

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

GUSTAVO NAOKI SHIBATTA KAGESAWA

**SIMULAÇÃO EM PYTHON DA OPERAÇÃO DO ECONOMIZADOR DE UMA
CALDEIRA PILOTO DE LEITO FLUIDIZADO CIRCULANTE**

PONTA GROSSA

2022

GUSTAVO NAOKI SHIBATTA KAGESAWA

**SIMULAÇÃO EM PYTHON DA OPERAÇÃO DO ECONOMIZADOR DE UMA
CALDEIRA PILOTO DE LEITO FLUIDIZADO CIRCULANTE**

**SIMULATION IN PYTHON OF THE ECONOMIZER'S OPERATION OF A LAB-
SCALE CIRCULATING FLUIDIZED BED BOILER**

Trabalho de conclusão de curso de graduação
apresentada como requisito para obtenção do título de
Bacharel em Engenharia Mecânica da Universidade
Tecnológica Federal do Paraná (UTFPR).
Orientador(a): Prof. Dr. Jhon Jairo Ramirez Behainne

PONTA GROSSA

2022



[4.0 Internacional](https://creativecommons.org/licenses/by-nc/4.0/)

Esta licença permite remixe, adaptação e criação a partir do trabalho, para fins não comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

GUSTAVO NAOKI SHIBATTA KAGESAWA

**SIMULAÇÃO EM PYTHON DA OPERAÇÃO DO ECONOMIZADOR DE UMA
CALDEIRA PILOTO DE LEITO FLUIDIZADO CIRCULANTE**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do título de
Bacharel em Engenharia Mecânica da Universidade
Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 13 junho de 2022

Jhon Jairo Ramirez Behainne
Doutorado
Universidade Tecnológica Federal do Paraná

Gerson Henrique dos Santos
Doutorado
Universidade Tecnológica Federal do Paraná

Thiago Antonini Alves
Doutorado
Universidade Tecnológica Federal do Paraná

PONTA GROSSA

2022

AGRADECIMENTOS

Aos meus pais e irmã, Roberto Toyofumi Kagesawa, Silvia Tiaki Shibatta Kagesawa e Thaís Shibatta Kagesawa, respectivamente, por sempre me proporcionarem apoio, cuidado e amor incondicional por todo o meu trajeto, a minha eterna gratidão.

A toda a minha família que sempre zelaram por mim.

Ao meu orientador, Prof. Dr. Jhon Jairo Ramirez Behainne, por toda a orientação e apoio durante o desenvolvimento desse trabalho. E, por ser um excelente profissional que busca sempre o sucesso no desenvolvimento de seus alunos.

A todos os professores que algum dia foram meus instrutores e contribuíram com a minha formação humana e profissional.

Aos meus amigos que sempre deram apoio, conselhos e companhia ao longo da minha jornada.

E a todas as outras pessoas que, em algum momento da minha vida, contribuíram com aprendizados e momentos valiosos.

RESUMO

Caldeiras de leito fluidizado circulante (LFC) são amplamente utilizadas para a geração de energia elétrica, devido à ampla variedade de combustíveis que podem usar e pela excelente eficiência de combustão. Os economizadores são trocadores de calor utilizados nessas caldeiras para realizar o pré-aquecimento da água alimentada ao tubo separador líquido-vapor. Esse dispositivo permite a troca térmica entre os gases de exaustão da caldeira e a água, até essa última atingir um valor apropriado abaixo do ponto de saturação na pressão de operação. Neste trabalho, um economizador de tubos concêntricos é analisado com auxílio de programação em linguagem computacional Python. A modelagem matemática foi realizada para seis condições específicas de operação de um economizador acoplado ao gerador de vapor piloto de leito fluidizado circulante, com o fim de determinar o efeito produzido sobre a vazão mássica da água pré-aquecida e a temperatura do gás de exaustão. Os resultados das simulações mostraram que a temperatura de entrada do gás de exaustão foi o parâmetro de maior influência sobre a sua temperatura de saída, sendo sempre superior a 150°C para evitar a condensação do vapor de água. Por outro lado, a pressão do líquido alimentado ao economizador foi a variável que promoveu as maiores mudanças na vazão mássica da água pré-aquecida que poderá ser útil para o controle futuro do equipamento instalado em escala de laboratório.

Palavras-chave: Caldeira de leito fluidizado circulante. Economizador. Trocador de calor de tubo concêntrico. Simulação em Python.

ABSTRACT

Circulating fluidized bed (CFB) boilers are widely applied for electric power generation, due to the wide variety of fuel that can be used and the excellent combustion efficiency. The economizers are heat exchangers used in those type of boilers to preheat of the water fed into the liquid-steam separator tube. This device allows the heat exchange between exhaust gases and the water, up to the last one achieve an appropriate value below the saturation temperature on the operational pressure. In this thesis, a concentric tube economizer is studied with the aid of programming in Python language. Mathematical modeling was performed for six specific conditions operating conditions of an economizer coupled to a lab-scale circulating fluidized bed boiler steam generator, in order to determine the effect produced on the mass flow of preheated water and the exhaust gas outlet temperature. The simulation results showed that exhaust gas inlet temperature was the parameter with the greatest influence on its outlet temperature, always being above 150 °C to avoid condensation of water vapor. On the other hand, the pressure of the liquid fed to the economizer was the variable that promoted the greatest changes in the mass flow of the water, which could be useful for further control of equipment installed on a laboratory scale.

Keywords: Circulating fluidized bed boiler. Economizer. Concentric tube heat exchanger. Simulation in Python.

LISTA DE FIGURAS

Figura 1 – Padrões de fluidização em sistemas gás-sólido	18
Figura 2 - Arranjo geral de uma típica caldeira de LFC	20
Figura 3 - Trocador de calor de tubos concêntricos em escoamento paralelo (a) e contracorrente (b)	22
Figura 4 - Vista frontal com alguns equipamentos identificados da caldeira de LFC do Laboratório de Processos Termoquímicos da UTFPR - PG	25
Figura 5 – Detalhamento do economizador da caldeira LFC.....	26
Figura 6 – Volume de controle para análise do economizador.....	26
Figura 7 - Região anular entre tubos concêntricos.....	33
Figura 8 – Etapas para o cálculo da taxa de transferência de calor (Equação (3))	34
Figura 9 - Fluxograma das fases de simulação.....	38
Figura 10 - Fluxograma de criação de gráficos a partir dos resultados de simulação.....	38

LISTA DE GRÁFICOS

Gráfico 1 - Índice TIOBE para Python.....	15
Gráfico 2 - Temperatura de saída do gás de exaustão em função da temperatura de entrada e pressão absoluta da água.....	39
Gráfico 3 - Vazão da água em função da temperatura de entrada e pressão absoluta da água	40
Gráfico 4 - Temperatura de saída de gás de exaustão em função da temperatura de entrada da água e da temperatura de entrada do gás de exaustão.....	41
Gráfico 5 - Vazão mássica da água em função da temperatura de entrada da água e da temperatura de entrada do gás de exaustão	42
Gráfico 6 - Temperatura de saída do gás de exaustão em função da temperatura de entrada do gás de exaustão e da pressão absoluta da água ...	44
Gráfico 7 - Vazão mássica da água em função da temperatura de entrada do gás de exaustão e da pressão absoluta da água	45
Gráfico 8 - Temperatura de saída do gás de exaustão em função da temperatura de entrada do gás de exaustão e da temperatura de entrada da água.....	46
Gráfico 9 - Vazão mássica da água em função da temperatura de entrada do gás de exaustão e da temperatura de entrada da água	47
Gráfico 10 - Temperatura de saída do gás de exaustão em função da pressão absoluta da água e da temperatura de entrada da água.....	48
Gráfico 11 - Vazão mássica da água em função da pressão absoluta da água e da temperatura de entrada da água.....	49
Gráfico 12 - Temperatura de saída do gás de exaustão em função da pressão absoluta da água e da temperatura de entrada do gás de exaustão	50

Gráfico 13 - Vazão mássica de água em função da pressão absoluta da água e da temperatura de entrada do gás de exaustão	50
Gráfico 14 - Temperatura de saturação da água em função da pressão.....	51

LISTA DE QUADROS

Quadro 1 - Classificação de aplicações de leitos fluidizados de acordo com o mecanismo predominante	19
Quadro 2 - Equipamentos para cada região da caldeira de LFC.....	20
Quadro 3 - Combinação de variáveis para simulação	37

LISTA DE TABELAS

Tabela 1 - Condições padrão de operação do economizador	27
Tabela 2 - Fatores de incrustação representativos	30
Tabela 3 - Coeficientes de influência para o escoamento laminar plenamente desenvolvido em uma região anular circular com fluxo térmico uniforme mantido nas duas superfícies	33
Tabela 4 - Variáveis principais e secundárias	36

LISTA DE ABREVIATURAS E SIGLAS

LFC	Leito Fluidizado Circulante
NO _x	Óxidos de Nitrogênio
SO ₂	Dióxido de Enxofre
MLDT	Média Logarítmica das Diferenças de Temperaturas
AISI	<i>American Iron and Steel Institute</i>

LISTA DE SÍMBOLOS

Letras Latinas

A	Área superficial de troca térmica	$[m^2]$
A_e	Área das superfícies em contato com o escoamento externo	$[m^2]$
A_i	Área das superfícies em contato com o escoamento interno	$[m^2]$
A_{tr}	Área da seção transversal do duto de escoamento	$[m^2]$
$c_{p,g}$	Calor específico do gás de exaustão	$[J/(kg \cdot K)]$
$c_{p,w}$	Calor específico da água	$[J/(kg \cdot K)]$
D	Diâmetro da seção transversal do tubo de escoamento	$[m]$
d_e	Diâmetro externo do tubo que separa os escoamentos	$[m]$
D_h	Diâmetro hidráulico	$[m]$
D_i	Diâmetro interno do tubo externo do economizador	$[m]$
d_i	Diâmetro interno do tubo que separa os escoamentos	$[m]$
$E_{\%}$	Erro percentual	$[\%]$
f	Fator de atrito de Darcy	$[-]$
h	Coefficiente convectivo	$[W/(m^2 \cdot K)]$
h_e	Coefficiente convectivo do escoamento externo	$[W/(m^2 \cdot K)]$
h_i	Coefficiente convectivo do escoamento interno	$[W/(m^2 \cdot K)]$
k	Condutividade térmica	$[W/(m \cdot K)]$
k_{mat}	Condutividade térmica do material	$[W/(m \cdot K)]$
L	Comprimento	$[m]$
$MLDT$	Média logarítmica das diferenças de temperatura	$[K]$
\dot{m}	Vazão mássica	$[kg/s]$
\dot{m}_g	Vazão mássica do gás de exaustão	$[kg/s]$
\dot{m}_w	Vazão mássica da água	$[kg/s]$
Nu_D	Número de Nusselt	$[-]$
$\overline{Nu_D}$	Número de Nusselt médio	$[-]$
Nu_e	Número de Nusselt para escoamento no diâmetro externo da região anular	$[-]$
Nu_i	Número de Nusselt para escoamento no diâmetro interno da região anular	$[-]$
Nu_{ee}	Número de Nusselt relacionado a coeficiente de influência no diâmetro externo da região anular	$[-]$
Nu_{ii}	Número de Nusselt relacionado a coeficiente de influência no diâmetro interno da região anular	$[-]$
P	Perímetro	$[m]$
Pr	Número de Prandtl	$[-]$
Q	Taxa de transferência de calor	$[W]$
q_e''	Fluxo de calor na superfície externa	$[W/m^2]$

q_i''	Fluxo de calor na superfície interna	$[W/m^2]$
Re_D	Número de Reynolds	$[-]$
$T_{f,entra}$	Temperatura do fluido frio na entrada	$[K]$
$T_{f,sai}$	Temperatura do fluido frio na saída	$[K]$
T_m	Temperatura média do fluido	$[K]$
$T_{q,entra}$	Temperatura do fluido quente na entrada	$[K]$
$T_{q,sai}$	Temperatura do fluido quente na saída	$[K]$
U	Coefficiente global de transferência de calor	$[W/(m^2 \cdot K)]$
u_m	Velocidade média do fluido	$[m/s]$
x	Valor de entrada	$[-]$
x_0	Primeiro valor dependente	$[-]$
x_1	Segundo valor dependente	$[-]$
y	Valor de saída	$[-]$
y_0	Primeiro valor independente	$[-]$
y_1	Segundo valor independente	$[-]$

