI, 2005; CHISTI, 1989). Nos experimentos foram utilizados soluções de HCl ou NaOH (10 M) e acompanhou-se o comportamento e estabilização desta variável. O pulso de pH foi injetado em um ponto na região base do reator (Figura 4.9). Após isso, a variação de pH foi acompanhada através de um eletrodo de pH DME-CV2ET da marca Digimed (faixa operação pH entre 0 e 14 e temperatura 0 e 130°C), imerso na região de descida (*downcomer*), como ilustrado na Figura 4.9. Foram feitas medições com o eletrodo em três pontos diferentes sobre uma reta no *downcomer*, como observado na Figura 4.9 e na Tabela 4.2.

Para a aquisição dos dados foram utilizados um sistema *in-house* composto por uma placa de aquisição da *National Instruments* modelo NI USB-6009 e um computador com programas em LabVIEW®. O sistema *in-house* também permite o monitoramento da temperatura da mistura através de um termopar. O sistema de aquisição se comunica com o computador via USB e trabalham com entradas e saídas analógicas e digitais. A transferência de dados é organizada em intervalos de tempo definidos e arquivados de acordo com as especificações do usuário. A Figura 4.10 ilustra uma imagem fotográfica da bancada experimental utilizada para as medições do pulso de pH.

Figura 4.9: Diagrama esquemático sistema de aquisição pulso de pH: (1) Ponto de injeção traçador; (2) Eletrodo de pH; (3) Termopar; (4) Placas de aquisição; (5) Software.

Tabela 4.2: Pontos de monitoramento pulso pH.

Pontos de monitoramento	
Ponto 1 (A)	(0,33, 0,095, 0,285)
Ponto 2	(0,33, 0,135, 0,285)
Ponto 3 (B)	(0,33, 0,155, 0,285)

Figura 4.10: Fotografia da bancada experimental pulso pH.

Para determinação do tempo de circulação, além da técnica do pulso de pH aplicou-se o método da esfera (VASCONCELOS et al., 2003 CERRI, 2005). Neste método, uma pequena esfera com densidade de aproximadamente 1 g.cm^{-3} é inserida no biorreator sob aeração e o tempo de cada volta desta partícula é cronometrado. Após a obtenção de várias

medidas determina-se a média de tempo necessário para completar uma volta, sendo este parâmetro definido como tempo de circulação do líquido.

Foi utilizada como partícula uma esfera de silicone esmaltada de diâmetro médio de 4 mm, sendo a densidade regulada pela película de esmalte, de modo que ela seja próxima a 1 g.cm^{-3} . A Figura 4.11 apresenta uma foto das partículas traçadoras utilizadas nos experimentos.

Figura 4.11: Partícula traçadora.

4.2.2.2 Determinação dos campos vetoriais de velocidade

Os dados referentes à fase líquida do reator *airlift* foram obtidos por meio da aplicação da técnica de velocimetria por imagem de partícula (PIV), que calcula os campos instantâneos de velocidade através da análise da distância percorrida por partículas inseridas no escoamento. A obtenção das imagens que permitam o cálculo real da trajetória percorrida pelas partículas é essencial e está diretamente ligada à fidelidade dos resultados.

Para determinar os campos de velocidade do escoamento foram aplicadas duas técnicas de PIV. Um primeiro experimento foi realizado utilizando-se um sistema *in-house* de aquisição de imagens e em seguida foi utilizado um sistema comercial da marca LaVision.

A técnica denominada como PIV *in-house* foi realizada com o auxílio de uma câmera de alta velocidade Olympus *i-speed* 3, focada em uma região lateral do reator. Devido a natureza geométrica do reator (faces planas) não foi necessário realizar nenhuma adaptação no reator para corrigir distorções na imagem devido a difração. Como partículas traçadoras foram utilizadas 50 partículas dos mesmos traçadores descritos na Seção 4.2.2.1, por apresentarem um bom contraste em relação ao fluido analisado. Outro ponto de extrema importância para a aquisição das imagens diz respeito à iluminação. A iluminação necessária para registrar as imagens foi fornecida por duas lâmpadas de LEDs de alta capacidade (20W de potência e 2300 lúmen). As lâmpadas foram posicionadas de tal forma a possibilitar aquisições de imagens no plano desejado com um contraste nas partículas e uma menor interferência externa. Testes realizados com a câmera revelaram que uma velocidade de 50

frames por segundo é suficiente para observar o escoamento. Após a aquisição das imagens torna-se necessário a utilização de um software para o processamento das mesmas. A qualidade das medições dos campos vetoriais é afetada por detalhes computacionais, tais como, pré-processamento da imagem, estimadores de picos de sub-pixel, procedimento de validação dos dados, algoritmos de interpolação e métodos de nivelamento (THIELICKE; STAMHUIS, 2014). Para isto, foi utilizada uma ferramenta aberta para a realização de análise de fluxo de PIV em MATLAB chamado PIVLab¹⁴ (THIELICKE; STAMHUIS, 2014).

As aquisições realizadas com o PIV *in-house* permitem uma análise qualitativa do escoamento, possibilitando a comparação das características principais do escoamento com os resultados do modelo CFD, como por exemplo, as regiões de recirculação.

A Figura 4.12 apresenta um desenho esquemático da bancada experimental desenvolvida e os principais equipamentos utilizados para as aquisições das imagens utilizando-se o PIV *in-house*. No início dos experimentos, uma imagem de calibração do reator foi capturada para possibilitar a correspondência entre pixels e milímetros, necessário para o processamento das imagens no software PIVLab.

A Figura 4.13-A apresenta uma fotografia da bancada de testes e na Figura 4.13-B tem-se um exemplo de imagem obtida com a câmera de alta velocidade utilizada para a análise PIV *in-house*.

A utilização do equipamento profissional da LaVision envolve um maior aparato de equipamentos. O modelo do equipamento utilizado é o PIV Imager Pro SX 5M SYSTEM, composto por uma câmera de alta performance com sistema CCD modelo PCO1600, cookeCorp, laser pulsado duplo Nd:YAG, lente óptica e um computador com software DAVIS 8.2, para controle do sistema, aquisição e processamento dos dados.

Durante os experimentos, o plano de estudo é iluminado duas vezes pelo laser e a luz emitida pelas partículas traçadoras é captada a uma frequência de 6,9 Hz por uma lente de alta qualidade em duas imagens separadas, através de um sensor CCD. Essas imagens capturadas são enviadas para o computador através de uma placa de aquisição de sinais. É assumido que as partículas movem-se com a velocidade local do fluido entre os dois pulsos de iluminação. A partir disso as imagens são pré-processadas e em seguida analisadas pelo software Davis da LaVision. A Figura 4.14 apresenta um diagrama esquemático da bancada experimental utilizada para a aquisição dos dados com o PIV LaVision. A Figura 4.15-A mostra uma foto da bancada experimental com a distribuição dos equipamentos.

¹⁴ Programa disponível no site do [MATLAB CENTRAL](http://pivlab.blogspot.com.br/) ou pelo link: <http://pivlab.blogspot.com.br/>

Figura 4.12: Desenho esquemático da bancada experimental: PIV *in-house*: (1) Câmera de alta velocidade; (2) Lâmpadas de LEDs; (3) Computador.

Figura 4.13: Fotografias PIV *in-house*: A) Bancada experimental; B) Imagem para análise qualitativa.

Os experimentos foram planejados com o objetivo de avaliar a evolução do perfil de velocidade da fase líquida em um plano axial no reator, de modo a possibilitar a distribuição de velocidade na região de subida e descida do reator. A região de análise é destacada na Figura 4.14 e Figura 4.15-B. A disposição do laser aplicado no experimento é como a representada na Figura 4.14 e Figura 4.15-A, pois as bolhas (fração volumétrica global da fase gasosa) e a potência do laser podem comprometer a reflexão da luz do laser e a visualização das partículas do escoamento. Portanto, testes preliminares foram necessários para definição da melhor disposição do laser. O sistema que mostrou melhores resultados foi o obtido com o laser incidindo inicialmente na região de descida, onde não há ocorrência de bolhas, como

pode ser observado na Figura 4.14. A Figura 4.15-B ilustra um exemplo de aquisição realizada com a aplicação do laser e a presença do traçador. As partículas traçadoras utilizadas são esferas ocas de vidro 110P8 diâmetro médio 10 – 100 μm que acompanham o equipamento da LaVision.

Figura 4.14: Desenho esquemático da bancada experimental: PIV LaVision. (1) Câmera CCD; (2) Lente óptica; (3) Laser pulsado duplo; (4) Computador.

Figura 4.15: Fotografias PIV LaVision: A) Bancada experimental; B) Imagem para processamento.

No início dos experimentos é necessário calibrar o sistema de medição no local onde serão captadas as imagens. A posição da câmera e do feixe de luz de laser deve ser perpendicular ao escoamento conforme esquematizado na Figura 4.15-A. As operações de

calibração, aquisição e processamento de dados são realizadas e acompanhadas por meio do software DAVIS.

Todos os experimentos foram realizados com as condições de operação descritas na Seção 4.2.1 e com vazão total de injeção de ar na câmara de 5 l/min.

4.2.3 Simulação numérica do escoamento

Nesta seção serão apresentados os modelos computacionais adotados para a solução numérica do escoamento bifásico água-ar no reator *airlift*. Como já mencionado, para a solução computacional das equações governantes dos escoamentos foi utilizado o software OpenFOAM. Como recomendado por BITOG et al., 2011, para o reator *airlift* o método VOF (*Volume of fluid*) é utilizado para análise do escoamento água-ar. O método VOF foi apresentado por HIRT; NICHOLS, 1981 e iniciou uma nova tendência na simulação de escoamentos multifásicos. Baseia-se na definição de uma função indicadora. Esta função permite saber se a célula é ocupada por um fluido ou por outro, ou uma mistura de ambos. Existem várias maneiras de implementar este conceito, mas nesta seção é apresentada a metodologia adotada no âmbito do OpenFOAM. A metodologia é descrita na íntegra por UBBINK, 1997 e RUSCHE, H., 2002 e uma explanação concisa é feita por BERBEROVIĆ et al., 2009. A seguir são citados alguns trechos destes trabalhos juntamente com uma explicação do laço de acoplamento pressão-velocidade, do modelo de turbulência, desenvolvimento do solver no âmbito do OpenFOAM e a malha computacional.

4.2.3.1 Discretização das equações governantes

Considerando a validade de princípios físicos da conservação de massa e da quantidade de movimento, para o problema do reator *airlift* é necessário aplicar um modelo multifásico de forma a representar as diferentes fases ou misturas dos fluidos presentes no escoamento.

A metodologia numérica empregada neste trabalho é baseada no método de volumes finitos (FVM), utilizando-se como ferramenta de discretização e análise numérica o pacote OpenFOAM. Para o reator *airlift* o método VOF (*Volume of fluid*) é aplicado para análise do escoamento água-ar. O método VOF foi originalmente introduzido por HIRT; NICHOLS, 1981 e pode ser aplicado a problemas com vários fluidos com diferentes densidades, sendo apropriado para simulações de interfaces interpenetrantes entre fluido-fluido (HORVATH et

al., 2009). Uma grande desvantagem da abordagem Euleriana-Euleriana frente ao método VOF é a interação das fases e o tamanho das bolhas, o qual é definido pelo usuário. Coalescência das bolhas e a dispersão (*ab-initio*) são difíceis a impossível de modelar utilizando-se o método de Euler (HORVATH et al., 2009). A modelagem das fases gasosa e líquida através de conceitos eulerianos (homogêneos) impede a distinção entre a ocorrência de cada fase, fornecendo uma concentração volumétrica de fases que reflete uma média volumétrica temporal de sua ocorrência. A modelagem correta da fase gasosa em reatores *airlift* tem influência direta no transporte de massa no reator, definindo assim a sua eficiência.

No método convencional do volume de fluido (HIRT; NICHOLS, 1981), uma equação de transporte para uma função indicadora, que representa a fração de volume de uma fase, é resolvida simultaneamente com as equações de continuidade e momento:

$$\nabla \cdot u = 0 \quad (4.22)$$

$$\frac{\partial \gamma}{\partial t} + \nabla \cdot (u\gamma) = 0 \quad (4.23)$$

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u u) = -\nabla p + \nabla \cdot D + \rho f_b \quad (4.24)$$

onde u representa o campo de velocidade compartilhada pelos dois fluidos em todo o domínio do escoamento, γ é a fração de fase, D representa o tensor de tensões $D = 2\mu S - 2\mu(\nabla \cdot u)I/3$, onde $S = 0,5[\nabla u + (\nabla u)^T]$ e $I = \delta_{ij}$, ρ é a densidade, f_b são forças por unidade de massa que incluem a gravidade e tensão superficial. A fração de fase pode assumir valores dentro do intervalo $0 \leq \gamma \leq 1$, com os valores zero e um correspondendo a regiões que acomodam uma única fase, por exemplo, $\gamma = 0$ para o gás e $\gamma = 1$ para o líquido. Por conseguinte, a fração de gradientes de fase é encontrada somente na região da interface.

Dois fluidos imiscíveis são considerados como um fluido efetivo em todo o domínio, as propriedades físicas são calculadas como médias ponderadas com base na distribuição da fração de volume de líquido, sendo, portanto igual para as propriedades de cada fluido nas suas correspondentes regiões ocupadas e variando apenas através da interface,

$$\rho = \rho_l \gamma + \rho_g (1 - \gamma) \quad (4.25)$$

$$\mu = \mu_l \gamma + \mu_g (1 - \gamma) \quad (4.26)$$

onde ρ_l, μ_l e ρ_g, μ_g são as densidades e viscosidades da fase líquida e gás, respectivamente.

No presente estudo, uma abordagem modificada semelhante à proposta por UBBINK, 1997 foi utilizada, com um modelo avançado formulado por OPENCFD, 2013, confiando a formulação convencional de dois fluidos do método VOF no âmbito do método dos volumes finitos. Neste modelo um termo convectivo adicional é introduzido na equação de transporte para a fração de fase. Este termo é proveniente da modelagem da velocidade em termos de médias ponderadas das correspondentes velocidades de líquido e de gás, proporcionando uma resolução mais nítida da interface. O modelo faz uso da formulação dois-fluidos Euleriana para escoamento bifásico, onde as equações de fração de fase são resolvidas separadamente para cada fase. Portanto, as equações para cada uma das frações de fase pode ser expressa como:

$$\frac{\partial \gamma}{\partial t} + \nabla \cdot (u_l \gamma) = 0 \quad (4.27)$$

$$\frac{\partial (1-\gamma)}{\partial t} + \nabla \cdot [u_g (1-\gamma)] = 0 \quad (4.28)$$

onde os subscritos l e g denotam a fase líquida e gasosa, respectivamente. Partindo do princípio de que as contribuições das velocidades de líquido e gás para a evolução da superfície livre são proporcionais à fração de fase correspondente, e definindo a velocidade efetiva do fluido no modelo VOF como uma média ponderada tem-se:

$$u = \gamma u_l + (1-\gamma) u_g. \quad (4.29)$$

A Equação (4.27) pode ser reescrita e utilizada como uma equação de evolução de fase γ :

$$\frac{\partial \gamma}{\partial t} + \nabla \cdot (u \gamma) + \nabla \cdot [u_r \gamma (1-\gamma)] = 0 \quad (4.30)$$

onde $u_r = u_l - u_g$ é o vetor da velocidade relativa, designada como a velocidade de compressão. Consequentemente, a equação que rege a fração de volume, Eq. (4.30), contém um termo convectivo adicional, referido como termo de compressão, tendo em mente o seu papel para “comprimir” no sentido de deixar a superfície livre mais nítida. É importante notar que a formulação de compressão representa apenas uma denotação e não se refere a escoamento compressível. Em comparação com a Eq. (4.23), este termo aparece como uma contribuição artificial à convecção da fração de fase, desde que a derivação da Eq. (4.30) seja

baseada na velocidade definida pela Eq. (4.29), obtendo-se um acoplamento entre o VOF clássico e o modelo de dois fluidos. O termo convectivo adicional contribui significativamente para uma resolução da interface superior, evitando assim a necessidade de instituir um esquema especial para a convecção, como o CICSAM (*Compressive Interface Capturing Scheme for Arbitrary Meshes*) (BERBEROVIC et al., 2009; BERBEROVIĆ et al., 2009; UBBINK; ISSA, 1999). Este termo é ativo somente na região de interface e desaparece em ambos os limites da fração de fase, consequentemente, não afetando a solução fora dessa região. Assim, além de refletir adequadamente a física do escoamento, a principal vantagem de tal formulação é na possibilidade de capturar a região da interface muito mais acentuadamente em comparação com a abordagem clássica (DAMIÁN, 2013).

A equação de quantidade de movimento (Eq. (4.24)) é modificada com o intuito de contabilizar os efeitos da tensão superficial. A tensão superficial é o resultado das forças atrativas entre as moléculas no fluido. Ela pode ser exemplificada através de uma bolha de gás imersa em um líquido, onde no seu interior a força resultante é zero, mas na sua superfície a resultante da força radial cria um efeito combinado das componentes radiais da força ao longo de toda a superfície moldando uma forma esférica à bolha. A tensão superficial pode ser vista como a força atuante na superfície requerida para manter um equilíbrio em um dado instante. Nas interfaces onde dois fluidos estão separados, mas um deles não está na forma de esferas, a tensão superficial atua para minimizar a energia livre, diminuindo a área da interface.

O modelo de Força Superficial Contínua (CSF - *Continuum Surface Force*) é aplicado no modelo como proposto por BRACKBILL; KOTHE; ZEMACH, 1992. Este modelo é implementado como um termo fonte (Eq. (4.31)) adicionado à equação de momento Eq. (4.24),

$$f_{\sigma} = \sigma k_c \nabla \gamma \quad (4.31)$$

onde k_c é a curvatura da superfície, determinada pela expressão:

$$k_c = -\nabla \cdot \left(\frac{\nabla \gamma}{|\nabla \gamma|} \right). \quad (4.32)$$

Considerando ambos os fluidos como sendo newtonianos e incompressíveis, a tensão é diretamente proporcional à taxa de deformação, a qual pode ser decomposta em uma forma mais conveniente para a discretização:

$$\nabla \cdot D = \mu[\nabla u + (\nabla u)^T] = \nabla \cdot (\mu \nabla u) + (\nabla u) \cdot \nabla \mu. \quad (4.33)$$

A fim de conferir estabilidade à solução e para simplificar a definição das condições de contorno (BERBEROVIĆ et al., 2009) o tratamento da pressão é realizado utilizando a pressão modificada p_d , definida na Eq. (4.34):

$$p_d = p - \rho g \cdot x \quad (4.34)$$

onde x é o vetor posição. Após a realização de algumas manipulações na Eq. (4.34), para obtenção do gradiente da pressão como descrito em DAMIÁN, 2013, é possível aplicá-la na equação do momento juntamente com a Eq. (4.33) obtendo-se:

$$\frac{\rho u}{\partial t} + \nabla \cdot (\rho u u) - \nabla \cdot (\mu \nabla u) - (\nabla u) \cdot \nabla \gamma = -\nabla p_d - g x \nabla \rho + \sigma k_c \nabla \gamma \quad (4.35)$$

Em suma, o presente modelo matemático é dado pela equação da continuidade, Eq. (4.22), equação fração de fase, Eq. (4.30), e a equação de momento, Eq. (4.35).

O cálculo preciso da distribuição da fração de fase é crucial para uma avaliação adequada da curvatura da superfície. O solver MULES (*Multidimensional Universal Limiter for Explicit Solution*) é utilizado no OpenFOAM para garantir uma correta solução do termo de compressão e garantir que a solução pertença ao intervalo (0, 1).

A discretização das equações governantes do escoamento é baseada no método de volumes finitos, formulado com arranjo co-localizado de variáveis (VERSTEEG; MALALASEKERA, 2007; VERSTEEG; MALALASEKERA, 1995), onde pressão e velocidade são resolvidas com métodos segregados. Para o acoplamento pressão-velocidade são utilizados os algoritmos SIMPLE (*Semi-Implicit Method for Linked Pressure*) ou PISO (*Pressure Implicit Splitting of Operators*) (ISSA, 1986). O PISO é uma escolha adequada para escoamentos transientes onde soluções intermediárias mais precisas são requeridas. O SIMPLE é mais adequado para escoamento estacionário. Deste modo, o algoritmo PISO é utilizado como método de acoplamento pressão-velocidade no problema. O OpenFOAM também possibilita a utilização de um algoritmo que combina a estrutura dos algoritmos PISO e SIMPLE denominado PIMPLE (OPENCDF, 2013). O algoritmo PIMPLE pode ser executado em solvers com o termo $\partial/\partial t$ nas equações, casos transientes não limitados ao Co , como o PISO e executar casos pseudo-transiente, grandes passos de tempo para alcançar o caso estacionário com o mínimo de *under-relaxation*. O solver CFD *interFoam* possibilita a

utilização do algoritmo PIMPLE. No entanto, o algoritmo PISO foi utilizado no presente trabalho juntamente com o critério de estabilidade de Courant.

4.2.3.2 Modelagem da turbulência

Considerando-se um escoamento de característica anisotrópica, o modelo de turbulência utilizado para o fechamento das equações na abordagem do reator *airlift* foi o $k-\omega SST$. Os grandes vórtices do problema dependem fortemente da geometria do escoamento e de como eles são gerados, por isso possuem características anisotrópicas.

A modelagem do tensor de Reynolds foi considerada segundo a Eq. (4.15), onde k é a energia cinética turbulenta, e ν_t a viscosidade turbulenta, definida com o modelo de transporte de tensão cisalhante (*Shear Stress Transport*), SST $k-\omega$ (MENTER; ESCH, 2001; MENTER; KUNTZ; LANGTRY, 2003). Tal modelo busca conciliar as vantagens do modelo $k-\varepsilon$ com as do modelo $k-\omega$. Para escoamentos em camada-limite, o modelo $k-\omega$ é superior ao $k-\varepsilon$ na solução da subcamada viscosa. No entanto, o modelo $k-\omega$ requer uma condição de contorno não nula para ω para correntes livres não turbulentas, sendo o modelo extremamente sensível a esse valor especificado (MENTER, 1992). Foi demonstrado em CAZALBOU; SPALART; BRADSHAW, 1994 que o modelo $k-\varepsilon$ não apresenta essa deficiência.