Letras Gregas

ρ	Densidade	$[kg/m^3]$
Δ	Gradiente	$[-]$
μ	Viscosidade dinâmica (ou absoluta)	$[N \cdot (s/m^2)]$
θ_e''	Coefficiente de influência no diâmetro externo da região anular	$[-]$
θ_i''	Coefficiente de influência no diâmetro interno da região anular	$[-]$

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Contextualização	14
1.2	Justificativa	15
1.3	Objetivos	16
1.3.1	Objetivo geral	16
1.3.2	Objetivos específicos	16
2	REVISÃO BIBLIOGRÁFICA	17
2.1	Fluidização	17
2.2	Caldeira de leito fluidizado circulante	20
2.2.1	Economizadores de Caldeiras de LFC	21
2.3	Trocador de calor de tubos concêntricos	21
2.4	Python	22
2.4.1	Bibliotecas Python	23
3	METODOLOGIA	25
3.1	Descrição do economizador	25
3.2	Modelagem operacional do economizador	27
3.3	Desenvolvimento do algoritmo em Python	35
4	RESULTADOS E DISCUSSÃO	39
4.1	Efeitos da Variação da temperatura de entrada e pressão absoluta da água	39
4.2	Efeitos da variação da temperatura de entrada da água e da temperatura de entrada do gás de exaustão	41
4.3	Efeitos da variação da temperatura de entrada do gás de exaustão e pressão absoluta da água	43
4.4	Efeitos da variação da temperatura de entrada do gás de exaustão e temperatura de entrada da água	45
4.5	Efeitos da Variação da pressão absoluta da água e temperatura de entrada da água	47
4.6	Efeitos da variação da pressão absoluta da água e temperatura de entrada do gás de exaustão	49
4.7	Correlações matemáticas	51
5	CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS	52
	REFERÊNCIAS	53
	APÊNDICE A - Código em Python completo para as simulações	55

APÊNDICE B - Código em Python para criação dos gráficos	75
ANEXO A - Tabela de propriedades termofísicas da água.....	85
ANEXO B - Tabela de propriedades termofísicas do ar	89
ANEXO C - Tabela de saturação da água	91

1 INTRODUÇÃO

1.1 Contextualização

Nos últimos anos, tem-se discutido muito sobre a crise energética mundial, principalmente quando se trata da utilização de petróleo e seus derivados como combustível para a geração de calor e outras formas de energia. Muitas alternativas tecnológicas foram criadas e desenvolvidas ao longo dos anos visando o melhor uso dos combustíveis, como por exemplo, caldeiras de leito fluidizado circulante (LFC), que apresentam alto rendimento térmico e possibilitam a utilização de outros tipos de combustíveis, principalmente de origem renovável (BASU, 2006).

As fornalhas das caldeiras de LFC operam no regime de fluidização rápida, em que um padrão de escoamento característico, conhecido como núcleo-anular, promove excelente transferência de calor e de massa para a conversão eficiente do combustível (DAVIDSON, 2000). As altas velocidades de fluidização, próprias desse regime fluidodinâmico, fazem com que uma parte das partículas do leito sejam arrastadas para fora da zona de combustão e retornadas novamente à região de queima, mediante o uso de um circuito de transporte de sólidos que mantém a estabilidade operacional (SINGH; BRINK; HUPA, 2013).

Dentre os trocadores de calor mais importantes instalados nas caldeiras de LFC estão os economizadores, que utilizam o calor residual transferido pelos gases de exaustão com a finalidade de realizar o pré-aquecimento da água que será alimentada ao evaporador. A utilização desse componente auxilia para a obtenção de um maior rendimento térmico do sistema, uma vez que a água alcançará o evaporador com uma temperatura mais próxima à da saturação. Desta maneira, precisa-se de menor quantidade de combustível na fornalha para a obtenção de vapor, contribuindo também para que o gás de exaustão deixe o sistema a uma temperatura menor (BASU, 2006).

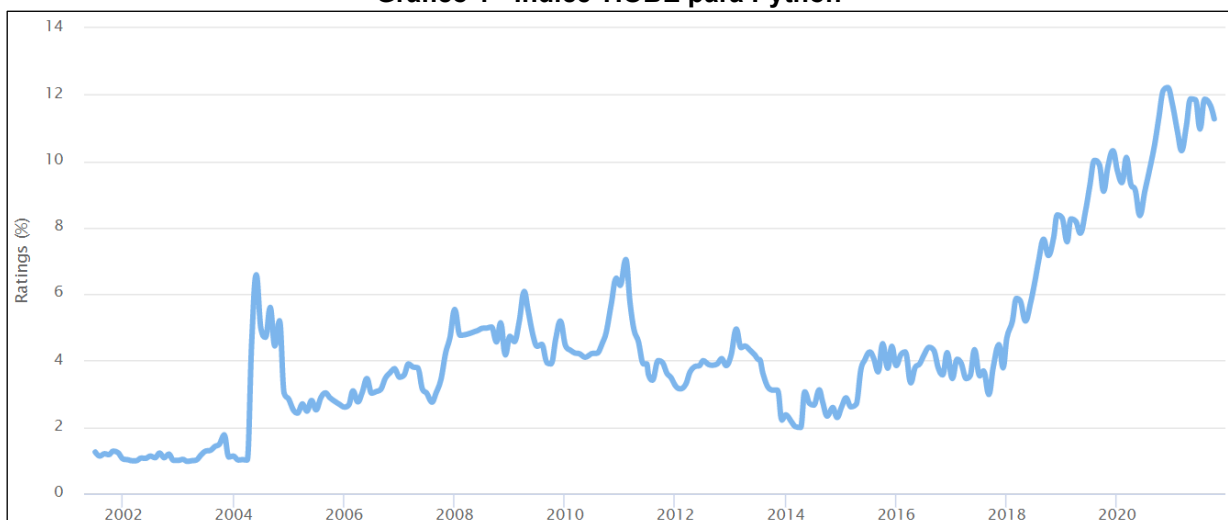
A análise operacional de uma caldeira de LFC, e de equipamentos em geral, pode ser realizada mediante testes experimentais e/ou simulações computacionais. No caso da aplicação de modelos computacionais, uma grande variedade de

ferramentas, envolvendo *softwares* e linguagens de programação, está disponível para realizar os estudos, dentre elas, a linguagem Python.

A linguagem Python foi criada em 1989 por Guido van Rossum. Seu lançamento oficial ocorreu em 1991, e surgiu como uma alternativa à linguagem ABC. As suas principais características são a sintaxe simples e a modalidade de código aberto, isto é, de domínio público (NOSRATI, 2011).

A popularidade do Python vem crescendo consideravelmente desde 2015. Segundo o Índice TIOBE (Gráfico 1), o Python foi a linguagem de programação mais utilizada no mundo em 2021, ultrapassando as linguagens C, Java e C++. Esse índice determina a nota para cada linguagem através de suas utilizações pela comunidade.

Gráfico 1 - Índice TIOBE para Python



Fonte: Adaptado de TIOBE (2021)

Outra importante característica da linguagem Python é o seu grande acervo de bibliotecas para as mais diversas aplicações. Neste trabalho, serão utilizadas as bibliotecas NumPy, Pandas e Matplotlib da versão 3.9.5 do Python para simular as condições operacionais de um economizador acoplado a uma caldeira de LFC de laboratório.

1.2 Justificativa

A simulação proposta nesse trabalho de conclusão de curso pretende analisar diferentes faixas de operação de um economizador instalado na linha de exaustão de

uma caldeira piloto de LFC, com o fim de nortear a realização de futuros testes experimentais. Esse equipamento está em fase de montagem no Laboratório de Processos Termoquímicos da UTFPR/Campus Ponta Grossa.

A linguagem Python foi escolhida devido a três motivos principais: o primeiro deles é pela linguagem de programação ser gratuita e *open-source*. O segundo motivo se deve ao fato dessa linguagem possuir um grande acervo de bibliotecas, que auxiliam na criação e desenvolvimento de diferentes tipos de projetos e plataformas. O terceiro se deve à característica de sintaxe mais simples e intuitiva, facilitando a sua aprendizagem e utilização. Ainda, acrescenta-se o incentivo à aplicação de uma linguagem de programação em constante crescimento no Brasil e no mundo.

1.3 Objetivos

Nesta seção são apresentados o objetivo geral e objetivos específicos do trabalho desenvolvido.

1.3.1 Objetivo geral

Esse trabalho tem, como objetivo geral, a realização de simulações computacionais da operação de um economizador acoplado a uma caldeira piloto de leito fluidizado circulante, utilizando a linguagem de programação Python.

1.3.2 Objetivos específicos

Os objetivos específicos desse trabalho são:

- Implementar um modelo matemático em linguagem Python que consiga simular a operação do economizador da caldeira piloto de LFC, envolvendo o uso das bibliotecas NumPy, Pandas e Matplotlib.
- Realizar simulações computacionais do economizador sob diversas condições operacionais, a fim de verificar as faixas adequadas para o seu funcionamento com base na demanda do gerador de vapor instalado em laboratório.

2 REVISÃO BIBLIOGRÁFICA

Esta seção aborda tópicos fundamentais relacionados à temática do presente trabalho. Sendo assim, são apresentados os conceitos de fluidização e de leito fluidizado circulante, a descrição das caldeiras de LFC e de seu evaporador, as características do trocador de calor de tubo concêntrico e a linguagem Python com algumas das suas bibliotecas.

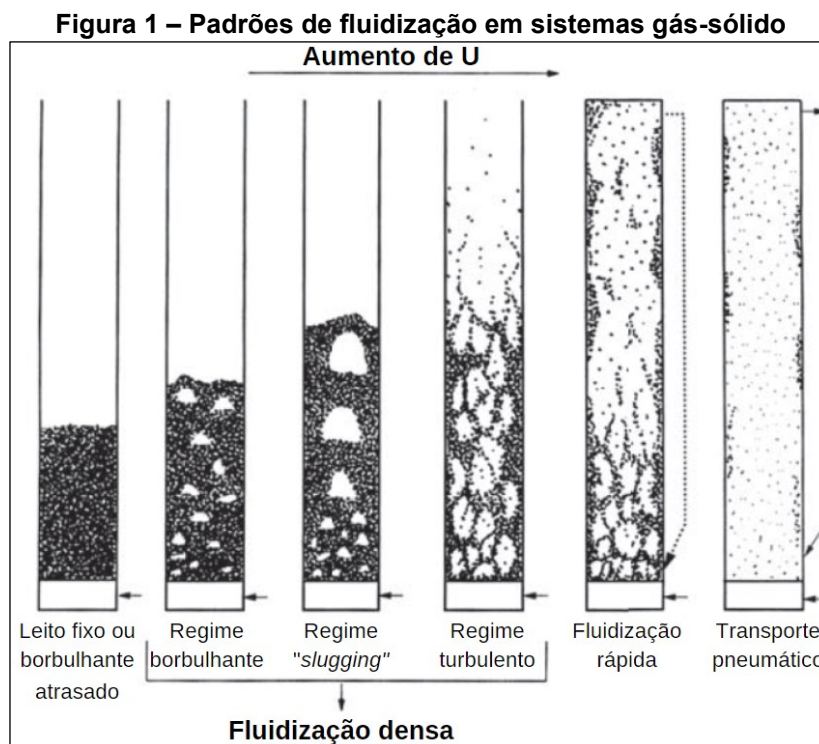
2.1 Fluidização

O fenômeno da fluidização ocorre quando um gás ou líquido, alimentado em sentido ascendente dentro de um determinado volume, atravessa um leito de partículas sólidas e finas a uma velocidade que promove a suspensão das mesmas, resultando em uma mistura com características semelhantes às de um fluido. Essa suspensão, conhecida como leito fluidizado, deve-se ao equilíbrio das forças da gravidade e de coesão das partículas com a força de arrasto causada pelo escoamento do fluido atravessando o leito (KUNII; LEVENSPIEL, 1991, p. 1; GRACE; BI; ELLIS, 2020, p. 1).