Entretanto, o modelo SST combina uma formulação robusta e precisa do modelo $k-\omega$ próxima à parede com a independência de escoamento livre do modelo $k-\varepsilon$, fora da camada limite. Para tal, o modelo $k-\varepsilon$ é reescrito em termos da taxa de dissipação específica, ω . Os modelos $k-\omega$ padrão e o $k-\varepsilon$ transformado são ambos multiplicados por uma função peso e somados. Essa função peso F_1 é zero (levando ao modelo $k-\omega$ padrão) na camada interna da camada limite turbulenta, e unitária (correspondendo ao $k-\varepsilon$ padrão), na camada externa da mesma. O modelo utilizado é similar ao descrito em MENTER; ESCH, 2001:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho u_j k)}{\partial x_j} = \tilde{P}_k - \beta^* \rho k \omega + \frac{\partial}{\partial x_j} \left[\Gamma_k \frac{\partial k}{\partial x_j} \right] \quad (4.36)$$

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial(\rho u_j \omega)}{\partial x_j} = \frac{g}{\nu_t} P_k - \beta \rho \omega^2 + \frac{\partial}{\partial x_j} \left[\Gamma_\omega \frac{\partial \omega}{\partial x_j} \right] + (1 - F_1) 2 \rho \sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}. \quad (4.37)$$

com

$$\Gamma_k = \mu + \frac{\mu_t}{\sigma_k}, \Gamma_\omega = \mu + \frac{\mu_t}{\sigma_\omega}, P_k = \tau_{ij} \frac{\partial u_i}{\partial x_j}, \tau_{ij} = \mu_t \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \right) - \frac{2}{3} \rho k \delta_{ij}, \tilde{P}_k = \min(P_k; c_l \varepsilon). \quad (4.38)$$

O último termo do lado direito da Eq. (4.37) é conhecido como termo de difusão cruzada. MENTER, 1992 demonstrou que introduzindo esse termo difusivo na equação de ω , a dependência de escoamento livre do modelo $k - \omega$ é reduzida. O efeito principal deste termo é o aumento da produção de ω , e consequentemente o aumento da dissipação de k . Na Eq. (4.37), o termo de difusão cruzada é multiplicado pela função peso, ou função de mistura, F_1 , com base na distância até a parede mais próxima, e definida como:

$$F_1 = \tanh(\arg_1^4); \arg_1^4 = \min \left[\max \left(\frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right), \frac{4\rho\sigma_{\omega 2}k}{CD_{k\omega}y^2} \right] \quad (4.39)$$

onde y é a distância para a superfície mais próxima e $CD_{k\omega}$ é a parte positiva do termo de difusão cruzada, dado por:

$$CD_{k\omega} = \max \left(2\rho\sigma_d \frac{1}{\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}; 1.0e^{-10} \right). \quad (4.40)$$

A definição da viscosidade turbulenta segundo a hipótese de Bradshaw assume que o tensor de Reynolds é proporcional à energia cinética turbulenta, sendo dada por:

$$\mu_t = \rho \frac{a_1 k}{\max(a_1 \omega; S \cdot F_2)} \quad (4.41)$$

onde S é o módulo da taxa de cisalhamento média, S_{ij} :

$$S_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \quad (4.42)$$

e F_2 é a função de mistura para a viscosidade turbulenta no modelo SST, definida como:

$$F_2 = \tanh(\arg_2^2); \arg_2 = \max \left(\frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{\omega y^2} \right). \quad (4.43)$$

Definindo ϕ como o conjunto de constantes para o fechamento do modelo SST, e sendo ϕ_1 e ϕ_2 as constantes dos modelos $k-\omega$ e $k-\varepsilon$, respectivamente. As constantes ϕ são calculadas utilizando-se a função de mistura entre as constantes de $\phi_1(k-\omega)$ e $\phi_2(k-\varepsilon)$:

$$\begin{aligned} \sigma_{k1} = 1,176, \sigma_{w1} = 2,000, k = 0,41, \gamma_1 = 0,5532, \beta_1 = 0,0750, \beta^* = 0,09, c_l = 10,0 \\ \sigma_{k2} = 1,000, \sigma_{w2} = 1,168, k = 0,41, \gamma_2 = 0,4403, \beta_2 = 0,0828, \beta^* = 0,09 \end{aligned} \quad (4.44)$$

No OpenFOAM 2.0.1 este método é implementado através do código *kOmegaSST* disponível para utilização nos modelos de turbulência RAS para fluidos incompressíveis e compressíveis.

4.2.3.3 Tratamento da região de parede

Em problemas envolvendo escoamentos em regime turbulentos a interação entre fluido e fronteiras sólidas originam diversos efeitos sobre o transporte das quantidades turbulentas devidos, por exemplo, ao baixo número de Reynolds e à alta tensão de cisalhamento. Uma vez que as paredes são fontes importantes de vorticidade e turbulência, o tipo de abordagem adotado na modelagem da região de parede pode influenciar significativamente a fidelidade das soluções obtidas numericamente. Portanto, uma representação apropriada desta região pode determinar o sucesso da resolução dos escoamentos com presença de paredes.

No estudo de escoamentos turbulentos o escoamento é dividido em duas grandes regiões, a camada externa, na qual os efeitos turbulentos dominam sobre os efeitos viscosos, e a camada interna, caracterizada pelo movimento aleatório dentro da camada limite turbulenta. A camada limite turbulenta pode ser dividida em três sub-regiões ou subcamadas, como apresentado na Figura 4.16. Na subcamada mais interna temos a subcamada viscosa. A subcamada viscosa se estende da parede até aproximadamente a região chamada de 5 unidades da parede, ou seja, $y^+ \approx 5$. É justamente nessa subcamada viscosa que os maiores gradientes de velocidade ocorrem e a viscosidade molecular possui um papel predominante na transferência de movimento, calor e massa.

Figura 4.16: Perfil de velocidade junto a uma superfície sólida. Fonte: SOUZA et al., 2011

A complexidade do escoamento próximo à parede faz com que o tamanho da malha computacional influencie na condição de contorno, conforme mostra a Equação (4.45):

$$y^+ = \frac{y u_*}{\nu}; u_* = \sqrt{\tau_0 / \rho} \quad (4.45)$$

onde y é a distância da parede, ν e ρ são respectivamente a viscosidade cinemática e a densidade do fluido, u_* é a velocidade de atrito e τ_0 é a tensão de cisalhamento na superfície de acordo com a Lei da viscosidade de Newton.

Em problemas turbulentos a interação entre fluido e parede tem uma importância significativa devido ao fato de existir grandes gradientes de velocidade que geram intensa turbulência. Fora dessa região os efeitos convectivos são muito mais importantes que as tensões viscosas.

Na subcamada mais externa ($y^+ > 30$), chamada de logarítmica, os efeitos de inércia dominam o escoamento turbulento. É importante destacar que entre a subcamada viscosa e a subcamada turbulenta ($5 < y^+ < 30$) existe uma região onde tanto o efeito viscoso como o turbulento são importantes (região de ajuste dos dois perfis).

Quando o domínio computacional é muito grande, o que normalmente ocorre em problemas de engenharia quando simulados em tamanho real, a malha próxima à parede se torna pouco refinada e a condição de não deslizamento é adotada. Ao adotar-se a condição nula de deslizamento, a componente normal da velocidade é zero. Isso facilita, por exemplo, a definição das condições de contorno em problemas viscosos. No entanto, para que a condição

de deslizamento seja aplicada, a distância do primeiro nó até a parede deve ser no máximo $y^+ \approx 1$, trazendo grandes restrições ao tamanho da malha devido ao elevado custo computacional associado. Assim, essa condição, quando aplicada exige a utilização de um modelo de turbulência de baixo Reynolds, pois os modelos implementados para escoamentos turbulentos completamente desenvolvidos não são aplicáveis a regiões próxima a parede.

Como alternativa ao refinamento excessivo na região de parede existem as chamadas funções de parede. As funções de parede são uma coleção de relações semi-empíricas as quais possibilitam a aplicação de condições de contorno a certa distância desta, de tal forma que as equações do modelo de turbulência não sejam resolvidas na região próxima à parede (Figura 4.17). Assim, é possível determinar os valores médios da velocidade e temperatura ou outros escalares na região próxima a parede.

Figura 4.17: Turbulência próxima a parede: A) Malha refinada próximo a parede ($y^+ \approx 1$). B) Malha para funções de parede.

De modo geral, o uso das funções de parede é uma boa abordagem se o interesse de estudo está concentrado na região de mistura da camada externa e se o custo computacional é um grande limitante. No entanto, se a previsão das forças ou da transferência de calor na parede são importantes para a simulação (arrasto aerodinâmico, desempenho das lâminas em turbomáquinas) não se recomenda o uso de funções de parede. Se o primeiro nó normal à parede está localizado na região da subcamada viscosa, não devem ser utilizadas as funções de parede, ou seja, se $y^+ < 30$. O modelo de turbulência recomendado para a maioria dos casos é o $k - \omega$ SST (CFD ONLINE 2015; GUERRERO, 2014).

Para o estudo em regiões próximas à parede são necessárias condições de contorno apropriadas. O OpenFOAM possui as funções disponíveis para fluidos compressíveis e incompressíveis (RANS/LES) nos respectivos diretórios chamados *wallFunctions*.

Considerando o problema de otimização do reator *airlift*, o custo computacional do problema torna-se um grande limitante, pois é necessária a avaliação de centenas de geometrias. Sendo assim, o modelo de turbulência $k-\omega$ SST e funções de parede foram utilizados para a solução do problema proposto.

4.2.3.4 Solver desenvolvido no OpenFOAM para a simulação computacional do reator *airlift*

A metodologia numérica empregada no presente trabalho é baseada no método de volumes finitos (FVM), utilizando-se como ferramenta de discretização e análise numérica do pacote OpenFOAM. Para o reator *airlift*, o método VOF (*Volume of fluid*) é utilizado para análise do escoamento água-ar. O solver base do software OpenFOAM utilizado foi o *interFoam* (solver para dois fluidos incompressíveis, isotérmicos, imiscíveis utilizando o método VOF). O *interFoam* utiliza modelos constitutivos para fluidos newtoniano e newtoniano generalizado. Para o acoplamento velocidade-pressão utilizou-se o laço PISO.

A fim de assegurar a estabilidade do processo de solução, os cálculos foram realizados usando-se um passo de tempo auto-adaptativo, que é ajustado no início do ciclo de tempo de iteração com base no número Courant, como definido na Eq. (4.21). De acordo com este conceito, como descrito na Seção 4.1.2.4, o novo passo de tempo irá diminuir se o número de Courant local calculado ultrapassar o número de Courant máximo permitido.

Tendo o solver base *interFoam* como ponto de partida e acoplando-se uma equação de transporte foi obtido o solver *interDyScalarFoam*. A equação de transporte para o fluido passivo é resolvida a partir do campo de velocidade previamente calculada no laço PISO.

O procedimento realizado para adicionar o cálculo da equação de transporte no solver base é o mesmo realizado para o caso anterior, descrito na Seção 4.1.2.4. No APÊNDICE C é apresentado o código fonte .C do solver desenvolvido.

4.2.3.5 Malha computacional - reator *airlift*

Para a geração da geometria e da malha computacional do reator *airlift* foi utilizado o programa *blockMesh*. O *blockMesh* é uma utilidade do OpenFOAM para a elaboração de

geometrias e geração de malhas ortogonais com células hexaédricas. A malha é gerada a partir de um arquivo do dicionário chamada *blockMeshDict*. O *blockMesh* lê este dicionário, gera a malha e escreve os dados da malha em arquivos para os pontos, faces, células, contornos e etc. no mesmo diretório. A utilização desta função nativa do OpenFOAM possibilita o uso direto da malha gerada, sem a necessidade de convertê-la como no caso de um gerador de malha externo (Seção 4.1.2.5).

A geração de malha computacional aplicada para o problema de otimização torna cada vez mais evidente o antagonismo entre qualidade da solução e custo computacional. A necessidade de avaliar centenas de geometrias com um custo computacional elevado implicaria na inviabilidade da aplicação de um processo de otimização. Para contrabalancear estas variáveis é necessário gerar uma malha que atenda as necessidades exigidas pelo método VOF e o refinamento próximo à parede. O refinamento próximo à parede é aliviado com a utilização das funções de parede, como descrito na Seção 4.2.3.3, que não exige um $y^+ \approx 1$.

O refinamento próximo à parede pode ser alcançado com o auxílio de uma função do OpenFOAM chamada *refineWallLayer*. Com o uso desta função é possível refinar as células próximas à parede e assim atender ao critério exigido para a aplicação de métodos de turbulência para baixo Reynolds sem o auxílio de funções de parede. A utilização desta função foi testada no processo de acoplamento do software PyCFD-O e pode ser habilitada para testes futuros.

Outro fator importante na simulação do reator *airlift* é a modelagem dos injetores. Para a simulação em questão, os injetores foram considerados como faces retangulares de 0,8 mm e 0,7 mm. A utilização de injetores como faces retangulares facilita a geração de malha nesta região, sem a exigência de um alto grau de refinamento. A geometria adotada para os injetores com estas dimensões ocasiona um aumento de 10% na área de injeção se comparado com os injetores reais de 0,8 mm de diâmetro. A utilização de injetores como faces retangulares impede a geração de zonas com alto grau de refinamento, o que resulta em uma maior uniformização do tamanho dos volumes nesta região da malha e uma maior estabilidade de convergência dos métodos numéricos.

A Figura 4.18 apresenta um exemplo de malha hexaédrica utilizada nas simulações computacionais do reator *airlift*. As especificações e a validação dos resultados numéricos obtidos com a malha computacional serão analisadas na Seção 5.2.1.

Figura 4.18: Exemplo de malha computacional e injetores utilizados reator *airlift*.

4.3 Software PyCFD-O

4.3.1 Estrutura do programa

Como descrito na seção 2.4, a fim de realizar a otimização automatizada utilizando um software CFD é necessário o desenvolvimento de um *script*. Este *script* tem a função de realizar a junção entre o método de otimização e o software CFD e automatizar o processo de pré-processamento, chamada do *solver* e pós-processamento realizado pelo CFD. Nesta seção será descrita a metodologia, os softwares e os códigos utilizados para realização desta tarefa.

O PyCFD-O é um conjunto de códigos orientado a objetos em python e C++ para sistemas Unix / Linux desenvolvido para acoplar ferramentas de engenharia/desenho assistida por computador (CAD/CAE), otimização multiobjetivo e dinâmica dos fluidos computacional (CFD). O software foi totalmente desenvolvido no Núcleo de Engenharia Térmica e Fluidos – NETeF/USP, como parte da presente pesquisa. O PyCFD-O pode ser resumido em seis etapas, como mostrado na Figura 4.19:

1. Parâmetros geométricos - CAD: Nesta etapa as variáveis de projeto iniciais são configuradas ou os valores obtidos do método de otimização são atualizados para geração da geometria (preparação CAD). Nesta etapa é possível utilizar um software CAD (SALOME¹⁵) (OPEN-CASCADE, 2012) para a criação

¹⁵ SALOME é um software de código aberto desenvolvido pela OpenCASCADE que fornece uma plataforma genérica para pré- e pós-processamento para simulações numéricas.

inicial da geometria base. Após a execução inicial temos acesso ao código fonte que gera a geometria e a partir daí são criadas rotinas que possibilitam a automatização do processo de criação das geometrias.

2. Geração da malha: Com a geometria definida a geração da malha é iniciada. O PyCFD-O permite a utilização de softwares de geração nativos do OpenFOAM (*blockMesh*, *snappyHexMesh*) (OPENCFD, 2013) e o uso de geradores externos como os utilizados no SALOME. A qualidade da malha gerada é verificada ao final de todo o processo. A malha em questão deve respeitar certas qualidades previamente definidas no código, como por exemplo, não ortogonalidade, máximo *skewness*, entre outras. Se a malha gerada não obtiver sucesso uma nova malha é gerada com os parâmetros dos métodos de geração de malha refinados. Esse processo é realizado com o auxílio da função *checkMesh* do OpenFOAM e um conjunto com códigos desenvolvidos para o PyCFD-O.
3. Pré-processamento: O estágio de pré-processamento é responsável pela configuração do solver CFD com as especificações do fluido, condições de contorno e métodos e modelos utilizados. Nesta etapa também é realizada a configuração e preparação para a execução em paralelo do solver CFD. Os métodos de decomposição de malha são os disponibilizados pelo OpenFOAM e a execução em paralelo utiliza roteiros em bash para o PBS (*Portable Batch System*).
4. Execução de solver CFD: Execução do OpenFOAM para simulação computacional do escoamento previamente configurado.
5. Pós-processamento: Após a execução do solver CFD a correta execução e convergência da simulação são verificadas. Se a execução ocorreu perfeitamente o software executa a extração dos dados e calcula os parâmetros de otimização que serão utilizados no método de otimização. No entanto, se ocorrer algum problema na execução do solver, a mesma geometria é executada novamente, com algum parâmetro de geração de malha refinado. O processo se repetirá até a execução retornar sucesso.
6. Método de otimização: Os parâmetros de otimização são exportados e o método de otimização completa a execução e retorna, se necessário, uma nova população com os parâmetros geométricos para as novas geometrias que serão avaliadas. No PyCFD-O foram implementados até o momento dois métodos de

otimização estocástica: NSGA-II (DEB, 2001; DEB et al., 2002) e um algoritmo genérico de otimização multiobjetivo evolutiva. No entanto, o software permite a implementação e integração com outros métodos de otimização. Outra característica do software é a possibilidade da avaliação de cada indivíduo da população em paralelo. O PyCFD-O trabalha com processamento paralelo em clusters com o PBS para realizar o agendamento de tarefas. Também é possível avaliar várias geometrias em paralelo em computadores comuns (*multicores*) sem a utilização do PBS. O software salva um arquivo de saída com todos os parâmetros avaliados no processo de otimização, possibilitando a criação de um banco de dados para futuras verificações e análise.

Figura 4.19: Metodologia de otimização automática: PyCFD-O.

4.3.2 Métodos de otimização multiobjetivo implementados

Como descrito na Seção 2.3 o algoritmo NSGA-II apresenta-se como um dos métodos de otimização multiobjetivo mais eficientes na atualidade. O NSGA-II difere de um algoritmo genético simples apenas na forma como é executada a seleção, uma vez que os operadores genéticos de recombinação e mutação permanecem os mesmos. Assim, a vantagem do NSGA-II é sua capacidade de a cada geração, utilizar uma trajetória diferenciada de busca, devido ao fato de que novos indivíduos sempre são obtidos por meio de combinações das populações anteriores. Esta estratégia diferenciada de busca permite explorar várias regiões do espaço de soluções. Deste modo, o algoritmo do NSGA-II foi totalmente implementado em python e disponibilizado para utilização no software PyCFD-O.

A fim de verificar o correto funcionamento do algoritmo implementado NSGA-II, utilizou-se outro algoritmo denominado EMOO (*Evolutionary Multi-Objective Optimization*) (BAHL et al., 2012). O algoritmo EMOO também é um modelo elitista baseado no próprio NSGA-II e é disponibilizado para download em www.g-node.org/emoo. O uso de ferramentas de otimização já desenvolvidas permitiu a verificação dos métodos implementados diretamente no PyCFD-O.

Deste modo, foi verificado primeiramente o correto funcionamento de ambos os códigos em um problema de otimização multiobjetivo teste como descrito no APÊNDICE A. Em seguida, os métodos foram acoplados no processo de otimização com CFD no software PyCFD-O.

4.3.3 Paralelização

Como mencionado anteriormente, um fator importante relacionado aos algoritmos genéticos (AG) é o alto esforço computacional necessário para realizar as avaliações dos objetivos associados com a população. A avaliação de um grande número de indivíduos durante muitas gerações pode conduzir a tempos de processamento impraticáveis em casos práticos de engenharia. Por outro lado, devido à sua característica intrínseca, os AGs são facilmente e eficientemente paralelizáveis. Em cada geração, todos os N indivíduos da população podem ser avaliados de forma independente em diferentes processadores, uma vez que o algoritmo principal requer somente os valores dos objetivos para iterar.

Aproveitando esta característica inerente dos algoritmos genéticos e o alto custo computacional associado com as avaliações CFD, um paradigma mestre-escravo

(GOLDBERG, 1989) de paralelização foi implementado no software PyCFD-O. O padrão de comunicação de dados MPI (*Message Passing Interface*) foi utilizado.

A Figura 4.20 apresenta um diagrama esquemático da utilização do paradigma mestre-escravo. Como pode ser observado na figura, o paradigma mestre-escravo envolve dois conjuntos de processadores: o mestre e o escravo. O mestre é responsável pelo pré- e pós-processamento da ordem de serviço; o escravo é responsável pela execução/processamento das ordens de serviço (execução do solver CFD). A Figura 4.21 ilustra a aplicação do paradigma mestre-escravo no processo de otimização multiobjetivo no PyCFD-O. Inicialmente, com a população inicial gerada verificam-se os indivíduos/geometrias dessa população serão executados em paralelo. Se esta opção estiver habilitada, o processamento de cada geometria é realizado em paralelo na estação de trabalho, de acordo com as especificações previamente configuradas. A alocação dos processos entre os recursos computacionais disponíveis são realizados através da utilização do programa PBS Professional (*Portable Batch System*) disponível na estação de trabalho. Ao executar o processamento em paralelo utiliza-se uma barreira de sincronização para garantir que todos os indivíduos da população do ciclo de otimização foram processados.

Usualmente no PyCFD-O, cada indivíduo/geometria da população é resolvido em um único nó de processamento da estação de trabalho. E este indivíduo/geometria é resolvido em paralelo dentro desse nó de processamento de acordo com os recursos computacionais disponíveis. Ou seja, o domínio da malha computacional da geometria é decomposto e o solver CFD é resolvido em paralelo. Esta divisão de recursos onde cada geometria é processada em paralelo em um nó garante um baixo nível de comunicação entre os nós.

Figura 4.20: Representação esquemática do paradigma mestre-escravo utilizado no PyCFD-O.

Figura 4.21: Processamento paralelo do método de otimização no PyCFD-O.

Para o processamento das simulações, a estação de trabalho dedicada Monalisa do Núcleo de Engenharia Térmica e Fluidos foi utilizada. A estação de trabalho Monalisa possui o sistema operacional SUSE Linux Enterprise Server 11, juntamente com os softwares e respectivas licenças (se aplicável) para: PBS Pro 12.0, ANSYS CFX/Fluent, modeFrontier, OpenFOAM 2.0.1 e Salome Meca 2011.2. As especificações de hardware da estação de trabalho são apresentadas na Tabela 4.3:

Tabela 4.3: Especificações da estação de trabalho Monalisa.

SGI Altix XE 1300: 10 nós de processamento	
6 compute nodes (3 servers Altix XE 340) Intel Xeon X5650, 2.4GHz, 12M cache, RAM 24GB	
4 compute nodes (2 servers SGI C2112) Intel Xeon E5-2660, 2.2GHz, 20M cache RAM 32GB	
Head node: Intel Xeon E5620, 2.4GHz, 12M cache	
Total de cores de processamento: 136	
HD total: 15TB	

4.4 Acoplamento CFD – Otimização software PyCFD-O

Nesta seção será descrita a metodologia utilizada para o cálculo dos parâmetros de otimização aplicada nos reatores e a definição do problema de otimização multiobjetivo. Feito isso, o acoplamento com o solver CFD é realizado através do software PyCFD-O.