A fluidização é utilizada numa grande variedade de operações industriais, sendo possível empregar diferentes tipos e tamanhos de partículas sólidas, dimensões de diâmetro e profundidade de leito, velocidades e tipos de fluido, dentre outros parâmetros.

Para logr -la,   preciso envolver um m nimo de componentes, sendo esses, um vaso ou recipiente, um distribuidor de g s, as part culas s lidas e um impulsor de g s de fluidiza o ou m quina de fluxo (GELDART, 1986, p. 2-3),

Dependendo da intensidade da velocidade vertical ascendente (U) do fluido, aplicada sobre o leito de part culas, surgem diferentes regimes ou padr es de fluidiza o. Esses padr es s o ilustrados na Figura 1.



Fonte: Adaptado de Grace, Bi & Ellis (2020, p. 1)

Este trabalho concentra seu interesse no regime de fluidização rápida, que representa o principal padrão dos sistemas de LFC. Esse regime de fluidização é atingido quando a velocidade vertical do fluido ultrapassa a velocidade terminal do agregado de partículas, causando o arrasto parcial de partículas para fora do recipiente. Para manter a operação nesse regime, faz-se necessário incluir um sistema de coleta e de retorno das partículas para a coluna de fluidização (KUNII; LEVENSPIEL, 1991, p. 195; GRACE; BI; ELLIS, 2020, p. 239).

De acordo com Basu (2006) e Grace, Bi e Ellis (2020), em relação aos leitos fixos, as fornalhas de leito fluidizado possuem as seguintes características:

- Temperatura uniforme no leito;
- Flexibilidade de combustíveis que podem ser utilizados;
- Queda de pressão relativamente baixa;
- Excelentes coeficientes de transferência de calor;
- Alta eficiência de combustão;
- Habilidade de adicionar e remover partículas constantemente, facilitando a regeneração catalítica e operação contínua;
- Aplicável em grandes escalas;
- Baixa emissão de NO_x e possibilidade de absorção in-situ de SO_2 .

Porém, ressalta-se que as vantagens citadas anteriormente devem sobrepor as seguintes desvantagens (GRACE; BI; ELLIS, 2020, p. 4):

- Grande dispersão de sólidos, ou seja, muitos sólidos acabam não permanecendo tempo suficiente na câmara de reação;
- Limitações nas propriedades físicas das partículas sólidas que podem ser aplicadas no processo;
- As partículas e a câmara podem sofrer forte abrasão devido às colisões e ao atrito com as superfícies internas;
- Maior complexidade de projeto, operação e modelagem do processo.

As aplicações industriais das técnicas de fluidização foram iniciadas em meados da década de 1920 com o processo de gaseificação de carvão. A partir desses anos, foram desenvolvidos muitos estudos sobre aplicações dos leitos fluidizados (GELDART, 1986). O Quadro 1 mostra algumas das aplicações dos leitos fluidizados em termos do seu mecanismo dominante.

Quadro 1 - Classificação de aplicações de leitos fluidizados de acordo com o mecanismo predominante

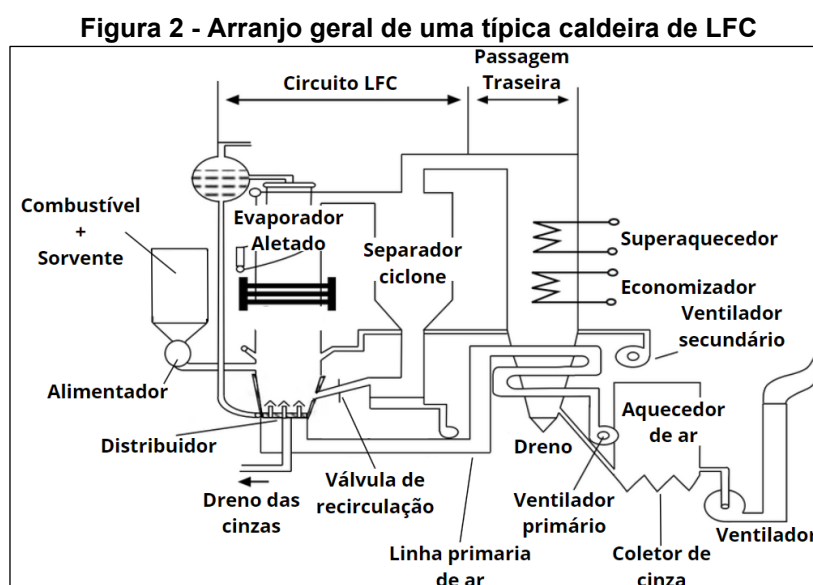
Processos industriais					
Físicos			Químicos		
Mecanismos dominantes	Transferência de calor e/ou massa entre gás/partículas	Transferência de calor e massa entre partículas/partículas ou partículas/superfícies	Transferência de calor entre leito/superfície	Reações gás/gás com sólidos agindo como catalizadores ou dissipadores de calor	Reações gás/sólidos no qual os sólidos são transformados
Aplicações	Secagem de sólidos	Revestimento plástico de partículas	Tratamento térmico de fibras têxteis, fios, borrachas, vidro, componentes metálicos	Quebra catalítica	Combustão e gaseificação de carvão e biomassa
	Absorção de solventes	Revestimento de fármacos	Banhos com temperatura constante	Manufatura de acrilonitrila, anídrico ftálico e polietileno	Queima de sulfatos de zinco e níquel
	Refrigeração de agregados de fertilizante	Granulação			Incineração de rejeitos sólidos e líquidos
	Congelamento de alimentos	Mistura de sólidos			Produção de tetracloro de titânio
		Filtragem de poeira			Reações de catalise
		Decomposição de calcário			

Fonte: Adaptado de Geldart (1986)

2.2 Caldeira de leito fluidizado circulante

As caldeiras de LFC ganharam importância nos últimos 40 anos, especialmente no mercado global de geração de energia elétrica, muito por conta das vantagens listadas na seção anterior. Os problemas apresentados pelas primeiras gerações foram resolvidos e caldeiras de LFC são consideradas atualmente uma tecnologia madura em unidades a pressões atmosféricas (BASU, 2006, p. 253).

Uma caldeira de LFC (Figura 2) pode ser dividida em duas grandes seções, sendo elas, o circuito de LFC e a região de passagem traseira dos gases (*back pass*).



Fonte: Adaptado de Basu (2006)

Para cada uma das regiões, tem-se usualmente os equipamentos resumidos no Quadro 2 (BASU, 2006, p. 253):

Quadro 2 - Equipamentos para cada região da caldeira de LFC

Circuito LFC	Região traseira
Tubos de evaporação e fornalha	Superaquecedor
Separador de gás/sólido (Ciclone)	Reaquecedor
Sistema de recirculação de sólidos (<i>loop-seal</i>)	Economizador
Trocador de calor externo (opcional)	Aquecedor de ar

Fonte: Adaptado de Basu (2006)

2.2.1 Economizadores de Caldeiras de LFC

Economizadores são trocadores de calor que têm o objetivo de pré-aquecer a água que será alimentada aos evaporadores, melhorando a eficiência do sistema de geração de vapor. A água é forçada através do economizador e realiza uma troca térmica com os gases de exaustão da caldeira. Normalmente, a temperatura da água na saída do economizador fica perto de 30°C abaixo da temperatura de saturação. Com isso, garante-se uma boa circulação da água no sistema. Há casos de algumas caldeiras de alto rendimento que permitem a formação de vapor dentro do economizador, porém, deve-se levar em conta os impactos no fluxo e outros aspectos durante seu funcionamento. A faixa do fluxo da água no economizador varia normalmente de 600 a 800 kg/m²s e a velocidade do gás de 7 a 15 m/s (BASU 2006, p. 256).

2.3 Trocador de calor de tubos concêntricos

Trocadores de calor são dispositivos que promovem a transferência de energia térmica entre dois fluidos escoando com temperaturas diferentes, na maioria dos casos, sem que ocorra mistura direta. Usualmente, o funcionamento desse dispositivo acontece em decorrência da convecção dos dois fluidos separados por uma parede. Nesse sentido, o calor é trocado, primeiramente, entre o fluido quente e a parede, através da convecção. Em seguida, entre os lados opostos da parede mediante o mecanismo da condução. E, por fim, entre a parede e o fluido frio, novamente por convecção. A utilização dos trocadores de calor abrange uma grande variedade de cenários, sendo eles, controle da temperatura de um ambiente, processos químicos, na produção de potência e na recuperação de calor em processos diversos (ÇENGEL, 2012, p. 629; BERGMAN et al., 2019, p. 419).

A determinação do coeficiente global de transferência de calor é fundamental para a análise de trocadores de calor. Na ausência de geração interna de energia, esse coeficiente pode ser calculado a partir das resistências térmicas equivalentes envolvidas nos mecanismos de transferência de calor. Deve-se ressaltar que, em sistemas reais, existe uma tendência na redução de desempenho dos trocadores de calor ao longo do tempo, uma vez que, dependendo das características dos fluidos de

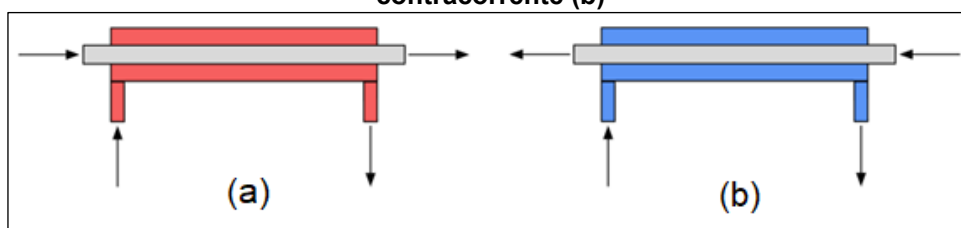
trabalho, pode haver a formação de uma camada de depósitos nas paredes internas do equipamento. Com isso, faz-se necessário, durante o cálculo da resistência total do sistema, a adição de um fator de incrustação ou deposição. Essa incrustação pode ter diferentes origens, sendo elas (ÇENGEL, 2012; BERGMAN et al., 2019):

- Químicas: Através de deposições de compostos químicos;
- Biológicas: Com a formação de camadas de algas;
- Físicas: Pela abrasão de impurezas na superfície interna do equipamento.

O desempenho de trocadores de calor pode ser avaliado a partir de vários métodos de análise. Um deles é o baseado na média logarítmica das diferenças de temperatura (MLDT), adotado no presente trabalho por causa das características das simulações. A determinação da MLDT depende da configuração do trocador de calor, estabelecida pelo movimento relativo existente entre as correntes dos fluidos. No caso dos trocadores de calor de tipo concêntrico, as configurações usuais são as de escoamento paralelo e escoamento contracorrente (ÇENGEL, 2012; BERGMAN et al., 2019).

O trocador de calor de tubos concêntricos é um dos mais simples em construção e operação. Na configuração de escoamento (Figura 3a), ambos os fluidos escoam no mesmo sentido, enquanto, na configuração de escoamento contracorrente (Figura 3b), os fluidos se movimentam em sentidos opostos (ÇENGEL, 2012, p. 630; BERGMAN et al., 2019, p. 419).

Figura 3 - Trocador de calor de tubos concêntricos em escoamento paralelo (a) e contracorrente (b)



Fonte: Autoria própria.

2.4 Python

Python é uma linguagem de programação orientada a objeto com características excelentes para aplicações em engenharia, pois possui sintaxe mais

simples e os programas escritos nele não necessitam ser compilados na linguagem de máquina, mas sim através de um interpretador. Um interpretador é um programa que processa uma aplicação através da execução de cada instrução contida nele, sendo então um processo semelhante ao de tradução (KNUTH, 1997, p. 201; KIUSALAAS, 2005; NUDDS, 1973, p. 1).

Algumas das vantagens ofertadas pela utilização da linguagem Python estão listadas a seguir (KIUSALAAS, 2005, p. 2):

- A linguagem Python é *open-source*, ou seja, é de livre acesso e utilização;
- O Python é disponível para a maioria dos sistemas operacionais (Linux, Windows, MacOS, entre outros);
- Python possui uma sintaxe mais fácil e limpa, auxiliando no momento da aprendizagem;
- Python possui uma vasta variedade de bibliotecas para as mais diversas aplicações, que possuem instalação rápida e fácil.

Em contrapartida, as aplicações criadas utilizando Python não produzem aplicações autônomas, ou seja, aplicações criadas em Python podem ser executadas apenas em computadores ou máquinas virtuais que possuem o interpretador do Python instalado (KIUSALAAS, 2005, p. 1).

2.4.1 Bibliotecas Python

Como descrito anteriormente, devido à linguagem de programação Python ser *open-source*, existe um vasto acervo de bibliotecas criadas pelos usuários para diversas aplicações.

Bibliotecas ou módulos são extensões, que devem ser instaladas no computador ou máquina virtual e contém funções pré-programadas. O programador pode importar determinada biblioteca ou apenas a função desejada dessa para sua utilização na aplicação em construção. Nesse trabalho, serão utilizadas as bibliotecas:

NumPy: Possibilita trabalhar com *arrays* multidimensionais, além de fornecer outras ferramentas que facilitam cálculos numéricos e científicos;

Pandas: Utilizada para realizar atividades de análise de dados, como manipulação de *dataframes*;

Matplotlib: Utilizada para criar diferentes tipos de gráficos, normalmente é utilizada em conjunto à biblioteca NumPy.

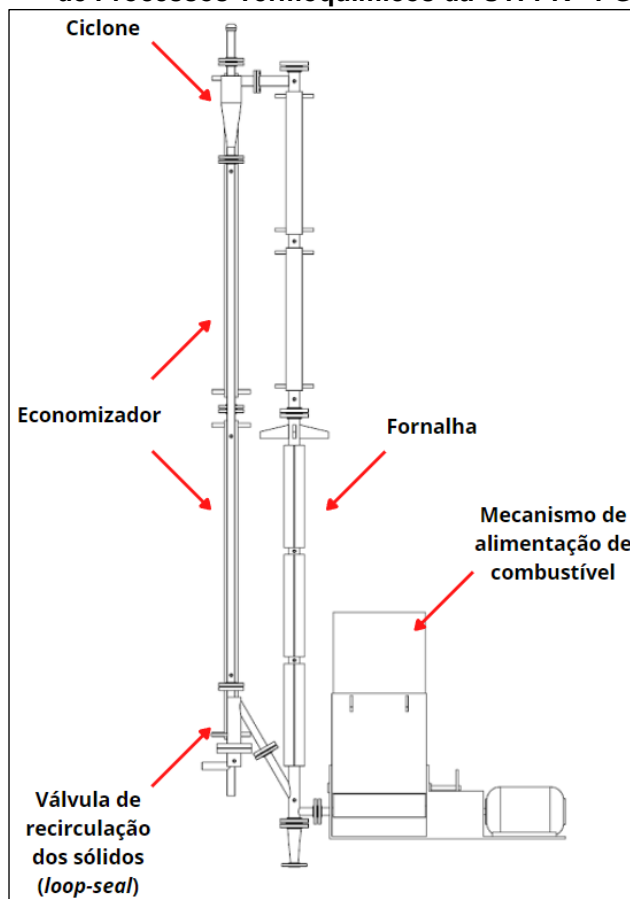
3 METODOLOGIA

Esta seção apresenta os detalhes construtivos do economizador e do procedimento desenvolvido para a realização das simulações computacionais da sua operação na linguagem Python.

3.1 Descrição do economizador

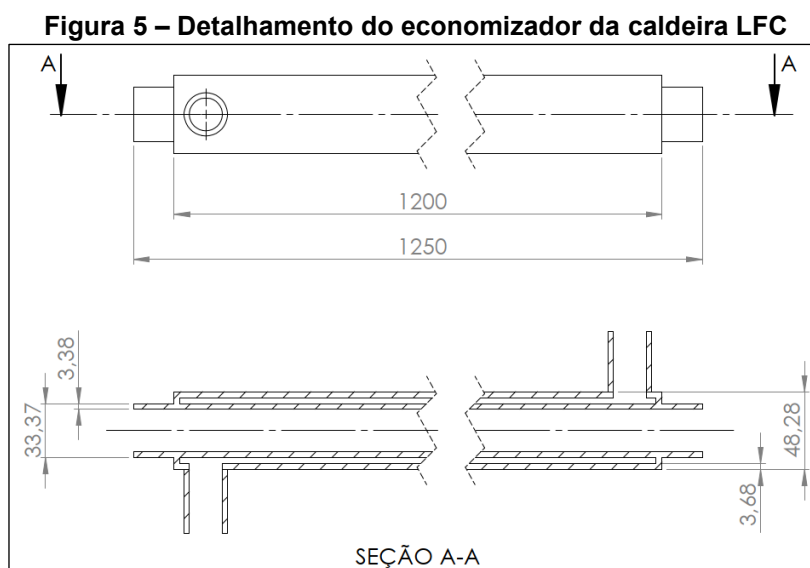
O economizador analisado nesse trabalho faz parte de uma caldeira piloto de leito fluidizado circulante, atualmente em construção no Laboratório de Processos Termoquímicos da UTFPR, Campus Ponta Grossa. O circuito de fluidização da caldeira, com a identificação dos seus principais componentes, está representado na Figura 4.

Figura 4 - Vista frontal com alguns equipamentos identificados da caldeira de LFC do Laboratório de Processos Termoquímicos da UTFPR - PG



Fonte: Autoria própria

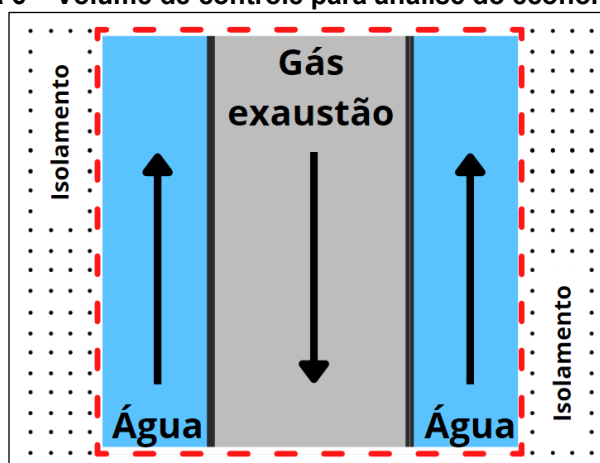
O economizador fica localizado na saída dos gases do separador de partículas (ciclone) do circuito de fluidização, tendo como fluidos operantes, gás de combustão resultante da queima do combustível na fornalha, e água que é alimentada externamente ao sistema. O sistema contém dois economizadores iguais, conectados em série e construídos em aço inoxidável AISI 304. Na Figura 5, apresentam-se as principais dimensões de cada economizador, em milímetros.



Fonte: Autoria própria

O escoamento do gás de combustão ocorre no tubo circular central e, o escoamento da água, na região anular. A configuração padrão de funcionamento é de escoamento contracorrente. O volume de controle utilizado para a análise do evaporador é ilustrado na Figura 6.

Figura 6 – Volume de controle para análise do economizador.



Fonte: Autoria própria

Esse trabalho não aborda a modelagem completa da caldeira. Porém, alguns parâmetros de operação padrão relativos ao gerador de vapor são necessários para a realização das simulações do economizador, como por exemplo, a vazão mássica e a temperatura de entrada do gás de combustão esperadas para o sistema. Esses e outros dados são apresentados na Tabela 1.

Tabela 1 - Condições padrão de operação do economizador

Parâmetro	Valor	Unidade
Vazão mássica gás de exaustão	8,6	kg/h
Pressão absoluta do gás de exaustão	1	bar
Temperatura de entrada do gás de exaustão	780	°C
Pressão absoluta da água	5	bar
Temperatura de entrada da água	25	°C

Fonte: Autoria própria

3.2 Modelagem operacional do economizador

Para a realização da modelagem operacional do economizador, foram consideradas as seguintes hipóteses:

- O economizador opera em regime permanente;
- A transferência de calor entre o gás de exaustão e a água acontece apenas na direção radial;
- O escoamento do gás de combustão e da água estão completamente desenvolvidos;
- O fluxo térmico imposto na superfície de troca térmica é uniforme;
- As quedas de pressão do gás de exaustão e da água que atravessam o economizador são desprezíveis;
- As paredes externas do economizador são adiabáticas;
- O gás de combustão é modelado como gás ideal;
- As propriedades termofísicas do gás de combustão são as mesmas do ar nas respectivas temperaturas;
- A condutividade térmica do material do evaporador é constante, no valor de $15,5 \text{ W}/(m \cdot K)$.

A partir das hipóteses listadas anteriormente, a taxa de transferência de calor (Q) foi calculada mediante as Equações (1) a (3):

$$Q = \dot{m}_g c_{p,g} (T_{q,ent} - T_{q,sai}) \quad (1)$$

$$Q = \dot{m}_w c_{p,w} (T_{f,sai} - T_{f,ent}) \quad (2)$$

$$Q = UA(MLDT) \quad (3)$$

Sendo que:

\dot{m}_g e \dot{m}_w - Vazão mássica do gás de combustão e da água, respectivamente;
 $c_{p,g}$ e $c_{p,w}$ - Calor específico a pressão constante do gás de combustão e da água, respectivamente;

$T_{q,ent}$ e $T_{f,ent}$ - Temperatura de entrada do gás de combustão e da água, respectivamente;

$T_{q,sai}$ e $T_{f,sai}$ - Temperatura de saída do gás de combustão e da água, respectivamente;

U - Coeficiente global de transferência de calor;

A - Área superficial de troca térmica;

$MLDT$ - Média logarítmica das diferenças de temperaturas.

Os valores das propriedades termofísicas dos fluidos, como calor específico e número de Prandtl (Pr), são necessários para os cálculos dos números de Reynolds e Nusselt. Seus valores foram determinados a partir de tabelas que se encontram nos Anexos A, B e C nesse trabalho. Segundo Bergman et al. (2019, p. 321), o valor da temperatura para a leitura desses valores é a temperatura média do fluido (T_m). Para sua determinação, é utilizada a Equação (4), que ilustra a média aritmética simples das temperaturas de entrada e saída do fluido (T_{entra} e T_{sai}).

$$T_m = \frac{T_{entra} + T_{sai}}{2} \quad (4)$$

Para valores de temperatura não apresentados nas tabelas, foi aplicado uma interpolação linear para determinar as propriedades. A interpolação linear é representada pela Equação (5) (NGUYEN; MARTIN, 2003):

$$y = y_0 + (y_1 - y_0) \frac{(x - x_1)}{(x_0 - x_1)} \quad (5)$$

Sendo:

y – Valor de saída;

x – Valor de entrada;

x_0 e y_0 – Primeiro valor dependente e independente, respectivamente;

x_1 e y_1 – Segundo valor dependente e independente, respectivamente.