4.4.1 Processo de otimização do projeto

Como já discutido, os parâmetros do escoamento têm influência direta no processo químico envolvido em biorreatores e fotobiorreatores. Assim a otimização destes parâmetros torna-se extremamente importante para o maior rendimento desses processos. Neste trabalho esta otimização é feita através da mudança dos parâmetros geométricos ideais para a estrutura do reator, de forma a encontrar um valor ótimo para os parâmetros do escoamento que podem influenciar o processo de conversão de biomassa.

Um dos parâmetros do escoamento que possui influência direta sobre a geometria do reator é a distribuição do tempo de residência (DTR). O DTR representa o tempo em que cada porção do fluido permanece no interior do reator. Sua análise é de extrema importância, pois se porções do fluido permanecem tempos muito discrepantes no interior do reator as reações químicas desejadas podem não ser obtidas (CHOI et al., 2004). O segundo parâmetro a ser estudado é a tensão de cisalhamento. Este parâmetro está relacionado com a garantia de sobrevivência dos micro-organismos envolvidos nos biorreatores e FBR. Tensões muito altas podem ser prejudiciais, causando a morte desses organismos. No entanto, este parâmetro também é necessário para a ocorrência da reação (MICHELS et al., 2010).

Nesta seção será apresentada a metodologia utilizada para o cálculo dos parâmetros de otimização (DTR e tensão de cisalhamento) aplicados nos reatores contínuo e *airlift*.

4.4.2 Cálculo do DTR

Os elementos de fluido podem adotar diferentes caminhos ao longo do reator percorrendo em diferentes tempos todo o recipiente (LEVENSPIEL, 2000). A idade de um elemento de fluido é o tempo que passa entre a entrada do elemento no equipamento e um instante dado sendo, portanto, menor ou igual ao tempo de residência. O tempo de residência de uma porção ou elemento de fluido é o tempo decorrido desde a sua entrada até a sua saída do equipamento. Uma vez que diversos elementos de um fluido escoam por diferentes trajetórias dentro do reservatório, desde a sua entrada até a saída, diferentes tempos de

residência podem ocorrer (LEVENSPIEL, 2000). A geometria e o tipo de escoamento têm influência direta sobre o tempo de residência das partículas fluidas. Alguns fenômenos de escoamento que podem ocorrer no interior dos equipamentos são:

1. Escoamento preferencial: ocorre quando grandes quantidades de elementos do fluido escoam através do equipamento em tempo menor que outros;
2. Curto circuito: ocorre quando elementos do fluido escoam através do equipamento sem percorrer todo seu interior. Este fenômeno indica deficiência de projeto e não é desejável, uma vez que diminui a eficiência do dispositivo. Os tempos de residência médios (t_m) são inferiores ao tempo de residência teórico (TRT);
3. Zonas mortas: são regiões do equipamento isoladas ou inacessíveis, onde o fluido nelas aprisionadas não interage com as regiões ativas. Ou seja, não há troca de matéria entre as regiões mortas e a fração ativa do equipamento. Tempos de residência médios relativamente altos, quando comparados ao TRT;
4. Zonas estagnantes: são regiões do equipamento onde o fluido aprisionado interage fracamente com a região ativa. A troca de matéria entre estas regiões ocorre lentamente;
5. Reciclo: ocorre quando parcelas do fluido são recirculadas para a entrada ou interior do equipamento;
6. Retromistura: ocorre quando parcelas do fluido apresentam movimento na direção contrária ao fluxo principal.

Algumas dessas anomalias do escoamento podem ser detectadas a partir da observação das curvas de distribuição de tempos de residência (DTR) e da análise de parâmetros calculados a partir de dados experimentais. Vários trabalhos podem ser encontrados na literatura, onde são feitas essas análises para biorreatores e fotobiorreatores, entre eles os trabalhos de: CHOI et al., 2004; GHASEMI et al., 2010; PATWARDHAN et al., 2005.

O cálculo da DTR é efetuado a partir da análise da função $E(t)$, que representa a distribuição de idades do fluido na saída do equipamento. $E(t).dt$ é a fração de elementos do fluido que deixa o dispositivo entre o intervalo $t + dt$. Para tempos suficientemente elevados todos os elementos do fluido que entraram no sistema em $t = 0$ terão deixado o sistema, então:

$$\int_0^{\infty} E(t) dt = 1. \quad (4.46)$$

Ou seja, a soma de todas as frações de material na saída do reservatório é igual a unidade. A função $E(t)$ é assim definida:

$$E(t) = \frac{C(t)}{\int_0^{\infty} C(t) dt}, \quad (4.47)$$

onde $C(t)$ é uma função que descreve a curva resultante da concentração da partícula fluida pelo tempo.

O tempo médio teórico de residência de um fluido em um reservatório é definido como:

$$TRT = \frac{V}{Q}, \quad (4.48)$$

onde V é o volume do reservatório e Q é a vazão volumétrica (onde as vazões de alimentação e de saída são iguais, ou seja, o volume de controle do reservatório é constante).

Com isso podemos representar o tempo em relação ao tempo médio de residência do fluido no reservatório, de forma adimensional:

$$\bar{t} = \frac{t}{TRT}. \quad (4.49)$$

Assim, normalizando-se a curva $E(t)$ em relação à variável $\bar{t}$, a relação entre $E(t)$ e $E(\bar{t})$ é dada:

$$\int_0^{\infty} E(t) dt = \int_0^{\infty} E(\bar{t}) d\bar{t} = 1. \quad (4.50)$$

Utilizando-se a Eq. (4.49):

$$TRT \cdot E(t) = E(\bar{t}). \quad (4.51)$$

A média da distribuição de tempos de residência constitui parâmetros associado à DTR conforme definido abaixo:

$$\bar{t} = \int_0^{\infty} t.E(t)dt. \quad (4.52)$$

A dispersão da distribuição é medida pela variância, e é dada por:

$$\sigma^2 = \int_0^{\infty} (t - \bar{t})^2 E(t)dt. \quad (4.53)$$

A magnitude deste momento é uma indicação de quão “ampla” é a distribuição; quanto maior é o valor deste momento, mais ampla é a distribuição (CHISTI, 1989).

O conhecimento da distribuição do tempo de residência de elementos de fluido é utilizado para caracterizar o grau de não idealidade do escoamento. Conforme salientado, esta informação pode ser facilmente determinada pela técnica de estímulo e resposta. Segundo LEVENSPIEL, 2000, a técnica de estímulo e resposta consiste em verificar a resposta do sistema a um estímulo qualquer. Qualquer sinal de entrada de traçador pode ser usado como estímulo, destacando-se pulso, degrau, casual ou periódico. Neste trabalho, considerou-se a técnica tipo pulso, por ser de simples aplicabilidade e atender de forma plenamente satisfatória aos objetivos delineados.

4.4.3 Cálculo da tensão de cisalhamento

A viscosidade de um fluido pode ser entendida como sendo a “aderência” interna de um fluido. Ela é responsável pelas perdas de energia associadas ao transporte de fluidos em dutos, canais e tubulações, e tem um papel primário na geração de turbulência.

Para uma determinada tensão de cisalhamento, um fluido altamente viscoso deforma-se numa taxa menor do que um fluido com baixa viscosidade, assim a viscosidade é a propriedade que determina o grau de sua resistência à força cisalhante. Fluidos nos quais a tensão cisalhante é diretamente proporcional à taxa de deformação são chamados fluidos Newtonianos. Assim:

$$\tau_{xy} \propto \frac{du}{dy}. \quad (4.54)$$

A constante de proporcionalidade na Eq. (4.54) é a viscosidade absoluta (ou dinâmica), denotada por μ . Deste modo, a tensão de cisalhamento viscosa e a viscosidade dinâmica podem ser relacionadas pela equação seguinte:

$$\tau_{ij}^v = \mu \frac{du_i}{dx_j}. \quad (4.55)$$

À velocidades mais elevadas o escoamento torna-se turbulento, ocorrendo a formação de turbilhões. Em situações práticas, a maioria dos escoamentos em tubos são turbulentos. Em um escoamento turbulento totalmente desenvolvido, as três componentes da velocidade são diferentes de zero, podendo ser escritas em termos de uma quantidade média e uma parte flutuante no tempo como a seguir:

$$u_x = \overline{u_x} + u'_x, u_y = \overline{u_y} + u'_y \text{ e } u_z = \overline{u_z} + u'_z. \quad (4.56)$$

Aplicando os termos da Eq. (4.56) na equação de quantidade de movimento, como descrito na Seção 4.1.2.2 Eq. (4.14) obtem-se o tensor de tensões de Reynolds ou tensão turbulenta. Considerando-se a hipótese de Boussinesq em um escoamento unidirecional ao longo de uma placa plana e infinita, a contribuição da turbulência no processo de transferência de quantidade de movimento poderia ser modelada como:

$$\tau_{ij}^t = -\rho \overline{u'_i u'_j} = \mu_t \frac{\partial \overline{u_i}}{\partial x_j}. \quad (4.57)$$

Segundo a proposta de Boussinesq as tensões de Reynolds são proporcionais à deformação do escoamento médio, agindo de maneira análoga às tensões viscosas. Esses termos estão associados ao aumento da resistência à deformação, ou seja, a turbulência manifesta-se num escoamento como um aumento aparente de sua viscosidade. Em contraste à viscosidade molecular, a viscosidade turbulenta não é uma propriedade física do fluido, mas sim uma medida local do nível de turbulência. Um valor que varia de ponto para ponto e de escoamento para escoamento.

A viscosidade turbulenta é calculada de acordo com o modelo de turbulência previamente adotado na simulação computacional do escoamento. O modelo de turbulência $k - \varepsilon$ foi o utilizado para o reator contínuo, como descrito na Seção 4.1.2.2. Para o reator *airlift* o modelo assumido foi o $k - \omega SST$, apresentado na Seção 4.2.3.2.

A tensão cisalhante total (TCT) em uma localização particular do escoamento seria dada pela contribuição da tensão de cisalhamento viscosa τ_v e da tensão de cisalhamento turbulenta τ_t , ou seja:

$$\tau_{Total} = \tau_v + \tau_t. \quad (4.58)$$

Em escoamentos turbulentos, a tensão de Reynolds é muito maior que a tensão viscosa, exceto para uma pequena região a subcamada laminar.

Uma vez que a tensão de cisalhamento viscosa e turbulenta são calculadas através das Eqs. (4.55) e (4.57) em cada nó da malha computacional, torna-se necessário converter estes valores nodais da tensão de cisalhamento em uma variável elementar para ser aplicada na função objetivo a ser otimizada. CHEN et al., 2011 utilizaram em seu trabalho uma média dos valores da tensão de cisalhamento em cada nó, como mostrado a seguir:

$$\tau^e = \frac{1}{ND} \sum_{i=1}^{ND} \tau^i, \quad (4.59)$$

onde τ^e denota o valor elementar médio da tensão de cisalhamento, τ^i o valor nodal da tensão de cisalhamento e ND o número total de nós no elemento considerado. Uma vez obtido o valor elementar médio da tensão de cisalhamento torna-se possível fazer uma classificação das melhores geometrias.

O cálculo utilizando uma média aritmética, Equação (4.59), pode mascarar certas características do escoamento. Através deste cálculo pequenas regiões com altos valores de tensão de cisalhamento apresentam grande influência sobre o resultado final considerando-se todos os pontos das geometrias. Visando dar um maior peso para pontos com baixos valores de tensão de cisalhamento mas que a quantidade de pontos na malha seja grande foi calculada uma média ponderada. O cálculo da média ponderada leva em conta a frequência dos pontos em certos intervalos de valores de tensões de cisalhamento. Quanto maior a frequência e maior o valor da tensão cisalhante desses pontos eles tornam-se mais relevantes para o cálculo final da média ponderada.

Esses cálculos estatísticos de frequência e média ponderada foram realizados com o auxílio de ferramentas estatísticas do python, como o *histogram*. O trecho do código python utilizado para o da média ponderada é apresentado na Figura 4.22.

```

136 sst_min = min(sst)
137 sst_max = max(sst)
138
139 # number of bins
140 k = 100
141
142 indx_k = range(1, (k+1))
143
144 #print sst
145 print 'minSST2 = ', sst_min
146 print 'maxSST2 = ', sst_max
147 print 'k = ', k
148 #print indx_k
149
150 binAux = (sst_min + (sst_max - sst_min))/k
151 Bins = multiply(binAux, indx_k)
152
153 freq, hist = histogram(sst, bins=k)
154
155 pesoFreq = freq/float(sum(freq))
156
157 sstPond = indx_k * Bins * pesoFreq
158
159 mediaSST_pond = sum(sstPond)/k
160 mediaSST_arit = sum(sst)/len(sst)
161
162 print 'mediaSST_pond', mediaSST_pond
163 print 'mediaSST_arit', mediaSST_arit
164

```

Figura 4.22: Código python para o cálculo da tensão de cisalhamento ponderada.

Para possibilitar estes cálculos foram desenvolvidas rotinas utilizando classes disponíveis no OpenFOAM e python. Estas rotinas foram agrupadas em um código que é executado na etapa de pós-processamento.

4.4.4 Parâmetros geométricos de otimização e funções objetivos: Reator contínuo

Como já discutido a distribuição do tempo de residência e a tensão de cisalhamento têm forte influência no processo químico. Esses componentes fluidodinâmicos sofrem grande influência dos parâmetros geométricos do biorreator.

Foram escolhidos dois parâmetros geométricos do biorreator conforme mostrado na Figura 4.23 que podem ser alterados a fim de otimizar o processo. O primeiro parâmetro é o ângulo de entrada do fluido, α . O ângulo de entrada pode variar de um caso tangencial ($\alpha = 0^\circ$) até um ângulo perpendicular ($\alpha = 90^\circ$). O outro parâmetro geométrico a ser otimizado é a altura da tubulação de saída, H_{out} [m].

Deste modo o processo de otimização consiste em minimizar a tensão de cisalhamento total ponderada (TCT_{pond}) e a variância da distribuição do tempo de residência de acordo com variações nos dois parâmetros geométricos α e H_{out} . O problema de otimização é formulado da seguinte maneira:

$$\text{Min: } \begin{cases} \text{Variância } DTR (\sigma^2) \\ TCT_{pond} \end{cases}, \text{ onde } \sigma^2 = f_1(\alpha, H_{out}) \text{ e } TCT_{pond} = f_2(\alpha, H_{out}) \quad (4.60)$$

$$\text{Sujeito a: } \begin{cases} 0^\circ \leq \alpha \leq 90^\circ \\ 0 \text{ m} \leq H_{out} \leq 1,3 \text{ m} \end{cases} \quad (4.61)$$

Figura 4.23: Representação esquemática do reator contínuo com as variáveis utilizadas no processo de otimização. A) Caso tangencial. B) Caso perpendicular.

As características aplicadas ao modelo CFD adotadas nas simulações são: escoamento incompressível, isotérmico, transiente, sem efeito da gravidade, sem deslizamento nas paredes (*no-slip*) e o modelo de turbulência $k - \varepsilon$. Como observado na revisão da literatura, a maioria das simulações computacionais de biorreatores utilizam este modelo de turbulência.

No presente estudo, o fluido de teste adotado para as simulações foi água a 30°C (303,15 K). A água foi utilizada, pois segundo informações da indústria a viscosidade do caldo no decantador é muito baixa devido a grande presença de água.

Para a entrada do fluido, foi atribuída uma velocidade axial de 1 m/s (condição de Dirichlet) e na saída a pressão média estática relativa foi definida como 0 Pa.

Definidas as funções de otimização com as restrições das variáveis da geometria e as condições de simulação do modelo CFD é possível realizar o processo de otimização multiobjetivo. No presente trabalho dois métodos heurísticos baseados em algoritmos genéticos foram analisados (EMOO e o NSGA-II). Todo o processo, desde a criação da geometria e geração da malha até o cálculo das funções de otimização foi realizado automaticamente através do software PyCFD-O, como descrito na Seção 4.3.

4.4.5 Parâmetros geométricos de otimização e funções objetivos: Reator *airlift*

Para o reator *airlift* foram consideradas as mesmas funções de otimização utilizadas para o caso do reator contínuo: variância da distribuição do tempo de residência e tensão de cisalhamento total ponderada, TCT_{pond} . No entanto, o cálculo dessas funções exige uma modelagem apropriada para o escoamento bifásico e o modelo de turbulência.

Foram escolhidos dois parâmetros geométricos do reator *airlift* conforme mostrado na Figura 4.24, que podem ser alterados a fim de otimizar o processo. O primeiro parâmetro é a distância da placa em relação à parede na região de subida, dy [m]. O outro parâmetro geométrico consiste na altura da placa, H_z [m].

Deste modo, o processo de otimização consiste em minimizar a tensão de cisalhamento total ponderada (TCT_{pond}) e a variância da distribuição do tempo de residência de acordo com variações nos dois parâmetros geométricos dy e H_z . A função objetivo pode ser escrita de maneira similar à adotada na Equação (4.60). Considerando-se como origem o lado inferior esquerdo do reator, como representado na Figura 4.24, os valores de dy e H_z podem assumir valores máximos e mínimos, como mostrado na Equação (4.62):

$$\text{Sujeito a: } \begin{cases} 0,03 \text{ m} \leq dy \leq 0,12 \text{ m} \\ 0,4 \text{ m} \leq H_z \leq 0,51 \text{ m} \end{cases} \quad (4.62)$$

Figura 4.24: Representação esquemática do reator *airlift* com as variáveis utilizadas no processo de otimização.

As características aplicadas ao modelo CFD adotadas nas simulações são: escoamento incompressível, isotérmico, transiente e modelo de turbulência $k-\omega SST$ com funções de parede. A tensão superficial entre o ar e a água é dada por $\eta = 0,07 N/m$. As propriedades da água foram consideradas a temperatura de $298 K$ ($24,85^\circ C$). As propriedades do ar são: viscosidade cinemática $1,48 \times 10^{-5} m^2/s$ e densidade $1,0 Kg/m^3$.

A vazão volumétrica de ar adotada foi de $5,0 l/min$ ($8,33 \times 10^{-5} m^3/s$) na câmara de distribuição de ar. Para as simulações CFD foi considerado o ar na câmara como incompressível ($Mach < 0,3$). Assim, o volume de ar introduzido na câmara é distribuído igualmente entre os onze injetores de acordo com o princípio de conservação da massa. Portanto, a vazão volumétrica para cada injetor é $7,57 \times 10^{-6} m^3/s$.

Assim, como realizado para o reator contínuo, foi definido o problema de otimização multiobjetivo e as condições para o processamento do solver CFD, foi utilizado o software PyCFD-O, a fim de obter as geometrias otimizadas.

CAPÍTULO 5 RESULTADOS E ANÁLISE

Neste capítulo são apresentados os resultados referentes aos reatores analisados. Primeiramente são descritos os resultados do reator operado em regime contínuo e em seguida o reator *airlift* do tipo placa plana. Os estudos realizados no reator contínuo e *airlift* buscam definir uma base para futuras análises voltadas para a obtenção de fermentadores contínuos otimizados e o desenvolvimento de fotobiorreatores para o tratamento de subprodutos da produção de etanol.

5.1 Reator operado em regime contínuo

Nesta seção são apresentados os resultados dos testes necessários para a correta obtenção dos resultados de otimização do reator contínuo na seguinte ordem: escolha da malha computacional, tempo necessário para estabelecer o regime permanente, avaliação dos métodos de otimização, análise da influência da geometria do reator nos parâmetros hidrodinâmicos e a influência desses parâmetros no processo de aumento de escala. Finalmente, serão apresentados os resultados do processo de otimização juntamente com as superfícies de resposta e a análise dos resultados.

5.1.1 Estudo de convergência de malha

O estudo de convergência de malha é de extrema importância para simulações computacionais. A escolha de uma malha que apresente uma boa convergência com um menor custo computacional é essencial para simulações em projetos de otimização, onde é necessária a análise de várias geometrias. Assim, foram realizados testes preliminares de convergência de malha em diferentes composições. Todas as malhas analisadas para o reator contínuo foram obtidas utilizando o gerador de malhas tetraédricas *open source* NETGEN (SCHOBERL, 1997). As informações a respeito das malhas estão descritas na Tabela 5.1.

A análise de convergência para as diferentes malhas foi realizada considerando uma geometria fixa ($H_{out} = 1.0$ m e $\alpha = 0^\circ$) por questões de simplicidade. As análises foram feitas

considerando o perfil de velocidade (magnitude U) ao longo de uma linha transversal, variando ao longo do eixo y e com tempo de simulação do escoamento de 500 segundos. Observa-se, na Figura 5.1, uma diferença significativa entre a malha 1 (Grossa) e a malha 5 (Fina).

Tabela 5.1: Características da malha computacional - Reator contínuo.

Malha	Número de nós	Volumes Tetraédricos	Volume total da geometria [m ³]
1 (Grossa)	18496	94416	1,17198
2	38044	198395	1,17222
3	53737	288221	1,17227
4	69108	374063	1,17229
5 (Fina)	109188	608236	1,17233

Figura 5.1: Perfis de velocidade U (500s) em um corte transversal na geometria de controle para as diferentes malhas analisadas.

As diferenças na qualidade das malhas ficam evidenciadas na Tabela 5.2. Nessa tabela, tem-se uma comparação quantitativa da média do erro relativo nodal, em percentual, dos resultados obtidos com as malhas mais grossas em relação à malha de referência (mais refinada).

Tabela 5.2: Média dos erros relativos, em percentual, das malhas analisadas em relação à malha muito fina para as curvas dos perfis de velocidade.

Malha (<i>i</i>)	MER_i
1	24,79%
2	13,47%
3	8,52%
4	6,42%

A média do erro relativo, em percentual, é dada por:

$$\%ER_j = \frac{|X_j^i - X_j^{ref}|}{X_j^{ref}} \times 100 \quad (5.1)$$

$$MER_i = \frac{\sum_{j=1}^N ER_j}{N} \quad (5.2)$$

onde X_j^i corresponde a pontos da curva do perfil de velocidades, o índice i corresponde à malha analisada e j corresponde aos pontos da curva que variam entre 1 e N . O termo X_j^{ref} corresponde a pontos da malha de referência, ou seja, a malha muito fina. O termo ER_j retorna o erro relativo de cada ponto da curva e o termo MER_i retorna a média dos erros relativos da malha analisada em comparação à malha de referência.

Percebe-se que a malha 1 (Grossa) apresenta o maior erro em relação à malha de referência (24,79%), e a malha com o menor erro foi a malha 4 (6,42%).