Para o cálculo do coeficiente global de transferência de calor (U), foi utilizada a Equação (6), que envolve a determinação da resistência térmica total (R_{tot}) existente entre os dois escoamentos:

$$UA = \frac{1}{R_{tot}} \quad (6)$$

Segundo Bergman et al. (2019, p. 421), a resistência térmica total de um trocador de calor concêntrico não aletado pode ser determinada pela Equação (7):

$$R_{tot} = \frac{1}{h_i A_i} + \frac{R_{d,i}''}{A_i} + \frac{\ln(d_e/d_i)}{(2\pi k_{mat} L)} + \frac{R_{d,e}''}{A_e} + \frac{1}{h_e A_e} \quad (7)$$

Sendo que:

h_i e h_e – Coeficientes convectivos do escoamento interno e externo, respectivamente;

A_i e A_e – Áreas das superfícies em contato com os escoamentos interno e externo, respectivamente;

$R_{d,i}''$ e $R_{d,e}''$ – Fatores de incrustação do escoamento interno e externo, respectivamente;

d_e e d_i – Diâmetros externo e interno da parede do tubo que separa os escoamentos;

k_{mat} – Condutividade térmica do material que separa os escoamentos;

L – Comprimento efetivo da superfície de troca térmica.

Os valores de $R_{d,i}''$ e $R_{d,e}''$, também chamados de fatores de deposição, são associados a uma fina camada de depósitos que se acumulam na superfície de troca térmica ao longo do tempo de operação do equipamento. Essa camada resulta em uma redução da eficiência da transferência de calor e pode ser representada como uma resistência térmica adicional na modelagem do sistema (ÇENGEL, 2012, p. 651). Valores representativos dos fatores são apresentados na Tabela 2. Para esse trabalho, foram selecionados os valores de $R_{d,i}'' = 0,0004 \text{ (m}^2 \cdot \text{K)/W}$ e $R_{d,e}'' = 0,0001 \text{ (m}^2 \cdot \text{K)/W}$, correspondente a ar e água abaixo de 50°C.

Tabela 2 - Fatores de incrustação representativos

Fluido	$R_d'' \text{ [m}^2 \cdot \text{K/W]}$
Água destilada, água-marinha, águas fluviais, água de alimentação de caldeiras	...
Abaixo de 50 °C	0,0001
Acima de 50 °C	0,0002
Outros fluidos	...
Óleo combustível	0,0009
Vapor (livre de óleo)	0,0001
Refrigerantes (líquidos)	0,0002
Refrigerantes (vapor)	0,0004
Vapores de álcool	0,0001
Ar	0,0004

Fonte: Adaptado de Çengel (2012, p. 651)

As áreas da superfície de contato com os escoamentos interno e externo foram determinadas pelas Equações (8) e (9), respectivamente:

$$A_i = \pi \cdot d_i \cdot L \quad (8)$$

$$A_e = \pi \cdot d_e \cdot L \quad (9)$$

Para a determinação dos coeficientes convectivos interno e externo (h_i e h_e), foi estabelecido o regime de escoamento dos fluidos em ambas as regiões. Para a região circular, utilizou-se a Equação (10), para determinação do número de Reynolds (Re_D) (BERGMAN et al., 2019, p. 306).

$$Re_D = \frac{\rho u_m D}{\mu} \quad (10)$$

Sendo que:

ρ – Massa específica do fluido;

D – Diâmetro da seção transversal do tubo de escoamento;

μ – Viscosidade dinâmica do fluido.

A velocidade média do fluido (u_m) é determinada pela Equação (11):

$$u_m = \frac{\dot{m}}{\rho A_{tr}} \quad (11)$$

Sendo que:

\dot{m} – Vazão mássica do fluido;

A_{tr} – Área da seção transversal do duto de escoamento.

Para a região anular, o diâmetro utilizado para a Equação (10) deve ser o diâmetro hidráulico, determinado pela Equação (12) a partir da área transversal (A_{tr}) e o perímetro (P) da região anular (BERGMAN, 2019, p. 326-327).

$$D_h \equiv \frac{4A_{tr}}{P} \quad (12)$$

Sendo que: $D_h = D_i - d_e$.

Uma vez conhecido o regime de escoamento do fluido, foi determinada a correlação apropriada para o cálculo do número de Nusselt.

Segundo Bergman et al. (2019, p. 317), para a região circular central com escoamento laminar ($Re_D \leq 2300$) desenvolvido e fluxo térmico uniforme aplicado na superfície, o valor de Nusselt médio (\overline{Nu}_D) é constante e igual à 4,36. Para o regime em escoamento em transição ($2300 < Re_D < 3000$) e turbulento ($3000 \leq Re_D \leq 5 * 10^7$) e com $0,5 < Pr \leq 200$, a correlação de Gnielinski, dada pela Equação (13), foi adotada nesse trabalho (BERGMAN et al., 2019, p.321).

$$\overline{Nu}_D = \frac{(f/8)(Re_D - 1000)Pr}{1 + 12,7(f/8)^{2/3}(Pr^{2/3} - 1)} \quad (13)$$

Sendo que o fator de atrito de Darcy (f) foi determinado pela Equação (14) (BERGMAN et al., 2019, p. 307):

$$f = [1,82 * \log_{10}(Re_D) - 1,64]^{-2} \quad (14)$$

Para as condições de escoamento e térmicas impostas, foram utilizadas as Equações (15) e (16) para se determinar o número de Nusselt médio para escoamento no diâmetro interno e externo da região anular (Nu_i e Nu_e) (BERGMAN et al., 2019, p. 327).

$$Nu_i = \frac{Nu_{ii}}{1 - (q_e''/q_i'')\theta_i''} \quad (15)$$

$$Nu_e = \frac{Nu_{ee}}{1 - (q_i''/q_e'')\theta_e''} \quad (16)$$

Sendo q_e'' e q_i'' o fluxo de calor nas superfícies externa e interna, respectivamente. Os valores dos coeficientes de influência (Nu_{ee} , Nu_{ii} , θ_i'' e θ_e'') foram escolhidos a partir da Tabela 3, conhecendo-se a razão entre diâmetro interno e o externo da seção anular.

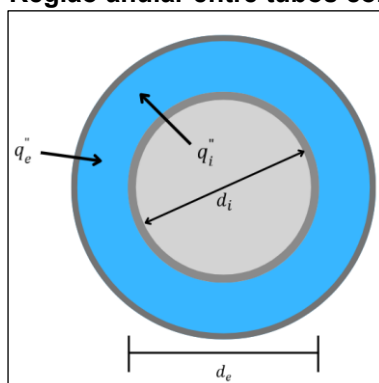
Tabela 3 - Coeficientes de influência para o escoamento laminar plenamente desenvolvido em uma região anular circular com fluxo térmico uniforme mantido nas duas superfícies

D_i/D_e	Nu_{ii}	Nu_{ee}	θ_i^*	θ_e^*
0	-	4,364	∞	0
0,05	17,81	4,792	2,18	0,0294
0,10	11,91	4,834	1,384	0,0562
0,20	8,499	4,883	0,904	0,1039
0,40	6,583	4,979	0,602	0,1822
0,60	5,912	5,099	0,474	0,2455
0,80	5,58	5,24	0,401	0,298
1,00	5,385	5,385	0,346	0,346

Fonte: Adaptado de Bergman et al. (2019, p. 327)

A Figura 7 ilustra a região anular entre tubos concêntricos. Com base nas hipóteses da modelagem, o fluxo de calor externo (q_e'') é nulo, uma vez que as paredes externas do equipamento são isoladas.

Figura 7 - Região anular entre tubos concêntricos



Fonte: Autoria própria

Desta forma, haverá apenas o valor de Nusselt da região interna (Nu_i) e a razão de fluxos de calor será igual a zero, logo:

$$Nu_i = Nu_{ii} \quad (17)$$

Para a região anular, com regime de escoamento em transição ($2300 < Re_D < 3000$) e turbulento ($3000 \leq Re_D \leq 5 * 10^7$) e com $0,5 < Pr \leq 200$, o número de Nusselt pode ser determinado através da mesma correlação de Gnielinski utilizada para a região tubular circular (Equação (13)).

Após isso, utilizou-se a Equação (18) para encontrar o valor do coeficiente convectivo (h) correspondente a cada região de escoamento, sendo a condutividade

térmica do fluido (k), determinada através da temperatura média do fluido (BERGMAN et al., 2019, p. 317).

$$\overline{Nu}_D = \frac{hD}{k} \quad (18)$$

$$h = \frac{\overline{Nu}_D k}{D} \quad (19)$$

Segundo Bergman et al. (2019, p. 423), em um trocador de calor operando em modo contracorrente, o valor da média logarítmica das diferenças de temperatura é calculado mediante a Equação (20).

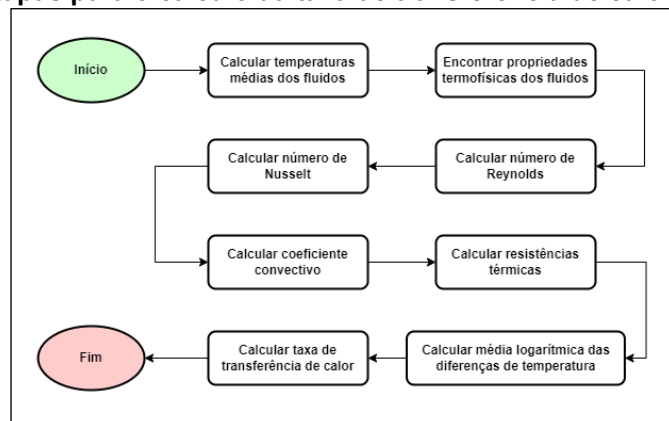
$$MLDT = \left[\frac{(\Delta T_2 - \Delta T_1)}{\ln \left(\frac{\Delta T_2}{\Delta T_1} \right)} \right] \quad (20)$$

Sendo que:

$$\begin{cases} \Delta T_1 = T_{q,entra} - T_{f,sai} \\ \Delta T_2 = T_{q,sai} - T_{f,entra} \end{cases} \quad (21)$$

A Figura 8 ilustra, através de um fluxograma, as etapas descritas nessa seção para a determinação da taxa de transferência de calor correspondente à Equação (3).

Figura 8 – Etapas para o cálculo da taxa de transferência de calor (Equação (3))



Fonte: Autoria própria

A fim de verificar se a configuração de operação do economizador é possível, é necessário que os valores de Q calculados pelas Equações (1) à (3) sejam iguais ou muito próximos. Nesse trabalho, foi estabelecido que a porcentagem de erro ($E_{\%}$) não pode exceder 1% acima ou abaixo do valor base (Q_{base}). O valor base corresponde à taxa de troca térmica calculada pela Equação (3), pois essa correlação aborda parâmetros de ambos os fluidos. O valor de teste (Q_{teste}) é determinado apenas pela taxa de transferência de calor calculada com a Equação (1), pois, uma vez atendido o requisito mínimo de erro, o valor da taxa de transferência de calor pela Equação (3) será utilizado para se determinar a vazão mássica da água. A expressão para o cálculo da porcentagem de erro é apresentada na Equação (22).

$$E_{\%} = \frac{|Q_{base} - Q_{teste}|}{Q_{base}} \cdot 100 \quad (22)$$

Após o cumprimento da restrição de 1%, encontra-se o valor de vazão mássica da água (\dot{m}_w) necessária para a operação do equipamento, conforme Equação (23):

$$\dot{m}_w = \frac{Q}{c_{p,w}(T_{f,sai} - T_{f,ent})} \quad (23)$$

3.3 Desenvolvimento do algoritmo em Python

Os códigos foram desenvolvidos na linguagem Python em sua versão 3.9.5. As bibliotecas utilizadas foram:

- Pandas (versão 1.4.1);
- Matplotlib (versão 3.5.1);
- Numpy (versão 1.22.3).

Os códigos completos são apresentados nos Apêndices A e B desse trabalho e, também, em um repositório *online* na plataforma GitHub¹.