Outro parâmetro utilizado para verificar a qualidade das malhas é a vazão volumétrica. Considerando a geometria do problema juntamente com a velocidade de entrada do fluido é possível calcular a vazão do mesmo. Ao se resolver computacionalmente o problema é possível verificar a ocorrência de variações nesses parâmetros, devido principalmente a qualidade da malha. Na Tabela 5.3 são apresentados os valores das vazões obtidos com a simulação computacional nas diferentes malhas analisadas. Como era de se esperar, observa-se que quanto mais refinada a malha menor o seu erro.

Também foi realizada uma análise da influência da qualidade da malha sobre os parâmetros de otimização do problema, como apresentado na Tabela 5.4. Observa-se que a tensão de cisalhamento (TCT) sofre grande influência da qualidade da malha, ao contrário da variância da DTR (σ^2), a qual não apresenta grandes variações. A tensão cisalhante total

(TCT) sofre mais com a qualidade da malha devido ao refinamento necessário para a captação dos vórtices de pequena escala.

Tabela 5.3: Vazões para as diferentes malhas e erro comparado com valor real.

Malha	Vazão (m^3 / s)	Vazão (l / min)	% erro
1 (Grossa)	$7,07 \times 10^{-3}$	424,26	9,97%
2	$7,43 \times 10^{-3}$	446,03	5,35%
3	$7,49 \times 10^{-3}$	449,99	4,51%
4	$7,55 \times 10^{-3}$	453,104	3,85%
5 (Fina)	$7,55 \times 10^{-3}$	453,10	3,85%
Vazão real	$7,85 \times 10^{-3}$	471,23	--

Tabela 5.4: Comparação entre os parâmetros de otimização para as malhas computacionais analisadas – Reator contínuo [500s].

Malha	$\bar{t}$ [s]	σ^2 [s ²]	TCT ponderada [m ² /s ²]	% Erro σ^2	% Erro TCT ponderada
1	152,18	10796,02	9,55E-5	4,19%	19,23%
2	146,63	10495,04	1,00E-4	1,28%	15,40%
3	145,76	10410,74	1,11E-4	0,47%	6,09%
4	145,21	10342,36	1,20E-4	0,19%	1,35%
5	145,41	10361,95	1,18E-4	-	-

Na Tabela 5.5 é exposto o custo computacional das diferentes malhas analisadas. Esta análise é decisiva para o processo de otimização acoplado com CFD, pois, a escolha de uma malha computacional que apresente uma boa convergência com um menor custo computacional é essencial para o sucesso dos projetos de otimização.

Tabela 5.5: Custo computacional das malhas analisadas [500s] – Reator contínuo.

Malha	Custo computacional ¹⁶
1	21561 s / 5,60 horas
2	84046 s / 23,34 horas
3	112861s / 31,35 horas
4	192835 s / 53,56 horas
5	317218 s / 88,11 horas

¹⁶ Custo computacional referente ao tempo de processamento do solver pisoScalarFoamAdjTime para 500s de simulação de escoamento [24 processadores Node[1-6] Monalisa.

Com as informações da qualidade da solução obtida e o custo computacional entre as diferentes malhas computacionais analisadas é possível escolher uma malha que garanta a precisão desejada com o menor custo computacional. Assim, considerando que a malha 5 (fina) possui uma quantidade de células alta, o que demanda um elevado custo computacional, faz com que se torne praticamente inviável para o problema proposto de otimização a sua utilização. Considerando os valores da análise de qualidade de malha e o custo computacional optou-se por iniciar o processo de otimização com uma malha computacional com as especificações (qualidades) da malha 2.

É importante destacar que todas as malhas analisadas respeitam a condição de estabilidade imposta pelos algoritmos/métodos utilizados pelo solver CFD utilizado, isto é, o número de Courant inferior a 1 (OPENCFD, 2013). É garantida a estabilidade dos algoritmos/métodos utilizados no problema somente para o número de Courant menores que um. Assim as malhas foram analisadas de forma que este limite fosse respeitado, onde o intervalo de tempo durante a simulação é ajustado de acordo com o número máximo de Courant.

5.1.2 Tempo de simulação necessário para estabelecimento do regime permanente

Como descrito por FORTUNA, 2000, “quando se resolve um problema transiente, busca-se a evolução temporal efetiva da grandeza física”. Portanto, uma condição inicial fisicamente correta é necessária para que esta evolução seja realmente representativa do problema abordado. Isto ocorre no presente trabalho ao calcular a distribuição do tempo de residência (DTR). Para que a DTR seja corretamente calculada é necessário que a injeção do traçador seja realizada quando o escoamento esteja em regime permanente, ou seja, não existam grandes flutuações nos perfis de velocidade.

Na aplicação do reator contínuo o escoamento parte do repouso no tempo de simulação $t=0s$. Assim, é necessário verificar o tempo de simulação necessário para que o escoamento entre em regime e a partir deste tempo iniciar a injeção do traçador.

Devido às grandezas físicas e o custo computacional envolvido no problema abordado torna-se imprescindível a correta metodologia de solução do problema. Para isto, foi utilizada a técnica conhecida como formulação pseudotransiente (FORTUNA, 2000). Nesta técnica, uma solução estacionária é utilizada como condição inicial para a simulação transiente. Assim, neste caso, inicia-se a simulação do escoamento utilizando-se um método estacionário, turbulento para escoamento incompressível (*simpleFoam*). Após a convergência do método

utiliza-se a sua solução, como por exemplo, os valores dos perfis de velocidade e pressão, como condição inicial do solver transiente (*pisoScalarFoamAdjTime*).

A avaliação do correto funcionamento do processo de simulação CFD entre o caso permanente e transiente foi feita considerando a geometria com $H_{out}=1,0$ m e $\alpha=0^\circ$ e nela foram introduzidos dois pontos de monitoramento no interior dos reatores como apresentado na Figura 5.2.

Figura 5.2: Pontos de monitoramento e malha analisada (Geometria $H_{out}=1.0$ e $\alpha=0^\circ$).

Esses pontos de monitoramento servem para que os perfis de velocidade sejam acompanhados durante o tempo total de simulação. O acompanhamento dos perfis de velocidade permite a obtenção de um gráfico da velocidade em função do tempo. Deste modo, quando os perfis de velocidades apresentam apenas uma pequena oscilação é considerado que o escoamento entrou em regime permanente (Figura 5.3-A). Este processo ocorre quando o método estacionário converge e a partir deste ponto utilizam-se esses perfis para que a simulação transiente seja iniciada, como descrito anteriormente. Na Figura 5.3-B, observa-se a não ocorrência de flutuações nos perfis de velocidade. Assim, inicia-se a injeção do traçador e o cálculo da DTR.

Figura 5.3: Perfis de velocidade em x, y e z adquiridos dos pontos de monitoramento: A) Solver estacionário – simpleFoam; B) Solver transiente - pisoScalarFoamAdjTime.

5.1.3 Desempenho computacional obtido com as estratégias de paralelização

Com o intuito de reduzir o tempo computacional do problema, o processo foi paralelizado como descrito na Seção 4.3.3. As primeiras análises comparam a paralelização do OpenFOAM para executar o solver pisoScalarFoamAdjTime até 50 iterações. A malha computacional utilizada para os testes de desempenho computacional é a malha 2 descrita na Tabela 5.1 com 198395 volumes tetraédricos.

A Figura 5.4 apresenta o tempo de processamento e o *speedup* obtido na estação de trabalho SGI Monalisa e Pegasus. A estação de trabalho Pegasus é composta por um processador Core i7 3930K, 3,2 GHz com 32GB de memória RAM e GPU Nvidia GTX 590.

Figura 5.4: Análise de desempenho obtido com a paralelização do solver CFD.

O *speedup* é definido aqui como a razão entre o tempo total de processamento obtido utilizando-se um processador dividido pelo tempo total de processamento utilizando-se N_{proc} processadores. O valor ótimo teórico de *speedup* utilizando-se N_{proc} processadores é N_{proc} . Em casos práticos, o tempo de comunicação entre os processadores reduz o valor do *speedup* para valores inferiores ao máximo teórico. No presente trabalho, o *speedup* obtido ultrapassa o *speedup* teórico em algumas situações devido principalmente às características de hardware. Para os processadores da estação de trabalho Monalisa e Pegasus, a quantidade de cores físicos são respectivamente 12 (nó 1 a 6 do cluster – Tabela 4.3) e 6, o que pode ocasionar um melhor desempenho do processador e utilização da memória cache.

Outro nível de paralelização envolvido no software PyCFD-O envolve a paralelização do método de otimização. Como já discutido na Seção 4.3.3 os métodos de otimização avaliados neste trabalho são paralelizáveis. Assim, ao utilizar-se a estação de trabalho Monalisa é possível designar que cada nó de processamento do cluster seja responsável pela solução de uma geometria do reator. Internamente, cada nó de processamento do cluster realiza o processamento em paralelo do solver CFD em seus processadores internos, obtendo-se o desempenho descrito acima.

Como exemplo de desempenho, considerando-se a estação de trabalho Monalisa, o custo computacional do solver CFD de uma geometria com a malha 2 executada em paralelo em um nó computacional com 24 cores e tempo de simulação de escoamento de 500s é

aproximadamente 23 horas. Considerando por exemplo um caso onde 100 avaliações do ciclo de otimização são necessárias, resultaria em um tempo computacional de aproximadamente 96 dias. Com a implementação da paralelização mestre-escravo como descrita na Seção 4.3.3, resolvendo as várias geometrias em paralelo é possível reduzir o tempo total de processamento linearmente. Isso possibilita que o tempo total de processamento do ciclo de otimização para o problema abordado se torne praticável, como exemplo de 96 dias para 16 dias utilizando-se 6 nós de processamento.

O desempenho obtido também se deve ao fato que a quantidade de informação transferida entre o mestre e escravos é muito pequena: quatro parâmetros em uma direção e dois na direção inversa. Assim, o tempo de comunicação despendido é praticamente desprezível se comparado com o tempo requerido pela CPU na avaliação dos objetivos em cada nó de processamento. Pequenos desvios do *speedup* ótimo podem ocorrer principalmente devido aos efeitos de contorno no final de cada iteração de otimização, isto é, quando algum escravo torna-se inativo por um curto período de tempo esperando pelo início da próxima iteração.

5.1.4 Análise da influência da geometria na hidrodinâmica do escoamento

Uma análise preliminar foi realizada a fim de determinar a influência dos parâmetros geométricos α e H_{out} no comportamento hidrodinâmico do reator contínuo analisado. O objetivo é analisar a maneira como os parâmetros geométricos afetam a DTR e a tensão de cisalhamento. Para a execução dos testes, cinco geometrias com diferentes α e H_{out} foram consideradas, como apresentado na Tabela 5.6. O tempo total de simulação de escoamento foi 500s e foi adotada uma malha computacional com as mesmas características da malha 5.

Tabela 5.6: Geometrias avaliadas – Reator contínuo.

Geometria	$\alpha[^{\circ}]$	$H_{out}[m]$
0_0.1	0 (tangencial)	0,1
0_1.3	0 (tangencial)	1,3
60_0.8	60	0,8
90_0.1	90 (perpendicular)	0,1
90_1.3	90 (perpendicular)	1,3

A Tabela 5.7 apresenta os valores dos parâmetros analisados no processo de otimização: tempo médio de residência ($\bar{t}$), variância (σ^2) e a média ponderada da tensão de

cisalhamento das geometrias analisadas. Na Figura 5.5 temos as curvas da DTR onde é possível ver com mais clareza a influência da geometria na hidrodinâmica do escoamento.

Tabela 5.7: Parâmetros analisados [500s de simulação de escoamento].

Geometria	$\bar{t}$ [s]	σ^2 [s ²]	TCT ponderada [Pa]
0_0.1	134,94	12852,79	0,150
0_1.3	145,70	9405,18	0,127
60_0.8	145,17	10652,17	0,102
90_0.1	116,68	18787,76	0,058
90_1.3	138,86	11541,30	0,067

Figura 5.5: Curvas DTR dos reatores contínuos analisados: A) $H_{out} = 0.1\text{ m}$ $\alpha = 0^\circ$ e $H_{out} = 1.3\text{ m}$; B) $H_{out} = 0.8\text{ m}$ $\alpha = 60^\circ$; C) $H_{out} = 0.1\text{ m}$ $\alpha = 90^\circ$; D) $H_{out} = 1.3\text{ m}$ $\alpha = 90^\circ$.

Os resultados apresentados na Figura 5.5 mostram que a porção prolongada da curva $E(t)$ para altos valores do tempo de residência médio ($\bar{t}$) indica a presença de zonas mortas. Também é possível observar nas curvas $E(t)$ a presença de curto-circuito. Os efeitos do curto-circuito são mais evidentes na Figura 5.5-C, curva 90_0.1. O curto-circuito na geometria 90_0.1 resulta em um valor grande na variância (σ^2) (Tabela 5.7). A magnitude da variância

é um indicador do espalhamento da distribuição do tempo de residência (FOGLER, 2005). A ocorrência de curto-circuito indica a existência de pouca ou nenhuma troca de material com regiões bem misturadas, e conseqüentemente, a não ocorrência de reações. As zonas de estagnação mantém o material dentro do reator por um curto ou longo período de tempo necessário para que a reação ocorra. Em ambos os casos, as zonas de estagnação são prejudiciais para o sucesso da reação química. A geometria 0_1.3 apresenta a menor variância, como pode ser visto na Tabela 5.7 e na Figura 5.5, curva 0_1.3. Entretanto, a geometria 0_1.3 não apresenta a menor tensão de cisalhamento.

Analisando a Tabela 5.7 é possível observar novamente que as geometrias que possuem o tempo de residência médio mais próximo do TRT (150s) são as geometrias que possuem as menores variâncias (σ^2). Nota-se também que as geometrias que possuem as maiores H_{out} apresentam os menores σ^2 . No entanto, estas geometrias (0_1.3 e 60_0.8) possuem valores altos para a tensão de cisalhamento ponderada. As melhores geometrias considerando a tensão de cisalhamento são as geometrias formadas pelo ângulo de entrada do fluido no reator perpendicular ($\alpha = 90^\circ$).

A influência dos parâmetros geométricos no comportamento do escoamento pode ser observada na Figura 5.6, Figura 5.7 e Figura 5.8. A Figura 5.6 apresenta a comparação das linhas de corrente nas diferentes geometrias estudadas. Ao variar o ângulo de entrada do escoamento é possível controlar o comportamento das linhas de corrente. A sensibilidade das linhas de corrente com a modificação do ângulo de entrada fica explícita entre a Figura 5.6-A e Figura 5.6-C. A entrada tangencial ($\alpha = 0^\circ$) impulsiona o fluido dentro do reator, minimizando recirculações formando linhas de correntes mais comportadas, resultando em uma melhor DTR, o que são desejáveis no processo químico.

A Figura 5.7 apresenta o comportamento da distribuição do traçador para diferentes geometrias do reator. Como já observado na distribuição das linhas de corrente a geometria com a entrada tangencial apresenta um melhor comportamento em relação à distribuição do fluido, diferentemente da geometria com o ângulo de entrada perpendicular ($\alpha = 90^\circ$). Na Figura 5.7-C é possível observar claramente que na geometria com o ângulo perpendicular o fluido se choca contra a tubulação de saída causando zonas de recirculação e curto-circuito.

Figura 5.6: Linhas de corrente: A) Geometria $H_{out} = 0.1$ e $\alpha = 0^\circ$.; B) Geometria $H_{out} = 0.8$ e $\alpha = 60^\circ$.; C) Geometria $H_{out} = 1.3$ e $\alpha = 90^\circ$.

Figura 5.7: Distribuição do traçador no reator contínuo [500s]: A) Geometria $H_{out} = 0.1$ e $\alpha = 0^\circ$.; B) Geometria $H_{out} = 0.8$ e $\alpha = 60^\circ$.; C) Geometria $H_{out} = 1.3$ e $\alpha = 90^\circ$.

Finalmente, na Figura 5.8 é possível verificar a influência da geometria do reator na tensão de cisalhamento. Observa-se que a geometria com entrada tangencial Figura 5.8-A apresenta os maiores valores da tensão de cisalhamento, conforme apresentado na Tabela 5.7. A ação do cisalhamento é observada na geometria tangencial em toda a parede do reator com maior intensidade se comparada com as outras geometrias (Figura 5.8-B e Figura 5.8-C). Percebe-se também a elevada taxa de cisalhamento no afunilamento do fluido na saída do reator.

Figura 5.8: Distribuição da tensão de cisalhamento ponderada no reator contínuo [500s]: A) Geometria $H_{out} = 0.1$ e $\alpha = 0^\circ$. B) Geometria $H_{out} = 0.8$ e $\alpha = 60^\circ$. ; C) Geometria $H_{out} = 1.3$ e $\alpha = 90^\circ$.

Como pode ser observado é possível verificar a influência significativa dos parâmetros geométricos, α e H_{out} na hidrodinâmica e consequentemente no rendimento da reação química desejada.

5.1.5 Avaliação dos métodos de otimização NSGA-II e EMOO

Com o objetivo de avaliar o correto funcionamento do software PyCFD-O e analisar o desempenho dos métodos de otimização utilizados, foram realizados testes preliminares em

uma geometria reduzida do reator contínuo, em razão do alto custo computacional da geometria de 1m^3 .

Para redução de escala do reator contínuo (*scale-down*) foram considerados os valores de um quarto para os parâmetros geométricos adotados no reator de 1m^3 , totalizando um volume de $0,019\text{m}^3$. A Tabela 5.8 apresenta os parâmetros geométricos adotados para a geração dos reatores.

Tabela 5.8: Parâmetros geométricos para reator reduzido.

Parâmetros	Valores
H_{Bio}	0,375 m
H_{out}	0,1 – 0,28 m
L_{in}	0,125 m
D_{Bio}	0,25 m
$d_{in} = d_{out}$	0,25 m
α	$0^\circ - 90^\circ$

Para a geometria reduzida do reator foram adotadas as mesmas configurações, condições de contorno e modelos numéricos para o *solver* CFD escolhidas para o reator de 1m^3 , alterando-se somente a velocidade de entrada do fluido para 0,25 m/s, respeitando o tempo de residência teórico (TRT) de 150 segundos. Para a malha computacional foram mantidos os mesmos padrões de refinamento do programa PyCFD-O que foram analisados no teste de malha. Com a redução da geometria do problema as malhas computacionais que antes variavam de aproximadamente 100000 a 600000 de volumes tetraédricos para o reator de 1m^3 foram reduzidas para 31000 a 180000 volumes tetraédricos.

Os parâmetros utilizados nos métodos de otimização, NSGA-II e EMOO são apresentados na Tabela 5.9. Com o objetivo de avaliar qual método apresenta uma melhor distribuição da fronteira de Pareto são utilizados os mesmos parâmetros para ambos os métodos.

Os parâmetros de otimização avaliados pelo programa PyCFD-O seguem os mesmos dos apresentados para a otimização do reator contínuo de 1m^3 . Estes parâmetros são a H_{out} e α , que variam de acordo com os valores apresentados na Tabela 5.8. Assim o método de otimização busca encontrar a menor variância da DTR e a menor tensão de cisalhamento ponderada. Após a execução do programa PyCFD-O para o problema abordado obtivemos as soluções não dominadas para ambos os métodos.

Tabela 5.9: Parâmetros dos algoritmos de otimização.

Parâmetro	Valor
Tamanho da população, N	10
Gerações, G	10
Probabilidade de mutação	0,5
Probabilidade de crossover	0,5
Distribuição de mutação	20
Distribuição de crossover	20

A Figura 5.9 apresenta as melhores fronteiras não dominadas dos dois métodos analisados. Os pontos pertencentes às fronteiras 0 correspondem aos pontos não dominados, possível fronteira de Pareto. Cada ponto pertencente à fronteira 0 corresponde a uma geometria otimizada de acordo com H_{out} e α . Durante o processo de busca, o fato de uma solução ser não dominada frente às propostas de solução já investigadas não implica que a mesma pertença à fronteira de Pareto.

A Figura 5.10 apresenta o comparativo entre as soluções não dominadas dos dois métodos analisados. Observa-se que ambos os métodos apresentam pontos próximos, no entanto com o método NSGA-II foram obtidas as melhores geometrias otimizadas.

Os testes realizados com uma geometria reduzida possibilitou a realização de uma análise de desempenho dos métodos e o correto funcionamento global do software PyCFD-O.

Figura 5.9: Fronteiras não dominadas: A) Algoritmo EMOO; B) Algoritmo NSGA-II.

Figura 5.10: Comparativo entre as melhores fronteiras não dominadas dos métodos de otimização EMOO e NSGA-II.

5.1.6 Dominância de Pareto e geometrias otimizadas – Reator contínuo (1m³)

Após a realização de vários testes, o algoritmo evolucionário de otimização foi executado utilizando-se 10 indivíduos para cada geração. As probabilidades aplicadas são apresentadas na Tabela 5.9. As simulações aplicadas no processo de otimização do reator contínuo de 1m³ foram configuradas de acordo com o que foi apresentado na metodologia, tendo o tempo total de simulação do escoamento de 200 segundos.

A variância e a tensão de cisalhamento ponderada resultante para todos os pontos do grupo de elite avaliado após as 10 gerações são apresentados na Figura 5.11. Os dois objetivos, variância e TCT ponderada são representados nos eixos x e y respectivamente. O conjunto de soluções não dominadas, possível fronteira de Pareto, representam as geometrias otimizadas. Na Figura 5.12, o conceito da busca pela fronteira de Pareto é novamente ilustrado, onde as geometrias otimizadas são comparadas com as geometrias base analisadas na Seção 5.1.4. Observa-se na Figura 5.12 o processo de busca das geometrias ótimas no espaço de soluções. A Tabela 5.10 apresenta os valores do tempo médio de residência, variância da DTR e a TCT ponderada para as geometrias pertencentes a fronteira não dominada.

Variações consideráveis foram testadas pelo procedimento de otimização, explorando todo o espaço de parâmetros antes de identificar as melhores soluções. É importante ressaltar que certamente essas variações das geometrias exploradas (220 geometrias) não teriam sido consideradas se o processo de otimização fosse realizado manualmente. A otimização automática requer neste caso um tempo computacional considerável, devido às dimensões do

reator analisado (1 m^3) e principalmente o tempo requerido de simulação do escoamento (200s). Mesmo com a limitação do tempo de máquina, considerando as geometrias otimizadas e as geometrias base houve uma redução de até 47% na variância do tempo de residência e uma diminuição em 48% na tensão de cisalhamento ponderada. Qualquer geometria na fronteira de Pareto pode ser utilizada para diferentes objetivos de projetos.