¹ Disponível em: <https://github.com/kagesg/bachelors-thesis>

Para o desenvolvimento dos algoritmos, inicialmente, as tabelas com as propriedades termofísicas da água e do ar (Anexo A e B), pressão e temperatura da água saturada (Anexo C) e a tabela com os valores dos coeficientes de influência (Tabela 3) foram inseridos no código. Porém, um tratamento desses dados foi necessário para possibilitar seu uso. O processo de tratamento dos dados consiste na conversão em *dataframes* (dados com estrutura de tabela). Em seguida, foram retiradas as 3 primeiras linhas referentes ao cabeçalho das tabelas (nome, sigla e unidade das colunas), e os índices do *dataframe* foram redefinidos. Por fim, os dados foram convertidos para o tipo *float* (números reais) e, as células em branco, preenchidas com valores nulos.

Na sequência, foram definidas as variáveis utilizadas para configurar o modo de operação e algumas funções, dentre elas, parâmetros geométricos do sistema, (diâmetros, comprimento efetivo do equipamento, áreas de seção transversal, áreas de troca térmica e razão de diâmetros da região anular), bem como listas com valores para automatizar o preenchimento de determinadas entradas para as funções dos cálculos.

Feito isso, a condição inicial padrão foi definida. Seus valores foram informados na Tabela 1 da Seção 3.1.

Para realizar as simulações, foi necessário se definir dois objetos: o primeiro se refere às condições de simulação definidas. Para tal, as variáveis foram divididas em dois grupos com intervalos e valores pré-selecionados: variáveis principais e variáveis secundárias. Seus intervalos e lista de valores estão descritos na Tabela 4.

Tabela 4 - Variáveis principais e secundárias

	Variável principal		Variável secundária
	Intervalo	Passo	Lista
Temperatura entrada água	[10;30]	1	(10,20,35)
Temperatura entrada do gás	[740;860]	5	(740,780,860)
Pressão absoluta água	[3;12]	0,5	(3,5,12)

Fonte: Autoria própria

Ao total, foram formadas 6 combinações de simulação. Cada combinação foi realizada entre uma variável principal e uma secundária de diferente grupo. As combinações, em ordem de simulação, são apresentadas no Quadro 3.

Quadro 3 - Combinação de variáveis para simulação

Variável principal	Variável secundária
Temperatura entrada água	Pressão absoluta água
	Temperatura entrada do gás
Temperatura entrada do gás	Pressão absoluta água
	Temperatura entrada água
Pressão absoluta água	Temperatura entrada água
	Temperatura entrada do gás

Fonte: Autoria própria

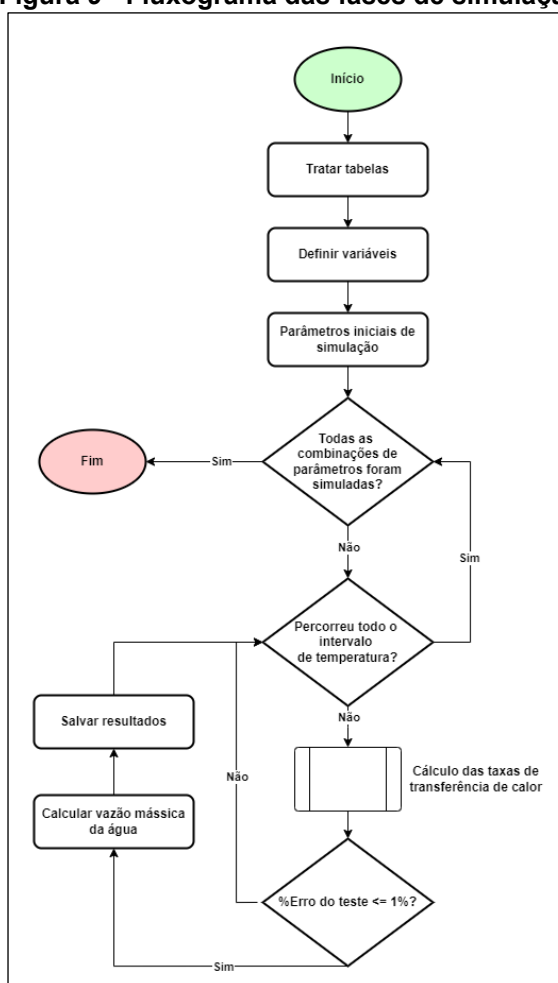
O segundo objeto foi o intervalo de variação da temperatura de saída do gás de exaustão. Foi utilizado o intervalo de 250 °C a 150 °C, com um passo de 0,5 °C.

Para realizar as simulações, foi necessário criar estruturas de *loop* (laços de repetição) para executar os cálculos, até que todo o intervalo das variáveis principais, secundárias e de temperatura de saída do gás de combustão, fossem percorridas.

Conforme descrito na seção secundária anterior, um teste condicional é utilizado para verificar o erro percentual existente entre as taxas de transferência de calor determinadas pelas Equações (1) e (3).

O processo de simulação é esquematizado no fluxograma da Figura 9:

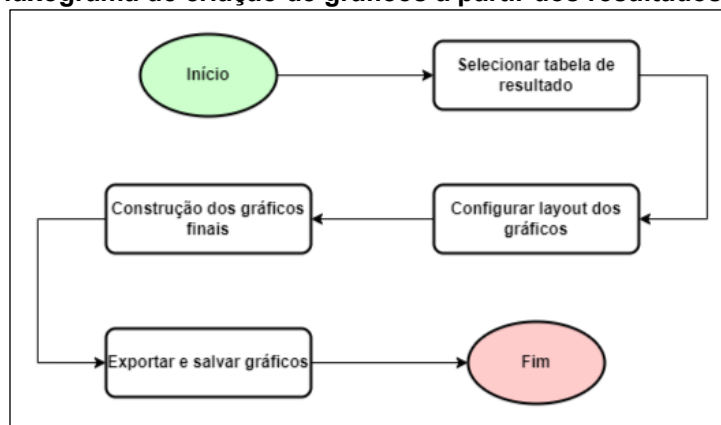
Figura 9 - Fluxograma das fases de simulação



Fonte: Autoria própria

Os gráficos foram criados mediante um segundo código elaborado em Python, apresentado no Apêndice B. A Figura 10 ilustra as etapas do processo para confecção desses gráficos.

Figura 10 - Fluxograma de criação de gráficos a partir dos resultados de simulação



Fonte: Autoria própria

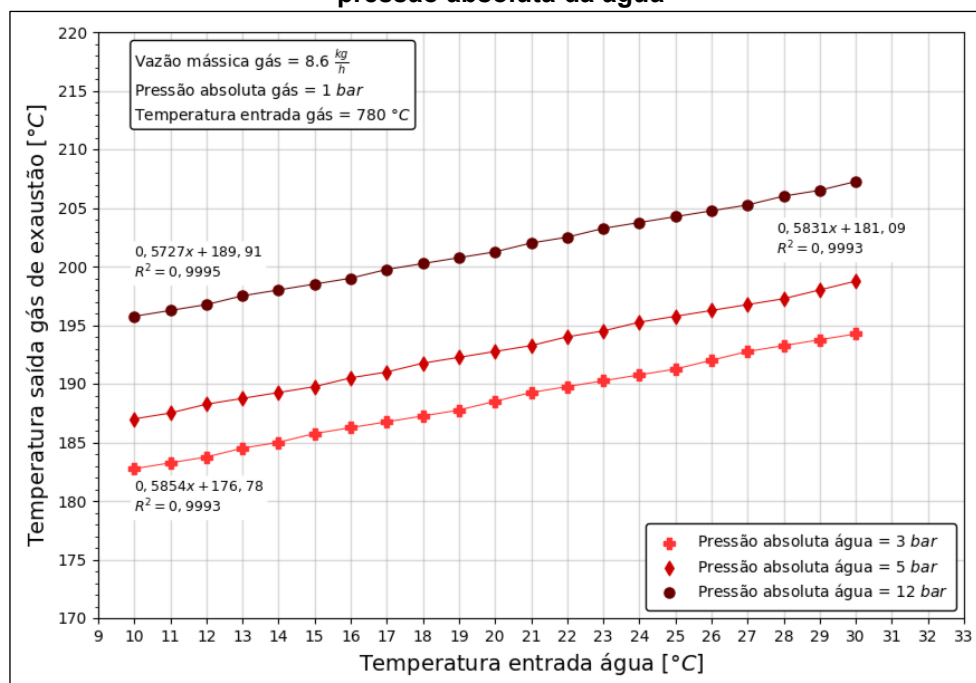
4 RESULTADOS E DISCUSSÃO

Nesta seção são apresentados e discutidos os resultados obtidos nas simulações realizadas. Os resultados foram focados na determinação dos valores da temperatura de saída do gás de exaustão e da vazão mássica da água em função da mudança combinada dos seguintes parâmetros: temperatura de entrada da água, temperatura de entrada do gás de exaustão e pressão absoluta da água.

4.1 Efeitos da Variação da temperatura de entrada e pressão absoluta da água

O Gráfico 2 mostra o comportamento da temperatura de saída do gás de exaustão em função da mudança da temperatura de entrada e da pressão absoluta da água.

Gráfico 2 - Temperatura de saída do gás de exaustão em função da temperatura de entrada e pressão absoluta da água



Fonte: Autoria própria

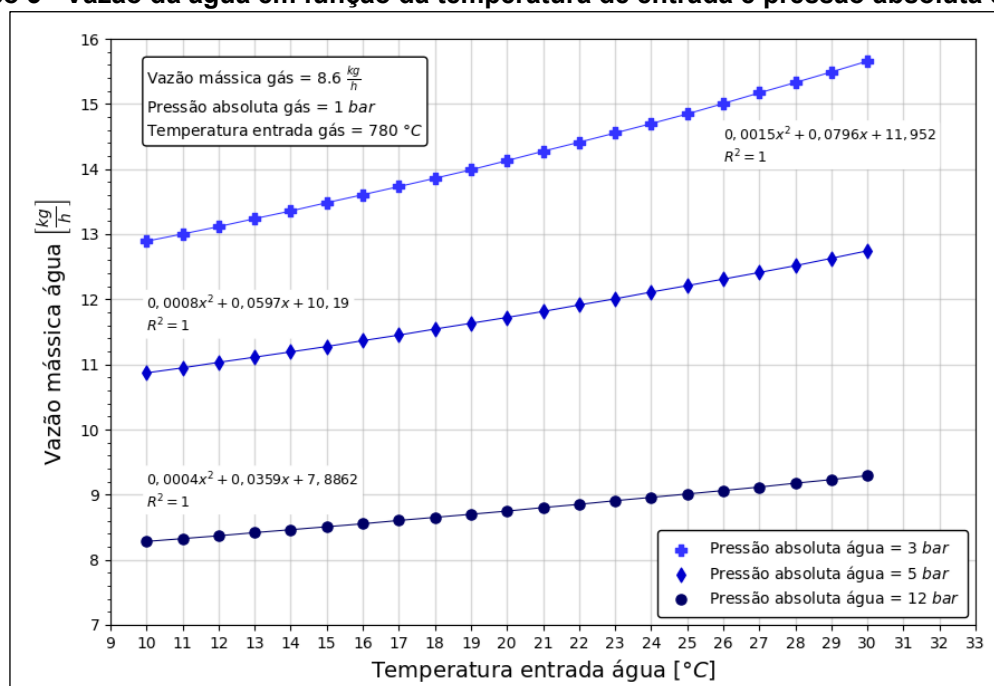
As simulações indicam uma tendência aproximadamente linear e diretamente proporcional entre a temperatura de saída do gás de exaustão e a temperatura de entrada da água. Para cada curva, o aumento da temperatura de entrada da água forneceu um incremento percentual de aproximadamente 6,3% para a operação em 3

e 5 bar, e de 5,9% para 12 bar. Esse resultado mostra-se coerente, uma vez que, conforme as Equações (1) e (2), ao manter constantes a vazão mássica e a temperatura de entrada do gás de exaustão, bem como a temperatura de saída da água (pressão fixa), a diminuição da variação da temperatura imposta na corrente da água também reduz o potencial para o resfriamento do gás de exaustão.