Figura 5.11: Resultado das melhores avaliações após 10 gerações do algoritmo NSGA-II: Possível fronteira de Pareto e configurações factíveis e infactíveis.

Figura 5.12: Comparativo das geometrias otimizadas e as geometrias base no domínio de busca [200s].

Tabela 5.10: Parâmetros de otimização das geometrias otimizadas [200s simulação de escoamento].

Geometria	$\bar{t}$ [s]	σ^2 [s ²]	TCT ponderada [m ² /s ²]
43.5_1.15	95,75	1790,12	8,41E-5
77.6_1.29	109,14	1944,20	7,10E-5
82.0_1.30	99,71	1987,79	7,00E-5
62.7_1.28	106,39	1992,90	6,80E-5
77.7_1.26	100,41	2033,31	6,64E-5
58.8_1.26	104,05	2059,15	6,52E-5
87.3_1.19	93,28	2285,59	6,50E-5

Na Figura 5.12 é possível observar a localização das geometrias base no domínio de soluções factíveis e compará-las com os parâmetros de otimização das geometrias da fronteira de Pareto. A Figura 5.13 apresenta o comparativo das curvas DTR de três geometrias otimizadas que estão na fronteira de Pareto.

Figura 5.13: Curvas DTR de três geometrias otimizadas.

A partir da Figura 5.11 e dos dados apresentados na Tabela 5.10, é possível inferir que a geometria que apresenta a menor variância (σ^2) é a geometria 43.5_1.15 ($\alpha = 43,5^\circ$, $H_{\text{out}} = 1,15$ m). No entanto, a geometria 43.5_1.15 possui a pior TCT ponderada entre as geometrias otimizadas. Por outro lado, a geometria 87.3_1.19 apresenta a melhor TCT ponderada, mas a maior variância; resultado da formação de curto-circuito como pode ser visto na curva 87.3_1.19 da Figura 5.13. Deste modo, dependendo das necessidades do problema é possível escolher uma das geometrias otimizadas com a maior ou menor variância ou TCT ponderada. É possível também considerar uma geometria que apresente os menores parâmetros de otimização combinados, como por exemplo, a geometria 62.7_1.28, ou a geometria que

apresente melhores condições de projeto. A comparação entre as linhas de corrente das três geometrias otimizadas analisadas pode ser observada na Figura 5.14.

Figura 5.14: Geometrias otimizadas e linhas de corrente: A) $\alpha = 43,5^\circ$, $H_{out} = 1,15$ m; B) $\alpha = 62,7^\circ$, $H_{out} = 1,28$ m; C) $\alpha = 87,3^\circ$, $H_{out} = 1,28$ m

Após a realização de vários testes e considerando-se todas as geometrias analisadas no processo de otimização é possível criar um banco de dados e em seguida ponderá-las. Esta análise é feita através do uso de superfície de resposta. Uma superfície de resposta corresponde a uma figura obtida quando uma variável resposta é representada graficamente em função de um ou mais fatores do processo. Com isso, é possível analisar a patologia de otimização do problema abordado.

Em muitas situações é interessante avaliar as relações existentes entre os principais fatores que compõem um processo e uma variável resposta de interesse. Nestas condições, tem-se como objetivo determinar qual é a condição de operação do processo que levará à obtenção de um valor ótimo para a variável resposta. Nas Figura 5.15 e Figura 5.16 tem-se a representação das superfícies de resposta para os parâmetros de otimização abordados no processo. Na Figura 5.15, tem-se a superfície de resposta considerando a variância (σ^2), onde é possível observar o comportamento deste parâmetro com a variação da geometria. Com as geometrias próximas aos menores valores de altura de saída (H_{out}) e o ângulo de entrada (α)

próximo a 90 graus observa-se um pico máximo na variância da DTR. Isto é explicado por essas geometrias apresentarem curto-circuito. A presença de curto circuito nessas geometrias, H_{out} baixos e com α igual a 90 graus foi observada na seção 5.1.4. O ângulo de 90° influência significativamente na variância. A formação de curto-circuito para essas geometrias também pode ser observada no tempo médio de residência (Figura 5.17), com valores extremamente baixos. Já os valores com as menores variâncias correspondem a uma região entre os ângulos de 60 e 80 graus e um pico mínimo próximo ao ângulo de 43 graus, como pode ser melhor observado nas curvas de nível da Figura 5.15. O efeito das menores variâncias que possuem uma relação direta com o tempo médio de residência também pode ser observado na Figura 5.17. A região com as menores variâncias corresponde à região com o maior tempo de residência, o qual se aproxima mais do tempo de residência teórico (TRT).

Já na Figura 5.16 temos a superfície de resposta correspondente a tensão de cisalhamento ponderada. Nesse caso, pode-se observar os menores valores da TCT ponderada decrescendo com o aumento do ângulo de entrada em conjunto com a altura de saída. Os valores elevados da tensão de cisalhamento para as geometrias próximas ao ângulo de entrada tangencial ($\alpha = 0^\circ$) se devem ao fato da elevada velocidade em que o fluido entra no reator tangenciando sua parede. Esse efeito pode ser observado na Figura 5.8-A. A tensão de cisalhamento também é alta no afunilamento do fluido na saída do reator. Observando-se novamente a Figura 5.16 é possível observar que esse efeito é minimizado com uma altura do tubo de saída mais elevada. Isto pode ser justificado devido à formação de um ciclone dentro do reator (Figura 5.6). Quanto mais elevado o tubo de saída do fluido menor é influência da formação do ciclone.

Destas análises conclui-se que as geometrias com α entre 60 e 80 graus, juntamente com uma altura de saída (H_{out}) entre 1,2 e 1,3 metros apresentam as melhores condições de projeto para os parâmetros de otimização avaliados. Um ângulo de entrada nesta faixa não causaria uma tensão de cisalhamento significativa na parede do reator. Uma maior altura H_{out} proporciona uma permanência mais longa do fluido no reator, diminuindo a influência da tensão de cisalhamento no afunilamento da tubulação na saída do reator.

Figura 5.15: Superfície de resposta: σ^2 - Reator contínuo.

Figura 5.16: Superfície de resposta: TCT ponderada – Reator contínuo.

Figura 5.17: Superfície de resposta: Tempo médio de residência ($\bar{t}$) – Reator contínuo.

A fronteira de Pareto juntamente com as superfícies de resposta possibilitam a escolha de uma geometria otimizada para o processo químico envolvido, de acordo com os parâmetros estabelecidos.

A Figura 5.18-A apresenta as linhas de corrente no reator e o campo vetorial em um plano aplicado em uma provável geometria ótima, com ângulo de entrada de 60 graus. Nessa figura, é possível observar o comportamento do fluido na entrada do reator. Buscou-se observar se ocorre a formação de zonas de recirculação na entrada do fluido na região cilíndrica do reator. Com o auxílio da Figura 5.18-A e juntamente com a plotagem da vorticidade do escoamento nas três componentes, como apresentado na Figura 5.18-B observa-se a formação de pequenas zonas de recirculação na região de entrada do fluido. No entanto, estas recirculações não causam grandes interferências na distribuição do tempo de residência do fluido.

Figura 5.18: Análise geometria $\alpha = 60^\circ$: A) Linhas de corrente e campo vetorial; B) Vorticidade.

5.2 Reator *airlift*

Nessa seção serão apresentados os resultados referentes ao reator *airlift*. Inicialmente foi realizada uma análise da malha computacional adotada no processo e a validação do modelo CFD. Também são apresentadas as análises prévias necessárias para o correto funcionamento do processo de otimização e a análise realizada para ponderar a influência da geometria do reator nos parâmetros de otimização. Finalmente, são descritos os resultados obtidos com o processo de otimização do software PyCFD-O no reator *airlift*.

Cada tópico a seguir expõe os resultados obtidos, de acordo com o que foi detalhado na metodologia e planejamento para cada experimento. As análises realizadas tem o intuito de

verificar e validar o correto funcionamento do software PyCFD-O e a obtenção de uma geometria de acordo com os parâmetros de otimização abordados.

5.2.1 Análise da malha computacional

Como exposto na Seção 4.2.3, a malha computacional deve apresentar certas características considerando-se a modelagem do escoamento bifásico e turbulência. Assim, considerando-se as necessidades da malha e o custo computacional associado foram utilizadas duas malhas, como descrito na Tabela 5.11.

Tabela 5.11: Características da malha computacional - Reator *airlift*.

Malha	Número de Nós	Número de Volumes Hexaédricos	Volume total da geometria [m ³]
1	1431936	1372908	0,061586
2	1991808	1920696	0,061818

Ambas as malhas possuem características que possibilitam a sua utilização para a solução do modelo CFD. Para as simulações utilizadas no reator *airlift* foi considerado o número de Courant inferiores a 0,1. A qualidade da malha 1 será validada nas seções seguintes.

Considerando os custos computacionais expostos na Tabela 5.12 o software PyCFD-O inicia a execução com a malha 1. Se algum critério de convergência não é satisfeito de acordo com a geometria abordada, automaticamente a malha 2, mais refinada é utilizada. Este processo é necessário para garantir a fluidez do programa, caso algum problema de convergência ocorra com a malha 1.

Tabela 5.12: Custo computacional das malhas analisadas - Reator *airlift*.

Malha	Custo computacional ¹⁷
1	334718s / 92,98 horas
2	505080s / 140,3 horas

¹⁷ Custo computacional referente ao tempo de processamento do solver interScalarDyFoam para 50s de simulação de escoamento em dois nós de processamento com 24 processadores cada na estação de trabalho Monalisa.

Como o custo computacional das simulações é um ponto crítico no processo de otimização, foi necessário ponderar entre a melhor qualidade da solução obtida com o menor computacional.

5.2.2 Tempo de simulação necessário para estabelecimento do regime permanente – Reator *airlift*

Como descrito na seção 5.1.2 para o reator contínuo, o escoamento no reator *airlift* parte do repouso no tempo de simulação $t=0s$. Assim, torna-se necessário verificar o tempo de simulação requerido para que o escoamento entre em regime e a partir desse momento iniciar a injeção do traçador. Para isso, três pontos de monitoramento foram inseridos no reator como ilustrado na Tabela 5.13 e Figura 5.19. Com esses pontos de monitoramento é possível acompanhar os perfis de velocidade durante o tempo total de simulação. A malha computacional utilizada é a 1 e o tempo total de simulação do escoamento é de 60 segundos.

Tabela 5.13: Pontos de monitoramento reator *airlift*.

Ponto 1	Ponto 2	Ponto 3
(0,33; 0,14; 0,015)	(0,33; 0,025; 0,015)	(0,052; 0,14; 0,35)

Figura 5.19: Pontos de monitoramento reator *airlift*.

A Figura 5.20 apresenta a velocidade versus o tempo para os três pontos de monitoramento. Analisando-se o gráfico é possível observar que nos pontos 1 e 3 ocorrem

flutuações durante todo o tempo de escoamento. Isto pode ser explicado pela localização dos pontos em regiões onde ocorre a formação de vórtices. Já no ponto 2 verifica-se uma grande variação das velocidades até o tempo 9 s. A partir desse tempo de simulação do escoamento é possível iniciar a injeção do fluido traçador para o cálculo da DTR.

Figura 5.20: Perfis de velocidade em x, y e z adquiridos dos pontos de monitoramento em 60s: Reator *airlift*.

5.2.3 Perfil de velocidade – Comparativo PIV – CFD

Nesta seção são apresentados os resultados obtidos com a aplicação da técnica PIV no reator *airlift*, como descrito na Seção 4.2 e uma comparação com os resultados obtidos com o CFD. Para os testes CFD foi utilizada a malha 1, como descrita na Tabela 5.12.

A Figura 5.21 apresenta as linhas de corrente obtidas com a análise PIV e CFD, juntamente com a região onde foram realizadas as aquisições experimentais para o PIV. O comparativo entre o PIV - LaVision e o CFD através dos campos de velocidades é apresentado na Figura 5.22. Na Figura 5.21 e Figura 5.22 observa-se a formação de zonas de recirculação tanto do lado esquerdo como do lado direito da placa central. A recirculação do lado esquerdo apresenta valores de magnitude de velocidade superiores, devido à zona de ascensão das bolhas. Já a zona de recirculação do lado direito é caracterizada pela menor

intensidade da velocidade e a formação de dois vórtices. A modelagem CFD (Figura 5.22-B) possui o mesmo comportamento da análise PIV (Figura 5.22-A), com a formação das zonas de recirculação, respeitando uma diferença no tamanho do vórtice. No lado direito da placa, tem-se a presença de vórtices formados pela presença do escoamento sobre a placa.

Figura 5.21: Linhas de corrente reator *airlift* e região de aquisição do PIV no reator.

Figura 5.22: Comparativo PIV - LaVision e CFD do campo de velocidades num plano frontal - Reator *airlift*.

A Figura 5.23 ilustra os perfis de velocidade em uma seção transversal do reator obtidos com o PIV (*in-house* e LaVision) e CFD. Nota-se que na região de ascensão das bolhas tem-se valores de velocidade muito superior às observadas nas outras regiões do

reator. Após a região de ascensão das bolhas ocorre um decaimento da velocidade e logo em seguida, ocorre novamente um leve aumento. Isto se deve à formação da região de recirculação, como observado nas Figura 5.21 e Figura 5.22. Na parte direita do reator verifica-se uma velocidade superior próximo à parede, devido à não influência da formação dos vórtices ocasionados pela placa (observar Figura 5.21 e Figura 5.22). Considerando os perfis de velocidade obtidos pela simulação CFD e PIV, constata-se um deslocamento do pico de velocidade, ocasionados pelas oscilações da pluma de bolhas. Também é possível observar na Figura 5.23 um comportamento mais oscilatório nas aquisições por PIV, o que poderia ser causado pela presença dos vórtices observados durante os experimentos. Assim, mesmo com as diferenças observadas entre as aquisições experimentais e CFD, considera-se a modelagem CFD satisfatória pelas características de sua malha e custo computacional.

Figura 5.23: Comparativo perfil de velocidade entre PIV e CFD - Reator *airlift*.

5.2.4 Tempo de circulação e mistura

Como descrito na Seção 4.2.2, as técnicas de pulso de pH e da partícula foram utilizadas para a determinação do tempo de circulação e mistura. Os dados experimentais foram comparados com os resultados das simulações CFD, como apresentado na Seção 4.2.2.1.

Na Figura 5.24 são apresentadas as curvas de concentração do fluido traçador obtidas nos três pontos de monitoramento para o caso experimental e CFD como detalhado na metodologia, Seção 4.2.2. As curvas foram normalizadas de acordo com a Equação (5.3). Com a normalização dos dados tem-se os valores da concentração do traçador variando entre 0 e 1. A normalização não altera a característica da curva e com ela é possível comparar lado a lado os dados experimentais e os obtidos a partir da simulação computacional.

$$C_{norm}[i] = \frac{C[i] - \min(C[i])}{\max(C[i]) - \min(C[i])} \quad (5.3)$$

Figura 5.24: Curvas com a concentração do fluido traçador normalizada - Experimental e CFD.

O tempo zero corresponde ao início da injeção do traçador no reator, com isso é possível obter o tempo necessário para que o fluido traçador alcance os pontos de monitoramento. Nos experimentos, os tempos medidos para os pontos 1, 2 e 3 foram 14, 13 e 14 segundos respectivamente. Já na simulação CFD os tempos foram 13, 12,3 e 12,7 segundos, respectivamente. Como o software de aquisição foi regulado para passos de tempo em segundos, não foi possível analisar os décimos de segundos para este caso. Como observado na Figura 5.24 o tempo medido experimentalmente e simulado apresentam uma boa concordância.

Ao analisar em detalhes a Figura 5.24, nota-se uma diferença nos tempos de medição entre os pontos e no comportamento das curvas ao longo das aquisições. Observa-se que o

fluido traçador atinge os pontos 2 e 3 antes que o ponto 1, indicando que a velocidade do fluido neste ponto é menor que nos outros pontos. Este comportamento pode ser explicado pela influência da região de recirculação. Comparando-se as linhas de corrente do problema nota-se claramente que os pontos 2 e 3 sofrem um efeito menos significativo em comparação ao ponto 1.

A Tabela 5.14 apresenta o resultado do tempo de circulação obtido pelo método da esfera para as dez medidas realizadas e a média. Como aferido na metodologia, o tempo de circulação foi definido como o tempo médio necessário para que um volume de controle se movimente da região de injeção do traçador até a região de ponto de monitoramento. Com isso foi possível verificar a ocorrência de concordâncias com o tempo médio obtido pela técnica de pulso de pH.

Tabela 5.14: Tempos de circulação obtidos pelo método da esfera.

Medição	Tempo de circulação [s]
1	15,44
2	16,87
3	15,03
4	15,80
5	15,85
6	15,75
7	14,02
8	14,56
9	17,84
10	15,10
Média	15,62

Com o tempo médio obtido pelo método da esfera de 15,62 segundos e pela técnica pulso de pH de 13,67 segundos é possível constatar uma concordância entre os valores, considerando-se que as medições pelo método da esfera são altamente suscetíveis a erros durante a aquisição.

Na Figura 5.25 é apresentado um comparativo entre os dados experimentais e os resultados da simulação para os diferentes pontos de monitoramento. Pode-se observar novamente uma concordância entre as curvas no início da medição do traçador e também no comportamento de estabilização das curvas. As diferenças de concentração verificadas no decaimento das curvas podem ser explicadas pela difusividade do ácido e o fluido traçador adotado nas simulações possuem a característica de um traçador passivo (sem difusividade).

O tempo necessário para que uma substância se misture uniformemente dentro do reator é considerado como o tempo de mistura. Quando o valor da variável eleita se estabiliza, como pode ser observado na Figura 5.25, o tempo de mistura é determinado. Mais precisamente, o tempo necessário para que o valor do fluido traçador atinja 95% de seu valor final é chamado tempo de mistura. Desse modo, considerando-se os valores de estabilização tanto das simulações CFD e dos experimentos nota-se que o tempo de mistura em ambos os casos é de aproximadamente 30 segundos.

Figura 5.25: Comparação entre CFD e experimental das curvas DTR nos pontos de monitoramento.

5.2.5 Análise da influência da geometria na hidrodinâmica do escoamento

Uma análise preliminar foi realizada com o intuito de determinar a influência dos parâmetros geométricos dy (variação horizontal da placa) e H_z (variação vertical da placa) no comportamento hidrodinâmico do reator. O objetivo é analisar como os parâmetros geométricos afetam a DTR e a tensão de cisalhamento. Cinco geometrias com diferentes dy e H_z foram definidas, como apresentado na Tabela 5.15. O tempo total de simulação de escoamento aplicado a estes testes foi 50s.

Tabela 5.15: Geometrias analisadas - Reator *airlift*.

Geometria	dy [m]	H _z [m]
A	0,03	0,40
B	0,03	0,51
C (Geometria base)	0,08	0,51
D	0,12	0,40
E	0,12	0,51

A Figura 5.26 ilustra a comparação entre as linhas de corrente para as geometrias analisadas. A formação de pequenos vórtices causada principalmente pela geometria retangular do reator e os vórtices formados entre as plumas de bolhas foram desconsiderados por não sofrerem grande influência da variação do posicionamento da placa interna do reator.

Com as placas próximas aos injetores (casos A e B) nota-se a não ocorrência de formação de grandes zonas de recirculação nesta região. No entanto, formam-se grandes vórtices na região de descida do reator. A característica destes vórtices se diferencia devido à altura da placa. Com a placa mais baixa (Figura 5.26-A), o vórtice apresenta uma característica mais alongada se comparado ao caso B (Figura 5.26-B). Deste modo, é possível inferir que ambas as geometrias A e B apresentam um tempo de permanência do fluido baixo na região de subida e um alto tempo de residência na região de descida do fluido no reator.

Considerando-se o caso C (geometria base - Figura 5.26-C) é possível observar a formação de duas grandes zonas de recirculação, uma na região de subida e outra na região de descida. A formação da recirculação na região de subida se deve ao fato da placa estar mais afastada dos injetores, permitindo assim, a formação de vórtices mais amplos devido a força motriz que as bolhas exercem na fase líquida. A segunda zona de recirculação presente na região de descida ocasionada pela placa, não é tão grande se comparada aos casos A e B, mas apresenta uma grande influência na DTR. No caso B já é possível observar um tempo de residência maior na região de subida.

Nos casos D e E a placa se encontra próxima a parede no lado oposto dos injetores. Neste caso extremo se forma uma grande zona de recirculação na parte esquerda da placa e um escoamento laminar entre a placa e parede direita do reator. As zonas de recirculação sofrem influência da altura da placa como observado nos casos A – B e D – E. Enquanto no caso D ocorre a formação de outro vórtice na região central, no caso E forma-se um pequeno vórtice na região superior da placa do lado direito.

A influência da posição da placa no reator e consequentemente das zonas de recirculação ficam mais evidentes na Figura 5.27 com a presença do fluido traçador. Observa-

se uma tendência do fluido a ficar aprisionado nestas regiões resultando em um tempo residência elevado. Também é possível verificar o comportamento do fluido traçador no caso C (geometria base) analisada anteriormente. Pode-se constatar o comportamento verificado com pontos de monitoramento do traçador, onde o fluido atinge os pontos de monitoramento na sequência 2, 3 e 1, respectivamente.

Figura 5.26: Influência da geometria nas linhas de corrente e localização dos pontos de monitoramento: Reator *airlift* [25s escoamento].

Figura 5.27: Influência da geometria na distribuição do fluido traçador: Reator *airlift* [25s escoamento].

A partir da análise dos resultados anteriores torna-se evidente a influência do posicionamento e a altura da placa na hidrodinâmica do escoamento no reator. Isto é extremamente importante, pois a formação e a localização das zonas de recirculação têm influência direta no rendimento do reator.

Como mencionado anteriormente, além da influência no tempo de residência, os vórtices gerados no escoamento podem ser utilizados vantajosamente na movimentação das células de algas de áreas mal iluminadas para áreas bem iluminadas. A turbulência pode acelerar o crescimento de algas até um ponto ótimo. Vários estudos como ALIAS et al., 2004; CAMACHO et al., 2000; PAPOUTSAKIS, 1991, concluem que uma redução no desempenho de crescimento celular é devido à diminuição do tamanho dos vórtices (micro-vórtices em pequena escala) com o aumento da turbulência. Isto é porque, quando os comprimentos dos micro-vórtices na cultura são menores do que o comprimento das próprias células, ocorre dano celular.

A mistura de líquido e, portanto, a capacidade para movimentar o fluido é necessária para proporcionar condições ótimas de cultura. A fim de se conseguir uma mistura mais eficiente, reatores de algas devem ser concebidos com o intuito de movimentar periodicamente as células através de um gradiente de luz, por meio de um gradiente de baixa pressão, e também manter uma tensão de cisalhamento baixa ao longo de todo o sistema. As duas características determinantes mais importantes para isso são a geometria, que é definida pelos limites de líquidos, e as forças mecânicas aplicadas sobre o líquido nos locais de mistura. Assim, a tensão de cisalhamento é um fator importante a ser considerado no projeto de reatores, como discutido na Seção 4.4.3 (cálculo tensão cisalhamento). BRONNENMEIER; MARKL, 1982 e MICHELS et al., 2010 afirmam que um excesso de carga hidrodinâmica irá danificar microrganismos, devido à tensão de cisalhamento aplicada. Assim, como descrito anteriormente na metodologia a tensão de cisalhamento total foi considerado como sendo um parâmetro no processo de otimização. O outro parâmetro considerado é a média das variâncias das DTR ($MA\sigma^2$), como apresentado na Equação (5.4). A média das variâncias é adquirida através da DTR obtida em cinco pontos em linha igualmente espaçados sempre na parte direita da placa (região de descida) (Figura 5.26), sendo definida como:

$$MA\sigma^2 = \frac{\sum_{i=1}^{i=N} \sigma_i^2}{N} \quad (5.4)$$

A Tabela 5.16 apresenta os resultados dos parâmetros de otimização para as cinco geometrias analisadas acima. É possível observar a influência da geometria nos parâmetros de otimização. A geometria E apresenta a menor $MA\sigma^2$, como observado na Tabela 5.16, isto se deve ao fato de a placa estar muito próxima a parede do reator, ocasionando um escoamento laminar, como observado na Figura 5.26-E. A geometria D apresenta o mesmo comportamento (Figura 5.26-D). O comportamento de um escoamento laminar entre a placa e a parede do reator nas geometrias D e E também são constatadas com o baixo valor do desvio padrão entre as curvas DTR. A geometria A igualmente apresenta um baixo $MA\sigma^2$, no entanto, isto é explicado pelos baixos valores da variância das curvas DTR nos pontos 1 e 2. Como observado na Figura 5.28-A ocorrem uma grande discrepância entre as curvas. Isto pode ser explicado pela formação de uma grande zona de recirculação nesta região, como discutido acima e notado na Figura 5.26-A. Este comportamento é verificado com o desvio padrão entre as variâncias da DTR.

Em relação à tensão de cisalhamento tem-se um valor elevado para a geometria E, ocasionado pela proximidade da placa com a parede do reator e a sua altura. A altura da placa pode ser um fator que causa uma tensão de cisalhamento elevada, devido ao estreitamento do escoamento entre a interface e a placa. Este comportamento é constatado na Tabela 5.16, onde as geometrias que apresentam as maiores tensões de cisalhamento são as que possuem as alturas máximas da placa. Para o projeto ideal de FBRs é importante saber o nível máximo de tensão de cisalhamento que pode ser tolerada pelas microalgas (MICHELS et al., 2010; UGWU; AOYAGI; UCHIYAMA, 2008).

Tabela 5.16: Parâmetros analisados: Reator *airlift* [50s escoamento]

Geometria	$MA\sigma^2$ [s ²]	Desvio padrão σ^2 [s]	TCT ponderada [m ² /s ²]
A	34,85	9,54	3,16E-05
B	48,46	5,99	3,95E-05
C (Geometria base)	51,17	3,47	4,23E-05
D	48,17	0,28	3,88E-05
E	30,57	0,32	5,17E-05

Figura 5.28: Curvas DTR para as diferentes geometrias analisadas – Reator *airlift*.

Na Figura 5.28 também é possível analisar o tempo despendido pelo fluido traçador para atingir os pontos de monitoramento em cada geometria. A injeção do fluido traçador se inicia no tempo de 9 segundos. Portanto, a partir deste tempo é possível inferir o tempo que o fluido traçador leva para atingir os pontos de monitoramento. Como se observa na Figura 5.28, a geometria A proporciona uma variação grande no tempo de aquisição do traçador nos pontos (ponto 1 em 28,5s e ponto 5 em 19,6s), ocasionados pela zona de recirculação. Com o aumento da altura da placa ocorre a diminuição da influência deste vórtice, resultando em uma menor variação da distribuição do fluido no reator. No entanto, o tempo é elevado como observado na geometria B (ponto 4 em 21,7s). A geometria C possui o mesmo comportamento já discutido acima, para o processo de validação. Já os casos D e E devido a proximidade da placa com a parede do reator apresentam tempos superiores, se destacando a geometria E, com a partícula traçadora atingindo o ponto 5 em 25,8 segundos.

Os resultados anteriores sugerem que o tempo de circulação é um parâmetro muito importante para garantir a qualidade de mistura no reator, por exemplo, em um fotobiorreator este parâmetro indica o tempo de permanência da alga em uma região clara (presença de luz)

e escura. Se o tempo de permanência das algas em uma das fases excede o limite máximo de tempo seria ocasionada uma baixa eficiência no crescimento das algas, podendo resultar em alguns casos na morte das algas. A mistura adequada é, portanto, importante não somente para a transferência de massa, mas para uma boa distribuição de luz (NAUHA; ALOPAEUS, 2013).

5.2.6 Dominância de Pareto e geometrias otimizadas – Reator *airlift*

Após a validação qualitativa do modelo CFD e a definição dos parâmetros de otimização serão apresentados nesta seção os resultados da otimização multiobjetivo do reator *airlift*, realizada através do software PyCFD-O. As probabilidades aplicadas no algoritmo de otimização são apresentadas na Tabela 5.9. Os dados de entrada e os correspondentes resultados para cada geometria testada no PyCFD-O são mantidos em um banco de dados criado ao longo do processo de otimização. O usuário pode manipular esses dados, gerando gráficos e realizar análises avançadas em diferentes combinações dos parâmetros para decidir qual é o melhor projeto.

Quando os dois parâmetros de otimização são plotados um contra o outro, como mostra a Figura 5.29, uma clara linha *trade-off* pode ser identificada onde as melhores geometrias são localizadas. O desenho final deve ser um compromisso entre a $MA\sigma^2$ e a tensão de cisalhamento ponderada. Na Figura 5.29 também é possível observar a localização da geometria base frente às geometrias otimizadas no espaço de busca.

A Figura 5.30 apresenta o comparativo das linhas de corrente entre as diferentes configurações de geometrias otimizadas. As mesmas análises realizadas na seção anterior (Seção 5.2.5) podem ser observadas com as geometrias otimizadas. As geometrias com maiores distância da placa em relação à região de injeção de ar (dy) apresentam as menores variâncias da DTR como pode ser observado pelas curvas de DTR na Figura 5.31. No entanto, isto se deve ao fato de a placa estar muito próxima à parede do reator, ocasionando escoamento laminar. O fato de a placa estar muito próxima à parede do reator e a sua elevada altura H_z podem explicar a alta tensão de cisalhamento (Tabela 5.17) Outra característica deste tipo de geometria é o elevado tempo que o fluido traçador despende para chegar aos pontos de monitoramento (Figura 5.31-A).

Figura 5.29: Fronteira de soluções não dominadas: Reator *airlift* (dy/Hz).

A geometria $dy0.030/Hz0.40$ proporciona um baixo $MA\sigma^2$, no entanto, isto é ocasionado pelos baixos valores da variância das curvas DTR nos pontos 1 e 2, como discutido na Seção 5.2.5. A Figura 5.30-B e Figura 5.31-B ilustram a formação da zona de recirculação e as curvas DTR nos pontos de monitoramento. Este efeito da variação entre as curvas DTR é constatado pelo valor do desvio padrão da σ^2 entre as curvas apresentado na Tabela 5.17. Assim, a escolha do projeto otimizado deve levar em consideração a análise das curvas de DTR para que este efeito seja verificado. Isto ocorre devido ao tempo total de processamento do escoamento na análise da geometria. Devido ao elevado custo computacional envolvido, um tempo de simulação de 50 segundos de escoamento foi utilizado. Se o tempo de simulação do escoamento for elevado o decaimento das curvas DTR dos pontos 1 e 2 (Figura 5.31-B) se completariam implicando em um valor elevado da $MA\sigma^2$.

A geometria $dy0.060/Hz0.45$ apresenta a menor tensão de cisalhamento ponderada, como pode ser verificado na Tabela 5.17. A aplicação do processo de otimização demonstrou que as geometrias com o posicionamento da placa próximo à região central do reator e com uma altura não elevada apresentam as menores tensões de cisalhamento. Esta análise foi verificada anteriormente na Seção 5.2.5, o que indica uma concordância com o processo de otimização. Considerando a $MA\sigma^2$ a geometria $dy0.060/Hz0.45$ possui o maior valor dentre as geometrias otimizadas. Isto é reflexo da formação da zona de recirculação como observado na Figura 5.31-B.

Figura 5.30: Linhas de correntes das geometrias otimizadas e localização dos pontos de monitoramento – Reator *airlift*.

Tabela 5.17: Parâmetros de otimização das geometrias otimizadas[50s escoamento] – Reatora *airlift*

Geometria	$MA\sigma^2$ [s ²]	Desvio padrão σ^2 [s]	TCT ponderada [m ² /s ²]
0.120/0.51	30,57	0,32	5,17E-05
0.030/0.40	34,85	9,54	3,16E-05
0.054/0.46	36,72	6,06	2,30E-05
0.060/0.45	43,16	6,57	2,23E-05
Geometria base	51,17	3,47	4,23E-05

A Figura 5.32 e a Figura 5.33 apresentam as superfícies de respostas obtidas com as avaliações das geometrias no processo de otimização. Na Figura 5.32 podemos observar a influência dos parâmetros geométricos de otimização na $MA\sigma^2$. Os pontos de mínimos localizam-se em geometrias com os valores máximos dos parâmetros dy e Hz . Isto indica a formação do escoamento laminar entre a placa e a parede do reator, acarretando um elevado tempo de permanência do fluido na região de *riser* do reator. Deste modo a correta escolha da geometria ótima deve levar em consideração os gráficos das curvas da DTR nos diferentes pontos de monitoramento.

A Figura 5.33 apresenta a superfície de resposta em relação à tensão de cisalhamento ponderada. Pode-se observar a tendência dos pontos de mínimos localizarem-se em uma posição intermediária da placa em relação em relação à dy e Hz , ou seja, valores entre 0,05 e 0,06 metros para dy e 0,45 e 0,47 metros para Hz . De acordo com MICHELS et al., 2010, em seu estudo dos efeitos da tensão de cisalhamento em um tipo de microalga (*Chaetoceros muelleri*) o valor limiar no qual a viabilidade de algas foi influenciada negativamente estava entre 1,0 e 1,3 Pa. Considerando os valores da tensão de cisalhamento ponderada obtidos para os reatores

analisados e multiplicando-os pela densidade da água nota-se que estes valores ficam abaixo deste limite, indicando a viabilidade da aplicação do FBR analisado para este tipo de alga.

Figura 5.31: Curvas DTR das geometrias otimizadas: A) $dy = 0,120$ m, $H_z = 0,51$ m; B) $dy = 0,030$ m, $H_z = 0,40$ m; C) $dy = 0,054$ m, $H_z = 0,46$ m; D) $dy = 0,060$ m, $H_z = 0,45$ m;

Figura 5.32: Superfície de resposta $MA\sigma^2$ - Reator *airlift*

Figura 5.33: Superfície de resposta tensão de cisalhamento ponderada - Reator *airlift*.

Assim como realizado no processo de otimização do reator contínuo, variações consideráveis de geometrias foram testadas pelo procedimento de otimização, explorando diferentes combinações pertencentes ao espaço de possíveis soluções. É importante ressaltar que certamente essas variações de geometrias exploradas dificilmente seriam consideradas em uma abordagem manual de otimização. A otimização automática requer para este problema um tempo computacional considerável, devido às dimensões do reator (61, 5 litros) e ao fato da abordagem bifásica. Mesmo considerando as limitações impostas ao processo de otimização foi possível obter diferentes geometrias otimizadas com diferentes características. Considerando a geometria base (dy 0,08 m e H_z 0,51 m) e a geometria otimizada (dy 0,054 m e H_z 0,46 m) houve uma redução de 28% na variância do tempo de residência e uma diminuição em 46% na tensão de cisalhamento ponderada como pode ser verificado na Tabela 5.17.

CAPÍTULO 6 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho teve como objetivo geral o desenvolvimento de um software de código aberto que permita o acoplamento de modelos CFD a heurísticas de otimização multiobjetivos, a fim de possibilitar a construção de configurações otimizadas de biorreatores industriais, incluindo fotobiorreatores para o cultivo de microalgas. A motivação para o desenvolvimento deste trabalho baseou-se na necessidade de se buscar equipamentos otimizados voltados para a indústria sucro-energética. O processo de otimização multidisciplinar de projetos já é amplamente difundida em várias áreas da engenharia. No entanto, esta aplicação de otimização multiobjetivos voltada para o desenvolvimento de equipamentos de alto rendimento para o setor sucroalcooleiro é algo inovador.

Desse modo, um software aberto denominado PyCFD-O foi desenvolvido. A aplicação do PyCFD-O foi realizada em dois estudos de caso. O primeiro estudo foi realizado em um reator contínuo, onde a análise da influência dos parâmetros geométricos na dinâmica do escoamento pode ser estendida para o caso de um fermentador contínuo para a produção de etanol. O segundo caso voltou-se para a aplicação em um reator *airlift* de circulação interna do tipo placa, voltado para o desenvolvimento de um fotobiorreator otimizado. Até a presente data não se encontrou na literatura técnica e científica nenhum trabalho relacionado ao processo de otimização multiobjetivo dos parâmetros geométricos acoplado com análise da hidrodinâmica do escoamento via CFD em FBRs. O desenvolvimento de fotobiorreatores no contexto do trabalho está inserido na aplicação deste tipo de equipamento para o reaproveitamento de subprodutos gerados na produção de etanol, como por exemplo, a vinhaça.

O software desenvolvido PyCFD-O mostrou-se robusto e versátil. Foram testadas duas implementações de métodos de otimização multiobjetivos, geradores de malhas e modelos de simulação numérica de escoamento: monofásico e bifásico. As técnicas de paralelização utilizadas para o processamento mostraram-se eficientes. O fato de se trabalhar com um software aberto possibilitou a aplicação do processamento paralelo sem nenhum custo adicional de licenças. O processamento paralelo para o processo de otimização é essencial,

devido às características inerentes do problema de otimização de avaliar centenas de geometrias e o custo computacional envolvido nas simulações CFD de cada geometria. Destaca-se neste trabalho as dimensões dos reatores envolvidos nas simulações, 1 000 litros e 61,5 litros para o reator contínuo e o *airlift* respectivamente.

Para os processos de otimização de ambos os reatores foram propostas modificações nas suas geometrias que possuem influência direta no escoamento e, consequentemente, nas reações químicas envolvidas (caso fermentador) e crescimento de algas (caso fotobiorreator). Dois parâmetros do escoamento foram considerados como função objetivo: a TCT ponderada e a variância da DTR. As geometrias que favorecem as melhores condições para o escoamento foram propostas através de um algoritmo genético aplicado à otimização multiobjetivo (NSGA-II).

O método de otimização por algoritmos genéticos se adequa perfeitamente à metodologia de MDO. O AG baseado em não-dominância apresentou-se satisfatoriamente robusto e os resultados mostraram uma tendência a encontrar um conjunto de soluções ótimas globais, a chamada fronteira de Pareto, embora os estudos realizados não permitam afirmar que as soluções encontradas no final de cada otimização seja de fato tal fronteira. Uma melhor definição da fronteira de Pareto pode ser alcançada através do ajuste dos parâmetros de otimização (probabilidade de mutação e *crossover*, número de gerações e números de indivíduos na população). Contudo, os resultados demonstram que o uso deste tipo de AG é bastante adequado para problemas de otimização multiobjetivo. O esquema de otimização proposto conduziu a soluções otimizadas para ambos os estudos de caso (reator contínuo e *airlift*), obtendo parâmetros coerentes para os melhores indivíduos da população, com valores bastante distintos para as variáveis de projeto.

Variações consideráveis foram testadas pelo procedimento de otimização, explorando o espaço de parâmetros antes de identificar as melhores soluções. Para o reator contínuo ao final do procedimento de otimização multiobjeto foi possível obter uma fronteira não dominada com as melhores soluções encontradas no processo. Considerando as geometrias otimizadas e as geometrias base houve uma redução de até 47% na variância do tempo de residência e uma diminuição em 48% na TCT ponderada. Qualquer geometria na fronteira de Pareto pode ser utilizada para diferentes objetivos de projetos. Analisando-se as superfícies de resposta do caso do reator contínuo conclui-se que as geometrias com α entre 60 e 80 graus, juntamente com uma altura de saída (H_{out}) entre 1,2 e 1,3 metros apresentaram as melhores condições de projeto para os parâmetros de otimização avaliados. Um ângulo de entrada nesta faixa não causaria uma tensão de cisalhamento significativa na parede do reator. Uma maior

altura H_{out} ocasionaria uma permanência mais longa do fluido no reator, diminuindo-se assim a influência da tensão de cisalhamento no afunilamento da tubulação na saída do reator.

No reator *airlift* foram realizados estudos prévios visando-se obter uma maior compreensão da hidrodinâmica do escoamento e a validação dos dados simulados computacionalmente. Os resultados experimentais e simulados apresentaram uma boa concordância tanto qualitativamente quanto quantitativamente. Os resultados dos campos de velocidades obtidos com a técnica PIV e do tempo de mistura e circulação obtidos com a técnica pulso de pH permitiram validar a abordagem CFD satisfatoriamente. Foi possível verificar a influência da geometria na formação de zonas de recirculação e o tempo que a partícula traçadora despende nas regiões de *riser* e *downcomer* do fotobiorreator. A correta simulação desses parâmetros é um fator importante para futuras análises voltadas para o projeto de fotobiorreatores industriais. A utilização do modelo VOF para esta abordagem mostrou-se eficiente possibilitando a sua aplicação para futuros estudos.

A aplicação do PyCFD-O para a otimização multiobjetivo do reator *airlift* abordou a minimização da TCT ponderada e a variância da DTR. Como apresentado para o caso do reator contínuo foi possível obter uma fronteira não dominada com as geometrias otimizadas. Considerando-se a geometria base (dy 0,08 m e H_z 0,51 m) e a geometria otimizada (dy 0,054 m e H_z 0,46 m) houve uma redução de 28% na variância do tempo de residência e uma diminuição em 46% na tensão de cisalhamento ponderada.

Todas as adaptações e desenvolvimentos de códigos necessários nos *solvers* CFD aplicadas no OpenFOAM apresentaram resultados satisfatórios, demonstrando o potencial e adaptabilidade da utilização do pacote CFD OpenFOAM.

Assim sendo, este trabalho abre caminhos para novos estudos voltados para aprimorar equipamentos envolvidos na produção de biocombustíveis. Esta metodologia proporciona a possibilidade de aplicações diretas na indústria, como por exemplo, no desenvolvimento de projetos otimizados voltados para a indústria sucro-energética. Melhorias neste campo são um desafio para pesquisadores, o que torna o presente trabalho ainda mais motivador. Grande destaque merece ser dado à multidisciplinaridade do tópico, o qual enriquece o trabalho acadêmico.

O uso do software PyCFD-O pode ser estendido para a otimização multiobjetivo de outros tipos de equipamentos que envolvem a simulação computacional do escoamento, abrindo um grande leque de possíveis futuras aplicações.

6.1 Sugestões para trabalhos futuros

Devido à multidisciplinaridade do trabalho várias frentes de trabalhos futuros podem ser levantadas. A seguir, serão apresentadas algumas sugestões restringindo-se à continuidade dos trabalhos voltados ao desenvolvimento de biorreatores e fotobiorreatores:

- Aperfeiçoamento do software PyCFD-O, tal como implementação de uma interface gráfica;
- Utilização de técnicas de processamento paralelo em placas gráficas (GPGPU) para o processamento de rotinas dos modelos CFD (OpenFOAM);
- Estudo de novas técnicas de otimização multiobjetivos e aplicação de métodos híbridos de otimização (determinístico - estocástico);
- Implementação de um método de otimização com incertezas utilizando filtros Bayesianos, processo este aplicado à problemas de CFD transiente.
- Aprofundamento nos estudos de otimização para o processo de *scale-up* de reatores;
- Implementação de um modelo matemático que descreva o crescimento fototrófico das microalgas e acoplamento com o modelo CFD.
- Implementação de relação entre as taxas de crescimento de algas e os parâmetros estruturais do fotobiorreator;
- Busca de um fotobiorreator industrial otimizado para o problema do acoplamento bio-foto-fluidodinâmico;
- Verificação da utilização de metamodelos para redução do custo computacional envolvido nas simulações CFD;
- Desenvolvimento de metamodelos para a previsão dos parâmetros de otimização do modelo CFD;
- Utilização do software PyCFD-O no processo de otimização multiobjetivo aplicado ao projeto de reatores industriais de pré-tratamento (ESALQ-USP).

REFERÊNCIAS BIBLIOGRÁFICAS

- ABARCA, C.D.G. Inovações tecnológicas na agroindústria da cana-de-açúcar no Brasil. **COPPE/Universidade Federal do Rio de Janeiro, DEPRO/Universidade Federal de Ouro Preto**, 2005.
- AL-RIFFAI, P.; DIMARANAN, B.; LABORDE, D. **Global Trade and Environmental Impact Study of the EU Biofuels Mandate** International Food Policy Institute. 2010.
- ALIAS, C.B. et al. Influence of Power supply in the feasibility of *Phaeodactylum tricornutum* Cultures. **Biotechnology and Bioengineering**, v.87, n.6, p.723-733, 2004.
- ALOK, S.; IMMANUEL, G. Effect of Different Impellers and Baffles on Aerobic Stirred Tank Fermenter using Computational Fluid Dynamics. **J Bioprocess Biotech**, v.4, n.184, 2014.
- ANDRIETTA, M.D.G.S.; STECKELBERG, C.; ANDRIETTA, S.R. Bioetanol – Brasil, 30 anos na vanguarda. **MultiCiência**, 2006.
- AUBIN, J.; FLETCHER, D.F.; XUERE, C. Modeling turbulent flow in stirred tanks with CFD: the influence of the modeling approach, turbulence model and numerical scheme. **Experimental Thermal and Fluid Science**, v.28, n.5, p.431-445, 2004.
- BAHL, A. et al. Automated optimization of a reduced layer 5 pyramidal cell model based on experimental data. **Journal of Neuroscience Methods**, v.210, n.1, p.22-34, 2012.
- BANNARI, R. et al. Mass transfer and shear in an airlift bioreactor: Using a mathematical model to improve reactor design and performance. **Chemical Engineering Science**, v.66, p.2057-2067, 2011.
- BARRETT, R. et al. **Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods**. 2 ed. [Philadelphia]: SIAM, 1994.
- BARROCAL, V.M. et al. Production of biomass by *Spirulina maxima* using sugar beet vinasse in growth media. **N Biotechnol.**, v.27, n.6, p.851-856, 2010.
- BERBEROVIĆ, E. et al. Drop impact onto a liquid layer of finite thickness: Dynamics of the cavity evolution. **Physical Review E**, v.79, n.3, p.036306, 2009.
- BICHARA, A. **Tratamento de vinhaça por processos físico-químicos de precipitação química e flotação e sua utilização como meio de cultivo para microalga de potencial bioenergético, *Chlorella vulgaris***. 2014. 152 pp. (Dissertação (Mestrado)) - Universidade de São Paulo, São Carlos, 2014.
- BITOG, J.P. et al. Application of computational fluid dynamics for modeling and designing photobioreactors for microalgae production: A review. **Computers and Electronics in Agriculture**, v.76, n.2, p.131-147, 2011.