Acerca do impacto causado pela variação da pressão absoluta da água, pode-se observar que houve uma translação das curvas no sentido positivo do eixo “y” com o crescimento do valor dessa pressão. Nos casos estudados, um aumento na pressão da água de 3 bar para 5 bar (equivalente a um aumento de 16,1% da temperatura de saída da água), houve uma alta de 2,3% da temperatura de saída do gás.

O Gráfico 3 ilustra a relação entre os parâmetros de temperatura de entrada e pressão absoluta da água com a vazão mássica da água.

Gráfico 3 - Vazão da água em função da temperatura de entrada e pressão absoluta da água



Fonte: Autoria própria

Nota-se uma relação de crescimento parabólico entre o valor da temperatura da água e a vazão mássica da mesma. Como previsto pela Equação (2), uma menor diferença na variação da temperatura da água é compensada com maior quantidade de líquido escoando pelo economizador.

A variação entre as vazões mássicas determinadas nas temperaturas de entrada de 10 °C e 30 °C, para pressão de 3 bar, é de 2,76 kg/h. Já, para o mesmo

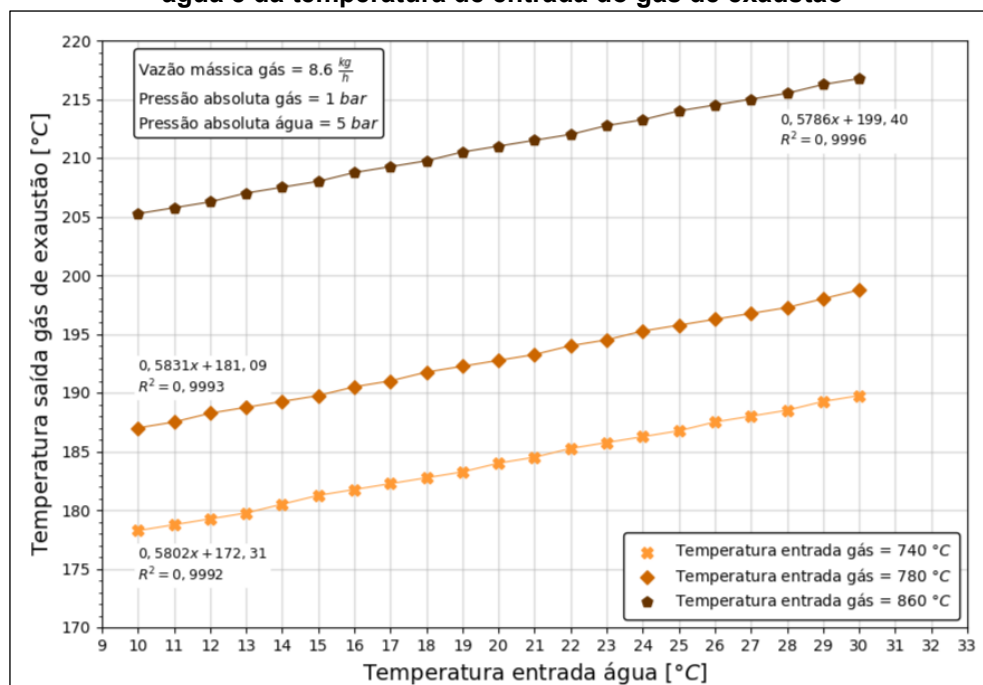
intervalo de temperatura, mas com a pressão igual à 12 bar, a diferença foi de apenas 1,01 kg/h. Isto ocorre porque, com o aumento da pressão da água, exige-se uma maior temperatura de saída, obrigando à diminuição da vazão mássica do líquido para o cumprimento da taxa de transferência de calor.

Portanto, o comportamento obtido sugere que o aumento da pressão de operação do economizador limita a variação na vazão mássica da água que pode ser pré-aquecida, fazendo-a também menos dependente da sua temperatura de entrada.

4.2 Efeitos da variação da temperatura de entrada da água e da temperatura de entrada do gás de exaustão

O Gráfico 4 ilustra o impacto da temperatura de entrada da água e da temperatura de entrada do gás de exaustão na temperatura de saída do gás de exaustão.

Gráfico 4 - Temperatura de saída de gás de exaustão em função da temperatura de entrada da água e da temperatura de entrada do gás de exaustão



Fonte: Autoria própria

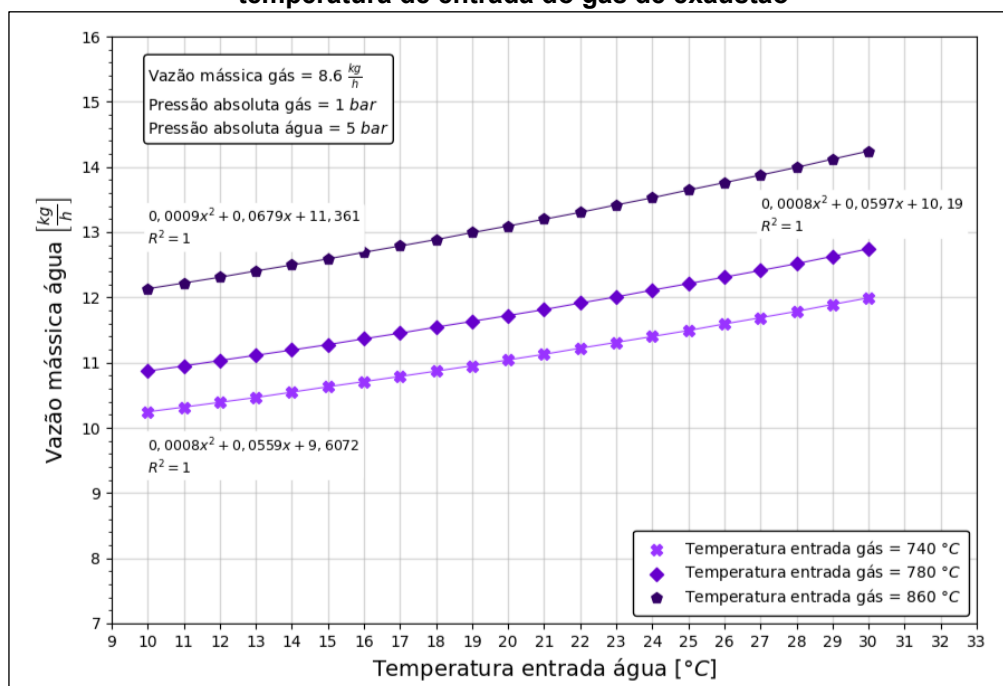
Como descrito na seção 4.1, observa-se a mesma relação crescente entre a temperatura de entrada da água e a temperatura de saída do gás de exaustão. Isso se deve, novamente, às implicações trazidas pelas variações de temperatura

experimentadas pelos fluidos nas Equações (1) e (2). Os aumentos percentuais, entre o limite inferior e superior de cada curva (740 °C, 780 °C e 860 °C), foram 6,5%, 6,3% e 5,6%, respectivamente.

Ainda com base na estrutura da Equação (1), era de se esperar que com o aumento da temperatura de entrada do gás de exaustão no equipamento, a curva realizasse uma translação positiva no eixo “y”. Para um aumento de 40 °C da temperatura de entrada do gás, ou seja, de 740 °C para 780 °C, houve um crescimento de 8,75 °C da temperatura de saída do gás, representando uma variação percentual de 78% em comparação com a diferença de temperatura de entrada do gás. Para um aumento de 780 °C para 860 °C da temperatura de entrada, houve um aumento de 18,25 °C da temperatura de saída do gás. Este valor representa uma variação percentual de 77,2%, resultado próximo ao valor encontrado na comparação anterior.

O Gráfico 5 apresenta os valores simulados de vazão mássica da água em função da mudança de temperatura de entrada da água e do gás de exaustão.

Gráfico 5 - Vazão mássica da água em função da temperatura de entrada da água e da temperatura de entrada do gás de exaustão



Fonte: Autoria própria

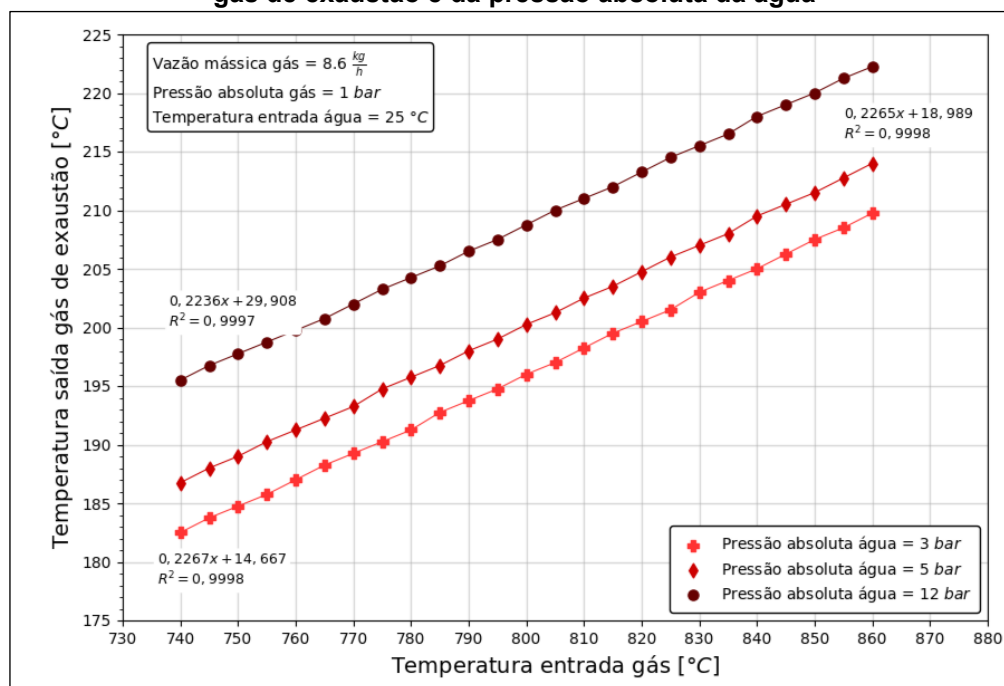
O Gráfico 5 possui comportamento semelhante ao Gráfico 3, porém, pode-se afirmar que o impacto da variação de temperatura de entrada do gás de exaustão é menor para a determinação da vazão mássica de água em comparação com a

variação da pressão absoluta da água. Para efeitos de validação, ao seleccionar o valor de vazão mássica encontrados nos Gráfico 3 e Gráfico 5, para a mesma configuração de operação, eles devem possuir valores iguais ou muito próximos. Tal hipótese é confirmada, uma vez que, em ambos os gráficos, para uma temperatura de entrada da água de 10 °C, pressão absoluta da água de 5 bar e temperatura de entrada do gás de exaustão de 780 °C, os valores de vazão da água e temperatura de saída do gás são de 10,9 kg/h e 187 °C, respectivamente.

4.3 Efeitos da variação da temperatura de entrada do gás de exaustão e pressão absoluta da água

O Gráfico 6 apresenta a influência da temperatura de entrada do gás de exaustão e a pressão absoluta da água na temperatura de saída do gás. Nota-se um comportamento linear e crescente da temperatura de saída do gás de exaustão em função do aumento da temperatura de entrada do mesmo fluido. Pode-se observar também que as curvas apresentam uma inclinação maior se comparado aos Gráficos Gráfico 2 e Gráfico 4. A variação percentual entre os valores extremos da temperatura de entrada do gás, para cada uma das pressões absolutas (3 bar, 5 bar e 12 bar), foi de 14,9%, 14,6% e 13,7%, respectivamente. A variação média é cerca de 33% maior que a do Gráfico 2.