- BNDES. **Bioetanol de cana-de-açúcar : energia para o desenvolvimento sustentável. Rio de Janeiro: BNDES e CGEE. 2008.**
- BRACKBILL, J.U.; KOTHE, D.B.; ZEMACH, C. A continuum method for modeling surface tension. **J. Comput. Phys.**, v.100, p.335-354, 1992.
- BRENNAN, L.; OWENDE, P. Biofuels from microalgae—A review of technologies for production, processing, and extractions of biofuels and co-products. **Renewable and Sustainable Energy Reviews**, v.14, n.2, p.557-577, 2010.
- BRONNENMEIER, R.; MARKL, H. Hydrodynamic Stress Capacity of Microorganisms. **Biotechnology and Bioengineering**, p.553-578, 1982.
- BRUCATO, A. et al. Computer simulation of turbulent flow in baffled and unbaffled tanks stirred by radial impellers. In: *Computer Applications to Batch Processes*, 1990, Cengio, Italy. Cengio, Italy: 1990.
- BÜCHLER, P.M. The Attenuation of Underground Water Contamination in a Vinasse Clay-Lined Pond. **Water Science & Technology**, v.19, n.12, p.183-185, 1987.
- CAMACHO, G.F. et al. Effects of mechanical and hydrodynamic stress in agitated sparged cultures of *Porphyridium cruentum*. **Process Biochemistry**, v.35, p.1045-1050, 2000.
- CAMPANA, E.F. et al. Shape optimization in ship hydrodynamics using computational fluid dynamics. **Computer Methods in Applied Mechanics and Engineering**, v.196, n.1–3, p.634-651, 2006.
- CARVALHO, A.P.; MEIRELES, L.A.; MALCATA, X. Microalgal Reactors: A Review of Enclosed System Designs and Performances. **Biotechnol. Prog.**, v.22, p.1490-1506, 2006.
- CAZALBOU, J.B.; SPALART, P.R.; BRADSHAW, P. **On the Behavior of Two-Equation Models at the Edge of a Turbulent Region.** 1994.
- CERRI, M., O. **Hidrodinâmica e transferência de oxigênio em três biorreatores airlift de circulação interna geometricamente semelhantes.** 2005. 148 pp. (Doutorado) - UFSCar, São Carlos, 2005.
- CETESB. **Vinhaça – Critérios e procedimentos para aplicação no solo agrícola.** Disponível em: <<http://www.cetesb.sp.gov.br/servicos/normas---cetesb/43-normas-tecnicas---cetesb>>. Acesso em: 11/02/2015.
- CFD ONLINE **Near wall treatment in k-omega SST.** Disponível em: <<http://www.cfd-online.com/Forums/openfoam-solving/86855-near-wall-treatment-k-omega-sst.html>>. Acesso em: 04/02/2015.
- CHEN, Y. et al. Design Optimization of Scaffold Microstructures Using Wall Shear Stress Criterion Towards Regulated Flow-Induced Erosion. **Journal of Biomechanical Engineering**, v.133, 2011.
- CHISTI, Y. **Airlift bioreactors.** [London, UK]: Elsevier, 1989.

- CHOI, B.S. et al. Residence Time Distributions in a Stirred Tank: Comparison of CFD Predictions with Experiment. **Industrial & Engineering Chemistry Research**, v.43, n.20, p.6548-6556, 2004.
- CHOI, J.; KIM, K.; CHUNG, D. Numerical optimization for design optimization of automotive cooling fan. **SAE Technical Paper 970935**, 1997.
- COELLO, C.A.C.; LAMONT, G.B.; VELDHUIZEN, D.A.V. **Evolutionary Algorithms for Solving Multi-Objective Problems**. [New York]: Springer, 2007.
- CORTÉS, O.E.J. **BIOCOMBUSTÍVEIS A PARTIR DE MICROALGAS: MODELGEM E ANÁLISE DE FOTOBIOREACTORES** 2009. 110 pp. (Doutorado) - Universidade Federal da Bahia, Salvador, 2009.
- COURANT, R.; FRIEDRICHS, K.O.; LEWY, H. Uber die partiellen differenz-gleichungen der mathematischen Physik. **Mathematische Annalen**, v.100, p.32-74, 1928.
- DAMIÁN, S.M. **AN EXTENDED MIXTURE MODEL FOR THE SIMULTANEOUS TREATMENT OF SHORT AND LONG SCALE INTERFACES**. 2013. 231 pp. (Ph.D.) - UNIVERSIDAD NACIONAL DEL LITORAL, Santa Fe, 2013.
- DAVIDSON, K.M. et al. Using Computational Fluid Dynamics Software to Estimate Circulation Time Distributions in Bioreactors. **Biotechnol. Prog.**, v.19, p.1480-1486, 2003.
- DEB, K. **Multi-Objective Optimization using Evolutionary Algorithms**. London, 2001.
- DEB, K. et al. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. **IEEE Transactions on Evolutionary Computation**, v.6, p.182-197, 2002.
- DEL NINNO, M.P. **Investigation of turbulent multiphase flows in a flat panel photobioreactor and consequent effects on microalgae cultivation; using computational Fluid Dynamics (CFD) simulation and Particle Image Velocimetry (PIV)**. 2012. 51 pp. (MASTER OF SCIENCE) - Iowa State University Ames, Iowa 2012.
- DELNOIJ, E. et al. Dynamic simulation of dispersed gas-liquid two-phase flow using a discrete bubble model. **Chem. Eng. Sci**, v.52, n.9, p.1429-1458, 1997.
- DHOTRE, M.T.; JOSHI, J.B. Design of a gas distributor: Three-dimensional CFD simulation of a coupled system consisting of a gas chamber and a bubble column. **Chemical Engineering Journal**, v.125, n.3, p.149-163, 2007.
- DIAS, M. et al. Simulation of integrated first and second generation bioethanol production from sugarcane: comparison between different biomass pretreatment methods. **Journal of Industrial Microbiology & Biotechnology**, v.38, n.8, p.955-966, 2011.
- DORIA, E. et al. Isolation and characterization of a *Scenedesmus acutus* strain to be used for bioremediation of urban wastewater. **Journal of Applied Phycology**, v.24, n.3, p.375-383, 2012.

- EL-SAYED, M.; SUN, T.; BERRY, J. Shape optimization with computational fluid dynamics. **Advances in Engineering Software**, v.36, n.9, p.607-613, 2005.
- ESPAÑA-GAMBOA, E. et al. Vinasses: Characterization and treatments. **Waste Manag Res.**, v.29, n.12, p.1235-1250, 2011.
- ESTECO. **ESTECO**. Disponível em: <<http://www.esteco.com>>. Acesso em: 21/10/2012.
- FERZIGER, J.H.; PERIC, M. **Computational Methods for Fluid Dynamics**. 3rd. [Berlin]: Springer, 2001.
- FOGLER, H.S. **Elements of Chemical Reaction Engineering**. 4th Edition. [Prentice Hall], 2005.
- FONSECA, C.M.; FLEMING, P.J. Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. In: Proceedings of the 5th International Conference on Genetic Algorithms, 1993, San Mateo, USA. San Mateo, USA: Morgan Kaufmann Publishers Inc., 1993. p.416-423.
- FORTUNA, A.O. **Técnicas computacionais para dinâmicas dos fluidos: conceitos básicos e aplicações**. [São Paulo]: EDUSP, 2000.
- GHASEMI, N. et al. CFD simulation of solid–liquid flow in a two impinging streams cyclone reactor: Prediction of mean residence time and holdup of solid particles. **Chemical Engineering and Processing: Process Intensification**, v.49, n.12, p.1277-1283, 2010.
- GÓIS, E.R.C. **Otimização dos parâmetros geométricos de fermentadores contínuos aplicados na produção de bioetanol através de simulação computacional do escoamento**. 2012. 87 pp. (Doutorado) - Universidade de São Paulo, São Carlos, 2012.
- GOLDBERG, D.E. **Genetic Algorithms in Search, Optimization and Machine Learning**. Reading, MA, 1989.
- GOLUB, G.H.; VAN LOAN, C.F. **Matrix Computation**. 3.ed. [Baltimore]: Johns Hopkins University Press, 1996.
- GRANBIO. **GranBio**. Disponível em: <<http://www.granbio.com.br/>>. Acesso em: 06/11/2014.
- GRIMA, E.M. et al. Photobioreactors: light regime, mass transfer, and scaleup. **Journal of Biotechnology**, v.70, n.1, p.231-247, 1999.
- GRIS, L.R.S. **Produção da microalga Nannochloropsis oculata em fotobiorreator airlift**. 2011. 148 pp. (Dissertação (Mestrado)) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2011.
- GUERRERO, J. **Introductory OpenFOAM® Course: University of Genova**. Disponível em: <http://www.dicat.unige.it/guerrero/of2014b/supplement_turbulence.pdf>. Acesso em: 04/02/2015

- HARRIS, C.K.; ROEKAERTS, D.; F.J.J.ROSENDAL. Computational Fluid Dynamics for Chemical Reactor Engineering **Chemical Engineering Science**, v.51, n.10, p.1569-1594, 1996.
- HERTEL, T.W.; TYNER, W.E.; BIRUR, D.K. The Global Impacts of Biofuel Mandates **The Energy Journal**, v.31, n.1, p.75-100, 2010.
- HICKS, R.; VANDERPLAATS, G. Design of Low-Speed Airfoils by Numerical Optimization. **SAE Technical Paper 750524**, 1975.
- _____. . Airfoil section drag reduction at transonic speeds by numerical optimization. **SAE Technical Paper** n.760477, 1976.
- _____. . Application of numerical optimization to the design of supercritical airfoil without drag creep. **SAE Technical Paper 770440**, 1977.
- HIRT, C.W.; NICHOLS, B.D. Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries. **Journal of Computational Physics**, v.39, n.1, p.201-225, 1981.
- HOLLAND, J.H. **Adaptation in Natural and Artificial Systems**. [Ann Arbor]: University of Michigan Press, 1975.
- HORN, J.; NAFPLIOTIS, N.; GOLDBERG, D.E. A Niche Pareto Genetic Algorithm for Multiobjective Optimization. In: First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, 1994, 1994. p.82-87.
- HORVATH, A. et al. CFD simulation of bubble columns using the VOF Model: Comparison of commercial and open source solvers with experiment. **Chemical engineering transactions**, v.18, 2009.
- ISSA, R.I. Solution of the implicitly discretised fluid flow equations by operator-splitting. **Journal of Computational Physics**, v.62, n.1, p.40-65, 1986.
- JASAK, H. **Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows**. 1996. 394 pp. (Doctor of Philosophy) - University of London and Diploma of Imperial College, London, 1996.
- JAVANMARDIAN, M.; PALSSON, B.O. High-density photoautotrophic algal cultures: Design, construction, and operation of a novel photobioreactor system. **Biotechnology and Bioengineering**, v.38, n.10, p.1182-1189, 1991.
- KRESTA, S.M.; WOOD, P.E. Prediction of the three-dimensional turbulent flow in stirred tanks. **AIChE Journal**, v.37, n.3, p.448-460, 1991.
- KRISHNA, R.; VAN BATEN, J.M.; URSEANU, M.I. Three-phase Eulerian simulations of bubble column reactors operating in the churn-turbulent regime: a scale up strategy. **Chemical Engineering Science**, v.55, n.16, p.3275-3286, 2000.
- LEVENSPIEL, O. **Engenharia das reações químicas: cálculo de reatores**. [São Paulo]: Edgard Blücher, 2000.

- LUO, H.P.; AL-DAHMAN, M.H. Analyzing and modeling of photobioreactors by combining first principles of physiology and hydrodynamics. **Biotechnology and Bioengineering**, v.84, n.4, p.382-393, 2004.
- MACEDO, I.C. Situação atual e perspectivas do etanol. **Estudos avançados**, v.21, n.59, p.157-165, 2007.
- MÄKINEN, R.A.E.; PERIAUX, J.; TOIVANEN, J. Multidisciplinary shape optimization in aerodynamics and electromagnetics using genetic algorithms. **International Journal for Numerical Methods in Fluids**, v.30, n.2, p.149-159, 1999.
- MALAN, A.G.; LEWIS, R.W. On the development of high-performance C++ object-oriented code with application to an explicit edge-based fluid dynamics scheme. **Computers & Fluids**, v.33, n.10, p.1291-1304, 2004.
- MALISKA, C.R. **Transferência de Calor e Mecânica dos Fluidos Computacional**. [Rio de Janeiro]: Livros Técnicos e Científicos, 1995.
- MARQUES, S.S. et al. Growth of *Chlorella vulgaris* on sugarcane vinasse: the effect of anaerobic digestion pretreatment. **Appl Biochem Biotechnol.**, v.171, n.8, p.1933-1943, 2013.
- MARSCHALL, K.J.; MLECZKO, L. CFD modeling of an internally circulating fluidized-bed reactor. **Chemical Engineering Science**, v.54, p.2085-2093, 1999.
- MELVIN, R.G. et al. Recent progress in aerodynamic design optimization. **International Journal for Numerical Methods in Fluids**, v.30, n.2, p.205-216, 1999.
- MENTER, F.; ESCH, T. Elements of Industrial Heat Transfer Prediction. In: 16th Brazilian Congress of Mechanical Engineering (COBEM), 2001, Uberlândia. Uberlândia: 2001.
- MENTER, F.R. Influence of Freestream Values on $k-\omega$ Turbulence Model Predictions. **AIAA Journal**, v.30, n.6, p.1657-1659, 1992.
- MENTER, F.R.; KUNTZ, M.; LANGTRY, R. Ten Years of Industrial Experience with the SST Turbulence Model. In: 4th International Symposium on Turbulence, Heat and Mass Transfer, 2003, Antalya, Turkey. Antalya, Turkey: Begell House, Inc., 2003. p.625-632.
- MICHELE, V.; HEMPEL, D.C. Liquid flow and phase holdup—measurement and CFD modeling for two-and three-phase bubble columns. **Chemical Engineering Science**, v.57, n.11, p.1899-1908, 2002.
- MICHELS, M.H.A. et al. Effects of shear stress on the microalgae *Chaetoceros muelleri*. **Bioprocess Biosyst Eng** v.33, p.921-927, 2010.
- MIRÓN, A.S. et al. Comparative evaluation of compact photobioreactors for large-scale monoculture of microalgae. **Journal of Biotechnology**, v.70, p.249-270, 1999.
- MONTGOMERY, D.C. **Design of Experiments**. [New York]: John Wiley & Sons, Inc, 1997.

- MOUSAVI, S.M. et al. CFD simulation and optimization of effective parameters for biomass production in a horizontal tubular loop bioreactor. **Chemical Engineering and Processing**, v.49, p.1249-1258, 2010.
- MOUZA, K.A.; KAZAKIS, N.A.; PARAS, S.V. BUBBLE COLUMN REACTOR DESIGN USING A CFD CODE In: 1st International Conference "From Scientific Computing to Computational Engineering, 2004, Athenes. Athenes: IC-SCCE 2004.
- MUYL, F.; DUMAS, L.; HERBERT, V. Hybrid Method for Aerodynamic Shape Optimization in Automotive Industry. **Computers & Fluids**, v.33, n.849-858, 2004.
- NAUHA, E.K.; ALOPAEUS, V. Modeling method for combining fluid dynamics and algal growth in a bubble column photobioreactor. **Chemical Engineering Journal**, v.229, p.559-568, 2013.
- NAVARRO, A.R.; DEL C. SEPÚLVEDA, M.; RUBIO, M.C. Bio-concentration of vinasse from the alcoholic fermentation of sugar cane molasses. **Waste Management**, v.20, n.7, p.581-585, 2000.
- NOCEDAL, J.; WRIGHT, S.J. **Numerical optimization**. [New York]: Springer, 1999.
- OLIVEIRA, A.C. **Produção de biomassa de microalgas Scenedesmus sp. em efluente de bovinocultura biodigerido**. 2013. 82 pp. (Dissertação) - Universidade Federal do Paraná, Curitiba, 2013.
- OPENCFD. **OpenFOAM: The Open Source CFD Toolbox, User guide**. 2013.
- OSAKI, M.R. **Análise de um modelo de conversão de cana-de-açúcar em vetores energéticos através da integração de tecnologias 1G e 2G - simulação de Monte Carlo para otimização multiobjetivos**. 2014. 236 pp. (Doutorado) - Universidade de São Paulo, São Carlos, 2014.
- PAPANIKOLAOU, A. Holistic ship design optimization. **Computer-Aided Design**, v.42, n.11, p.1028-1044, 2010.
- PAPOUTSAKIS, E.T. Fluid-Mechanical damage of animal-cells in bioreactors. **Trends in Biotechnology and Bioengineering**, v.9, n.12, p.427-437, 1991.
- PARAVIEW. **Paraview**. Disponível em: <<http://www.paraview.org/>>. Acesso em: 10/10/2012.
- PATWARDHAN, A.W. et al. Optimization of gas-liquid reactor using computational fluid dynamics. **Chemical Engineering Science**, v.60, n.11, p.3081-3089, 2005.
- POLLARD, D.J.; A. P. ISON, A.P.; SHAMLOU, P.A. Reactor heterogeneity with Saccharopolyspora erythraea airlift fermentations. **Biotechnology and Bioengineering** v.68, n.2, p.160-172, 1998.
- POLONI, C. et al. Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for a complex design problem in fluid dynamics. **Comput. Methods Appl. Mech. Engrg.**, v.186, p.403-420, 2000.

- PORTO, L.D.M. **Modelagem de processo industrial de fermentação alcoólica contínua com reatores de mistura ligados em série**. 2005. 122 pp. (Doutorado) - Universidade Estadual de Campinas, Campinas, 2005.
- POSTEN, C.; WALTER, C. **Microalgal Biotechnology: Potential and Production**. [Berlin]: Hubert & Co. / De Gruyter, 2012.
- PRESS, W.H. et al. **Numerical Recipes in C: The Art of Science Computing**. [New York]: Cambridge University Press, 1992.
- PRUVOST, J.; POTTIER, L.; LEGRAND, J. Numerical investigation of hydrodynamic and mixing conditions in a torus photobioreactor. **Chemical Engineering Science**, v.61, n.14, p.4476-4489, 2006.
- QUAGLIARELLA, D. et al. **Genetic Algorithms and Evolution Strategy in Engineering and Computer Science: Recent Advances and Industrial Applications**. John Wiley Sons Ltd (Import), 1998.
- RAMIREZ, N.N.V.; FARENZENA, M.; TRIERWEILER, J.O. Growth of microalgae *Scenedesmus* sp in ethanol vinasse. **Brazilian Archives of Biology and Technology**, v.57, n.5, p.630-635, 2014.
- RANADE, V.V.; JOSHI, J.B. Flow generated by a disc turbine: Part II Mathematical modelling and comparison with experimental data. **Trans. IChemE**, v.68A, p.34-50, 1990.
- RAO, S.S. **Engineering Optimisation – Theory And Practice**. 1996.
- RICHARDSON, L.F. The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam. **Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character**, v.210, n.459-470, p.307-357, 1911.
- RUSCHE, H. **Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions**. 2002. (Doctor of Philosophy) - Imperial College of Science University of London, London, 2002.
- _____. **Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions**. 2002. 343 pp. (Ph.D. Thesis) - Imperial College of Science, London, 2002.
- SCHAFFER, J.D. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: Proceedings of the 1st International Conference on Genetic Algorithms, 1985, L. Erlbaum Associates Inc., 1985. p.93-100.
- SCHMIDELL, W. et al. **Biotechnologia Industrial**. [São Paulo]: Edgard Blücher Ltda, 2001.
- SCHOBERL, J. NETGEN: An advancing front 2D/3D-mesh generator based on abstract rules. **Computing and Visualization in Science**, v.1, p.41-52, 1997.

- SEABRA, J.E.A. **Avaliação técnico-econômico de opções para o aproveitamento integral da biomassa de cana no Brasil**. 2008. 274 pp. (Doutorado) - Universidade de Campinas, Campinas, 2008.
- SEARCHINGER, T. et al. Use of U.S. Croplands for Biofuels Increases Greenhouse Gases Through Emissions from Land Use Change **Science**, v.319, n.5867, p.1238-1240, 2008.
- SEO, I.-H. et al. Numerical investigation of a bubble-column photo-bioreactor design for microalgae cultivation. **Biosystems Engineering**, v.113, n.3, p.229-241, 2012.
- SHAO, T. et al. Simulation of Solid Suspension in a Stirred Tank Using CFD-DEM Coupled Approach. **FLUID DYNAMICS AND TRANSPORT PHENOMENA**, v.21, n.10, p.1069-1081, 2013.
- SHAO, X. et al. Reactor scale up for biological conversion of cellulosic biomass to ethanol. **Bioprocess Biosyst Eng**, v.33, p.485-493, 2010.
- SHI, X. Design optimization of insulation usage and space conditioning load using energy simulation and genetic algorithm. **Energy**, v.36, n.3, p.1659-1667, 2011.
- SILVA, C.G. **De Sol a Sol: a energia no século XXI**. [São Paulo]: Oficina de textos, 2010.
- SILVA, L.F.L.R. **Desenvolvimento de metodologias para simulação de escoamentos polidispersos usando código livre**. 2008. 262 pp. (Doutorado) - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2008.
- SIMONNET, M. et al. CFD simulation of the flow field in a bubble column reactor: Importance of the drag force formulation to describe regime transitions. **Chemical Engineering and Processing: Process Intensification**, v.47, n.9–10, p.1726-1737, 2008.
- SOMAN, A.; SHASTRI, Y. Optimization of novel photobioreactor design using computational fluid dynamics. **Applied Energy**, v.140, p.246–255, 2015.
- SOUZA, J.F.A. et al. UMA REVISÃO SOBRE A TURBULÊNCIA E SUA MODELAGEM. **Revista Brasileira de Geofísica**, v.29, n.1, p.21-41, 2011.
- SPOGIS, N.; NUNHEZ, J.R. Design of a high-efficiency hydrofoil through the use of computational fluid dynamics and multiobjective optimization. **AIChE Journal**, v.55, n.7, p.1723-1735, 2009.
- SRIVIVAS, N.; DEB, K. Multiobjective optimization using nondominated sorting in genetic algorithms. **Evol. Comput.**, v.2, n.3, p.221-248, 1995.
- STROUSTRUP, B. **The C++ Programming Language: Special Edition**. 3rd. [Murray Hill]: Addison-Wesley, 2000.
- SU, Z. et al. Study on the destabilization mixing in the flat plate photobioreactor by means of CFD. **Biomass and Bioenergy**, v.34, n.12, p.1879-1884, 2010.

- SZAFRAN, R.G.; KMIEC, A. Application of CFD Modelling Technique in Engineering Calculations of Three-Phase Flow Hydrodynamics in a Jet-Loop Reactor. **INTERNATIONAL JOURNAL OF CHEMICAL REACTOR ENGINEERING**, v.2, p.A30, 2004.
- TANNEHILL, J.C.; ANDERSON, D.A.; PLETCHER, R.H. **Computational fluid mechanics and heat transfer**. [New York]: Taylor & Francis, 1997.
- TENNEKES, H.; LUMLEY, J.L. **A first course in turbulence**. [Massachusetts]: The MIT Press, 1972.
- THÉVENIN, D.; JANIGA, G. **Optimization and Computational Fluid Dynamics**. [Berlin]: Springer, 2008.
- THIELICKE, W.; STAMHUIS, E.J. PIVlab – Towards User-friendly, Affordable and Accurate Digital Particle Image Velocimetry in MATLAB. **Journal of open research software**, v.2, n.30, p.1-10, 2014.
- TICONA, W.G.C. **Algoritmos evolutivos multi-objetivo para a reconstrução de árvores filogenéticas**. 2003. 134 pp. (Tese de Doutorado) - Universidade de São Paulo, São Carlos, 2003.
- TREDICI, M.R. **Mass Production of Microalgae: Photobioreactors**. [Oxford]: Blackwell Publishing Ltd, 2003.
- TURNER, M.J. et al. Stiffness and deflection analysis of complex structures. **Aeronautical Science Journal**, v.23, p.805-23, 1956.
- UBBINK, O. **Numerical prediction of two fluid systems with sharp interfaces**. 1997. 138 pp. (Ph.D. Thesis) - Imperial College of Science, London, 1997.
- UBBINK, O.; ISSA, R.I. A Method for Capturing Sharp Fluid Interfaces on Arbitrary Meshes. **Journal of Computational Physics**, v.153, n.1, p.26-50, 1999.
- UGWU, C.U.; AOYAGI, H.; UCHIYAMA, H. Photobioreactors for mass cultivation of algae. **Bioresource Technology**, v.99, n.10, p.4021-4028, 2008.
- UM, B.H.; HANLEY, T.R. A comparison of simple rheological parameters and simulation data for *Zymomonas mobilis* fermentation broths with high substrate loading in a 3-L bioreactor. **Applied Biochemical Biotechnology**, v.145, p.29-38, 2008.
- UNICA. **Revisão de Safra 2012/13**. Disponível em: <<http://www.unicadata.com.br/listagem.php?idMn=39>>. Acesso em: 28/09/2012.
- VAN BATEN, J.M.; ELLENBERGER, J.; KRISHNA, R. Hydrodynamics of internal air-lift reactors: experiments versus CFD simulations. **Chemical Engineering and Processing: Process Intensification**, v.42, n.10, p.733-742, 2003.
- VASCONCELOS, J.M. et al. Effect of contaminants on mass transfer coefficients in bubble column and airlift contactors. **Chem. Eng. Sc.**, v.58, p.1431-1440, 2003.

- VERSTEEG, H.K.; MALALASEKERA. **An Introduction to Computational Fluid Dynamics**. 2ª. [Essex]: Longman Scientific & Technical, 2007.
- VERSTEEG, H.K.; MALALASEKERA, W. **An introduction to computational fluid dynamics: The finite volume method**. [Essex]: Longman Scientific & Technical, 1995.
- WANG, L.-L.; TAO, Y.; MAO, X.-Z. A novel flat plate algal bioreactor with horizontal baffles: Structural optimization and cultivation performance. **Bioresource Technology**, v.164, p.20-27, 2014.
- WELLER, H.G. et al. A tensorial approach to continuum mechanics using object-oriented techniques. **Computers in Physics**, v.12, n.6, p.620-631, 1998.
- WILLIAMS, K.A.; SAINI, S.; WICK, T.M. Computational Fluid Dynamics Modeling of Steady-State Momentum and Mass Transport in a Bioreactor for Cartilage Tissue Engineering. **Biotechnology Progress**, v.18, n.5, p.951-963, 2002.
- WU, B. Advances in the use of CFD to characterize, design and optimize bioenergy systems. **Computers and Electronics in Agriculture**, n.0, 2012.
- XIA, J.-Y. et al. Computational investigation of fluid dynamics in a recently developed centrifugal impeller bioreactor. **Biochemical Engineering Journal**, v.38, p.406-413, 2008.
- XUE, Q.; HEINDEL, T.J.; FOX, R.O. A CFD model for biomass fast pyrolysis in fluidized-bed reactors. **Chemical Engineering Science**, v.66, n.11, p.2440-2452, 2011.
- YANG, D. **C++ and Object-Oriented Numeric Computing for Scientists and Engineers**. [New York]: Springer, 2001.
- YU, G. et al. A novel method using CFD to optimize the inner structure parameters of flat photobioreactors. **Journal of Applied Phycology**, v.21, n.6, p.719-727, 2009.
- ZHANG, H.L.; BAEYENS, J.; TAN, T.W. The bubble-induced mixing in starch-to-ethanol fermenters. **Chemical Engineering Research and Design**, v.90, n.12, p.2122-2128, 2012a.
- _____. . Mixing phenomena in a large-scale fermenter of starch to bio-ethanol. **Energy**, v.48, n.1, p.380-391, 2012b.
- ZHANG, K.; KURANO, N.; MIYACHI, S. Optimized aeration by carbon dioxide gas for microalgal production and mass transfer characterization in a vertical flat-plate photobioreactor. **Bioprocess and Biosystems Engineering**, v.25, n.2, p.97-101, 2002.
- ZITZLER, E.; DEB, K.; THIELE, L. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. **Evol. Comput.**, v.8, n.2, p.173-195, 2000.
- ZITZLER, E.; LAUMANN, M.; THIELE, L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. In: Communication Networks Lab (TIK), 2001, Zurich, Switzerland. Zurich, Switzerland: 2001.

ZITZLER, E.; THIELE, L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. **Evolutionary Computation, IEEE Transactions on**, v.3, n.4, p.257-271, 1999.

APÊNDICE A Exemplo de otimização multiobjetivo

A fim de ilustrar o correto funcionamento dos métodos de otimização aplicados no software PyCFD-O foi utilizado um exemplo teste como explicado a seguir. O exemplo pode ser analisado como o processo de otimização de um fermentador contínuo. O problema consiste em maximizar o volume do reator com a menor área externa. A geometria do reator esta restrita a forma cilíndrica devido a facilidade de construção. O processo de aumento do volume permite processar grandes volumes de sacarose, no entanto, quanto menor é a área externa desse reator menor será sua perda de calor com o meio externo.

A Figura 6.1 ilustra a geometria cilíndrica do problema com os parâmetros geométricos de otimização: H altura do reator e D diâmetro do reator. As variáveis de otimização do projeto são volume e a área e são calculadas de acordo com as equações (A.1) e (A.2).

Figura 6.1: Representação da geometria do problema.

$$\text{Área} = 2\pi \frac{D^2}{4} + \pi DH \quad (\text{A.1})$$

$$\text{Volume} = H\pi \frac{D^2}{4} \quad (\text{A.2})$$

O problema de otimização consiste em obter o maior volume com a menor área, respeitando uma variação do diâmetro e altura previamente definida. Deste modo, pode-se definir o problema de otimização multiobjetivo de acordo com a Equação (A.3):

$$\begin{aligned}
\text{Min: } f1 &= 1/\text{Volume} \\
\text{Min: } f2 &= \text{Área} \\
\text{Sujeito: } &\begin{cases} 0,2 \leq D \leq 1,0 \\ 0,2 \leq H \leq 1,0 \end{cases}
\end{aligned} \tag{A.3}$$

Na Tabela 6.1 os parâmetros dos métodos de otimização adotados em ambos os métodos para avaliar o problema proposto. Os métodos avaliam as possíveis soluções do problema e ao final da execução é possível representar graficamente as soluções não dominadas, como representado na Figura 6.2.

Tabela 6.1: Parâmetros dos algoritmos de otimização.

Parâmetro	Valor
Tamanho da população, N	80
Gerações, G	20
Probabilidade de mutação	0,5
Probabilidade de crossover	0,5
Distribuição de mutação	20
Distribuição de crossover	20

A Figura 6.2 ilustra as fronteiras de Pareto para o problema obtidas com a aplicação do método EMOO e NSGA-II respectivamente. A fronteira de Pareto é formada por valores das funções objetivos $f1$ e $f2$ correspondentes a cada solução no espaço de busca. Um dos objetivos principais de algoritmos que solucionam problemas com múltiplos objetivos é encontrar soluções o mais próximo possível da fronteira de Pareto, e ainda buscarem uma maior diversidade possível no espaço de soluções.

Observa-se na Figura 6.2 a fronteira de Pareto obtidas com a aplicação dos dois métodos analisados no problema de otimização. A partir das fronteiras de Pareto pode-se então empreender uma análise no sentido de auxiliar a tomada de decisão e de identificar as soluções ótimas. Nota-se em ambos os casos que as geometrias com os maiores volumes possuem uma área maior (caso $H=1$ e $D=1$), no entanto, para uma geometria com a menor área o volume também é reduzido ($D=0,25$ e $H=0,23$). Para ambos os métodos a curva de Pareto apresentou o mesmo comportamento com uma boa diversidade de pontos.

Figura 6.2: Fronteiras de Pareto e localização das menores F.A. do problema de otimização: A) EMOO; B) NSGA-II.

A fim de analisar uma geometria otimizada em ambos os métodos, foi definido uma função de análise (F.A.) definida de acordo com a Equação (A.4). Nesta equação ambas as funções objetivos são elevadas ao quadrado e somadas definindo um peso para cada geometria. Dentre os pesos obtidos com F.A. em cada geometria busca-se a que possui a menor. Com isso obtemos a geometria que apresente a melhor combinação entre área e volume para o problema proposto.

$$F.A. = f1^2 + f2^2 \quad (A.4)$$

A Tabela 6.2 apresenta as geometrias com as menores F.A. em ambos os métodos. Podemos observar que os valores os da F.A. para ambos os métodos estão próximas (15,32 e 15,30) para os correspondentes diâmetro e altura 0,78 e 0,77 para o EMOO e 0,82 e 0,74 para o NSGA-II. Um comparativo entre as geometrias pode ser observado na Figura 6.3 e suas localizações nas fronteiras são ilustradas na Figura 6.2.

Tabela 6.2: Geometrias com as menores F.A.

Método	D	H	f1	f2	F.A.
EMOO	0,78	0,77	2,67	2,87	15,36
NSGA-II	0,82	0,74	2,51	3,00	15,30

Figura 6.3: Comparativo entre as geometrias com as menores F.A. obtidas com os métodos EMOO e NSGA-II.

Com a aplicação dos métodos neste exemplo foi possível verificar o correto funcionamento da implementação do código em python utilizado posteriormente no software PyCFD-O.

APÊNDICE B Código fonte solver pisoScalarFoamAdjTime

Nesta seção é apresentado o solver pisoScalarFoamAdjTime, o qual consiste em uma adaptação do solver pisoFoam, para ajustar o intervalo de tempo durante a simulação de acordo com o número máximo de Courant. Neste solver também foi acoplada a solução de uma equação de transporte necessário para as análises com a partícula traçadora.

Algoritmo 4: createFields.H

```

1  Info<< "Reading transportProperties\n" << endl;
2
3  IOdictionary transportProperties
4  (
5      IOobject
6      (
7          "transportProperties",
8          runTime.constant(),
9          mesh,
10         IOobject::MUST_READ_IF_MODIFIED,
11         IOobject::NO_WRITE
12     )
13 );
14
15 // --- Scalar Diffusion
16 dimensionedScalar DC
17 (
18     transportProperties.lookup("DC")
19 );
20
21 // --- Pressure
22 Info<< "Reading field p\n" << endl;
23 volScalarField p
24 (
25     IOobject
26     (
27         "p",
28         runTime.timeName(),
29         mesh,
30         IOobject::MUST_READ,
31         IOobject::AUTO_WRITE
32     ),
33     mesh
34 );
35
36 // --- Velocity
37 Info<< "Reading field U\n" << endl;
38 volVectorField U
39 (
40     IOobject
41     (

```

```

42     "U",
43     runTime.timeName(),
44     mesh,
45     IOobject::MUST_READ,
46     IOobject::AUTO_WRITE
47 ),
48     mesh
49 );
50
51 // --- Scalar
52 Info<< "Reading field C\n" << endl;
53 volScalarField C
54 (
55     IOobject
56     (
57         "C",
58         runTime.timeName(),
59         mesh,
60         IOobject::MUST_READ,
61         IOobject::AUTO_WRITE
62     ),
63     mesh
64 );
65
66 #include "createPhi.H"
67
68
69 label pRefCell = 0;
70 scalar pRefValue = 0.0;
71 setRefCell(p, mesh.solutionDict().subDict("PISO"), pRefCell, pRefValue);
72
73 singlePhaseTransportModel laminarTransport(U, phi);
74
75 autoPtr<incompressible::turbulenceModel> turbulence
76 (
77     incompressible::turbulenceModel::New(U, phi, laminarTransport)
78 );

```

Algoritmo 5: pisoScalarFoamAdjTime.C

```

1  /*-----*
2  =====|
3  \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox
4  \ \ / O p e r a t i o n |
5  \ \ / A n d | Copyright (C) 2004-2010 OpenCFD Ltd.
6  \ \ M a n i p u l a t i o n |
7  -----*
8  License
9  This file is part of OpenFOAM.
10
11  OpenFOAM is free software: you can redistribute it and/or modify it
12  under the terms of the GNU General Public License as published by
13  the Free Software Foundation, either version 3 of the License, or
14  (at your option) any later version.
15
16  OpenFOAM is distributed in the hope that it will be useful, but WITHOUT
17  ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
18  FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
19  for more details.

```

```

20
21 You should have received a copy of the GNU General Public License
22 along with OpenFOAM. If not, see <http://www.gnu.org/licenses/>.
23
24 Application
25   pisoScalarFoamAdjTime
26
27 Description
28   Transient solver for incompressible flow (AdjustTimeStep).
29   + Scalar Transport
30   Turbulence modelling is generic, i.e. laminar, RAS or LES may be selected.
31 *
32 * Modifications made by: Jonas L. Ansoni
33
34 \*-----*/
35
36 #include "fvCFD.H"
37 #include "singlePhaseTransportModel.H"
38 #include "turbulenceModel.H"
39
40 // *****
41
42 int main(int argc, char *argv[])
43 {
44   #include "setRootCase.H"
45
46   #include "createTime.H"
47   #include "createMesh.H"
48   #include "createFields.H"
49   #include "initContinuityErrs.H"
50   #include "readTimeControls.H"
51
52   // *****
53
54   Info<< "\nStarting time loop\n" << endl;
55
56   while (runTime.loop())
57   {
58     Info<< "Time = " << runTime.timeName() << nl << endl;
59
60     #include "readPISOControls.H"
61     #include "CourantNo.H"
62     #include "setDeltaT.H"
63
64     // Pressure-velocity PISO corrector
65     {
66       // Momentum predictor
67
68       fvVectorMatrix UEqn
69       (
70         fvm::ddt(U)
71         + fvm::div(phi, U)
72         + turbulence->divDevReff(U)
73       );
74
75       UEqn.relax();
76
77       if (momentumPredictor)
78       {
79         solve(UEqn == -fvc::grad(p));

```

```

80     }
81
82     // --- PISO loop
83
84     for (int corr=0; corr<nCorr; corr++)
85     {
86         volScalarField rAU(1.0/UEqn.A());
87
88         U = rAU*UEqn.H();
89         phi = (fvc::interpolate(U) & mesh.Sf())
90             + fvc::ddtPhiCorr(rAU, U, phi);
91
92         adjustPhi(phi, U, p);
93
94         // Non-orthogonal pressure corrector loop
95         for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
96         {
97             // Pressure corrector
98
99             fvScalarMatrix pEqn
100             (
101                 fvm::laplacian(rAU, p) == fvc::div(phi)
102             );
103
104             pEqn.setReference(pRefCell, pRefValue);
105
106             if
107             (
108                 corr == nCorr-1
109                 && nonOrth == nNonOrthCorr
110             )
111             {
112                 pEqn.solve(mesh.solver("pFinal"));
113             }
114             else
115             {
116                 pEqn.solve();
117             }
118
119             if (nonOrth == nNonOrthCorr)
120             {
121                 phi -= pEqn.flux();
122             }
123         }
124
125         #include "continuityErrs.H"
126
127         U -= rAU*fvc::grad(p);
128         U.correctBoundaryConditions();
129     }
130 }
131
132 turbulence->correct();
133
134 // --- Scalar Transport
135 Info<< "nCalculating scalar transport\n" << endl;
136     C.storePrevIter();
137     for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
138     {
139         volScalarField DCEff = DC+turbulence->nut()/0.7;

```

```
140         fvScalarMatrix TEqn
141         (
142             fvm::ddt(C) + fvm::div(phi, C) -
143             fvm::laplacian(DCEff, C)
144         );
145         TEqn.relax();
146         TEqn.solve();
147     }
148
149     runTime.write();
150
151     Info<< "ExecutionTime = " << runTime.elapsedCpuTime() << " s"
152     << " ClockTime = " << runTime.elapsedClockTime() << " s"
153     << nl << endl;
154 }
155
156 Info<< "End\n" << endl;
157
158 return 0;
159 }
```


APÊNDICE C Código fonte solver interDyScalarFoam

```

1 /*-----*\
2 ===== |
3 \ \ / F ield | OpenFOAM: The Open Source CFD Toolbox
4 \ \ / O peration |
5 \ \ / A nd | Copyright (C) 2004-2011 OpenCFD Ltd.
6 \ \ M anipulation |
7 -----\
8 License
9 This file is part of OpenFOAM.
10
11 OpenFOAM is free software: you can redistribute it and/or modify it
12 under the terms of the GNU General Public License as published by
13 the Free Software Foundation, either version 3 of the License, or
14 (at your option) any later version.
15
16 OpenFOAM is distributed in the hope that it will be useful, but WITHOUT
17 ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
18 FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
19 for more details.
20
21 You should have received a copy of the GNU General Public License
22 along with OpenFOAM. If not, see <http://www.gnu.org/licenses/>.
23
24 Application
25 interDyScalarFoam
26
27 Description
28 Solver for 2 incompressible, isothermal immiscible fluids using a VOF
29 (volume of fluid) phase-fraction based interface capturing approach,
30 with optional mesh motion and mesh topology changes including adaptive
31 re-meshing.
32
33 + Scalar Transport
34
35 /*-----*/
36
37 #include "fvCFD.H"
38 #include "dynamicFvMesh.H"
39 #include "MULES.H"
40 #include "subCycle.H"
41 #include "interfaceProperties.H"
42 #include "twoPhaseMixture.H"
43 #include "turbulenceModel.H"
44 #include "pimpleControl.H"
45
46 // *****
47
48 int main(int argc, char *argv[])
49 {
50     #include "setRootCase.H"
51     #include "createTime.H"
52     #include "createDynamicFvMesh.H"
53     #include "initContinuityErrs.H"
54     #include "createFields.H"

```

```

55 #include "readTimeControls.H"
56
57 pimpleControl pimple(mesh);
58
59 #include "correctPhi.H"
60 #include "CourantNo.H"
61 #include "setInitialDeltaT.H"
62
63 // * * * * *
64 Info<< "\nStarting time loop\n" << endl;
65
66 while (runTime.run())
67 {
68     #include "readControls.H"
69     #include "alphaCourantNo.H"
70     #include "CourantNo.H"
71
72     // Make the fluxes absolute
73     fvc::makeAbsolute(phi, U);
74
75     #include "setDeltaT.H"
76
77     runTime++;
78
79     Info<< "Time = " << runTime.timeName() << nl << endl;
80
81     scalar timeBeforeMeshUpdate = runTime.elapsedCpuTime();
82
83     // Do any mesh changes
84     mesh.update();
85
86     if (mesh.changing())
87     {
88         Info<< "Execution time for mesh.update() = "
89             << runTime.elapsedCpuTime() - timeBeforeMeshUpdate
90             << " s" << endl;
91         gh = g & mesh.C();
92         ghf = g & mesh.Cf();
93     }
94     if (mesh.changing() && correctPhi)
95     {
96         #include "correctPhi.H"
97     }
98     // Make the fluxes relative to the mesh motion
99     fvc::makeRelative(phi, U);
100    if (mesh.changing() && checkMeshCourantNo)
101    {
102        #include "meshCourantNo.H"
103    }
104    twoPhaseProperties.correct();
105
106    #include "alphaEqnSubCycle.H"
107
108    // --- Pressure-velocity PIMPLE corrector loop
109    for (pimple.start(); pimple.loop(); pimple++)
110    {
111        #include "UEqn.H"
112
113        // --- PISO loop
114        for (int corr=0; corr<pimple.nCorr(); corr++)

```

```

115     {
116         #include "pEqn.H"
117     }
118     if (pimple.turbCorr())
119     {
120         turbulence->correct();
121     }
122 }
123 // --- Scalar Transport
124 Info<< "\nCalculating scalar transport\n" << endl;
125 C.storePrevIter();
126 for (int corr=0; corr<pimple.nCorr(); corr++)
127 {
128     volScalarField DCEff = DC+turbulence->nut()/0.7;
129     fvScalarMatrix TEqn
130     (
131         fvm::ddt(C) + fvm::div(phi, C) - fvm::laplacian(DCEff, C)
132         //fvm::ddt(C) + fvm::div(phi, C)
133     );
134     TEqn.relax();
135     TEqn.solve();
136 }
137 runTime.write();
138 Info<< "ExecutionTime = " << runTime.elapsedCpuTime() << " s"
139     << " ClockTime = " << runTime.elapsedClockTime() << " s"
140     << nl << endl;
141 }
142 Info<< "End\n" << endl;
143 return 0;
144 }
145 // *****

```