Gráfico 6 - Temperatura de saída do gás de exaustão em função da temperatura de entrada do gás de exaustão e da pressão absoluta da água

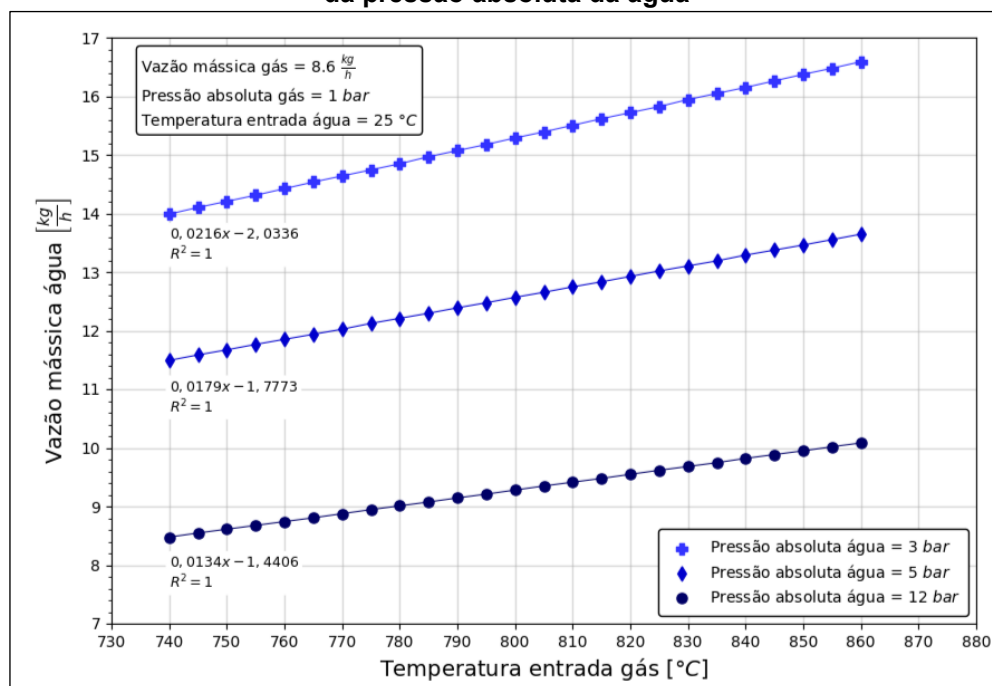


Fonte: Autoria própria

O Gráfico 6 ainda demonstra uma translação vertical crescente das curvas no sentido do eixo “y” com o aumento da pressão absoluta da água. O valor de mudança percentual, entre as curvas de 3 bar e 5 bar, é igual à 2,3%, mesma proporção encontrada no Gráfico 2.

No Gráfico 7, nota-se que o impacto da variação de pressão absoluta é semelhante ao obtido no Gráfico 3. Porém, as curvas apresentam um comportamento linear em relação à variação da temperatura de entrada do gás.

Gráfico 7 - Vazão mássica da água em função da temperatura de entrada do gás de exaustão e da pressão absoluta da água

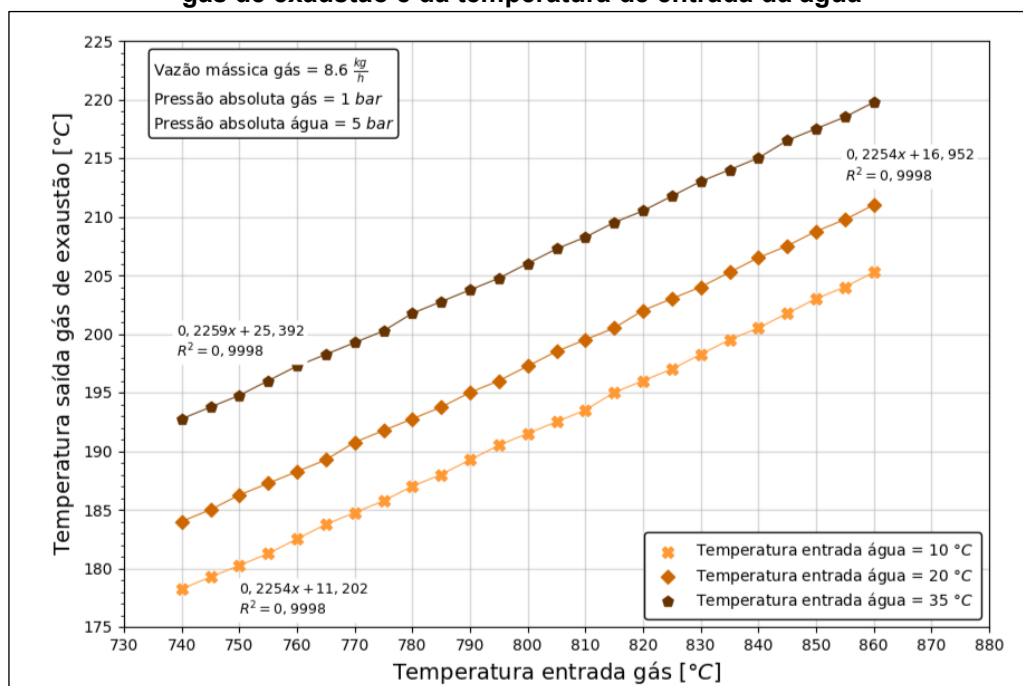


Fonte: Autoria própria

4.4 Efeitos da variação da temperatura de entrada do gás de exaustão e temperatura de entrada da água

O Gráfico 8 mostra um comportamento semelhante ao obtido no Gráfico 6. Tal representação é explicada pela relação direta das temperaturas de entrada e saída do gás de exaustão na Equação (1). Os valores dos acréscimos na temperatura de saída do gás foram de 15,1%, 14,8% e 14%, para as temperaturas de entrada de água de 10 °C, 20 °C e 35 °C, respectivamente. Observa-se também que, com o aumento da pressão e conseqüente aumento da temperatura de saída da água, é necessário que a temperatura de saída do gás aumente para se obter o equilíbrio de transferência de calor do sistema. A variação percentual entre os resultados simulados à 10 °C e 20 °C de temperatura de entrada da água, com uma temperatura de entrada do gás igual à 740 °C, foi de 3%.

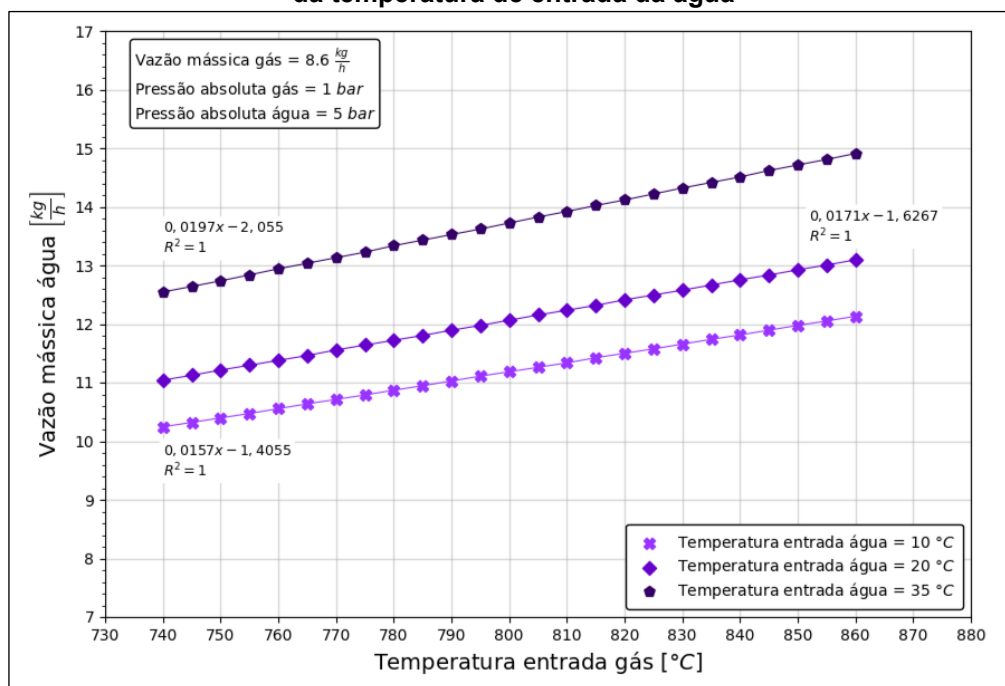
Gráfico 8 - Temperatura de saída do gás de exaustão em função da temperatura de entrada do gás de exaustão e da temperatura de entrada da água



Fonte: Autoria própria

O Gráfico 9 apresenta uma tendência de crescimento da vazão mássica em função do aumento da temperatura de entrada do gás. Ao comparar, para a mesma temperatura de entrada do gás, o valor de vazão mássica obtido em função da temperatura de entrada da água, pode-se observar que conforme a temperatura de entrada do gás aumenta, uma maior vazão mássica de água pode ser pré-aquecida pelo economizador. O Gráfico 9 se assemelha com o Gráfico 5. Realizando as análises numéricas, pôde-se determinar o valor de variação percentual das vazões calculadas, nos limites inferior e superior da temperatura do gás de exaustão, obtendo-se um valor médio de 18,6% para as três curvas.

Gráfico 9 - Vazão mássica da água em função da temperatura de entrada do gás de exaustão e da temperatura de entrada da água

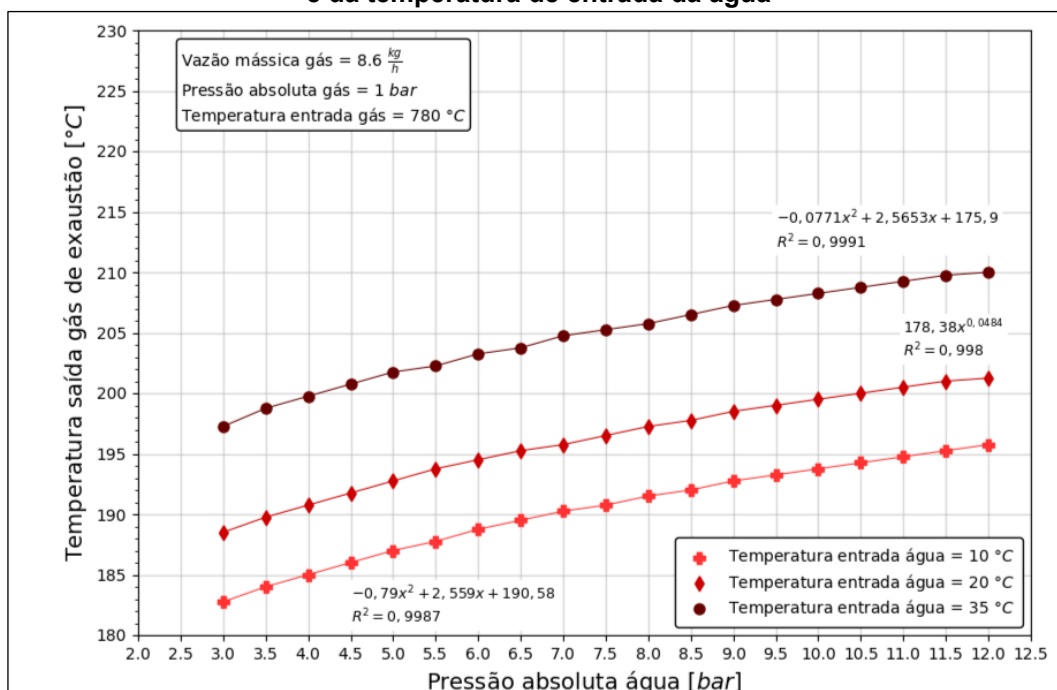


Fonte: Autoria própria

4.5 Efeitos da Variação da pressão absoluta da água e temperatura de entrada da água

O Gráfico 10 mostra um comportamento de acréscimo da temperatura de saída do gás de exaustão com o aumento da pressão absoluta da água. Porém, diferente do apresentado nos gráficos anteriores, as curvas possuem uma tendência quadrática. Observa-se também a representação de aumento da temperatura de saída do gás em função do aumento da temperatura de entrada da água. Esses comportamentos ocorrem pelos mesmos motivos descritos na Seção 4.1.

Gráfico 10 - Temperatura de saída do gás de exaustão em função da pressão absoluta da água e da temperatura de entrada da água

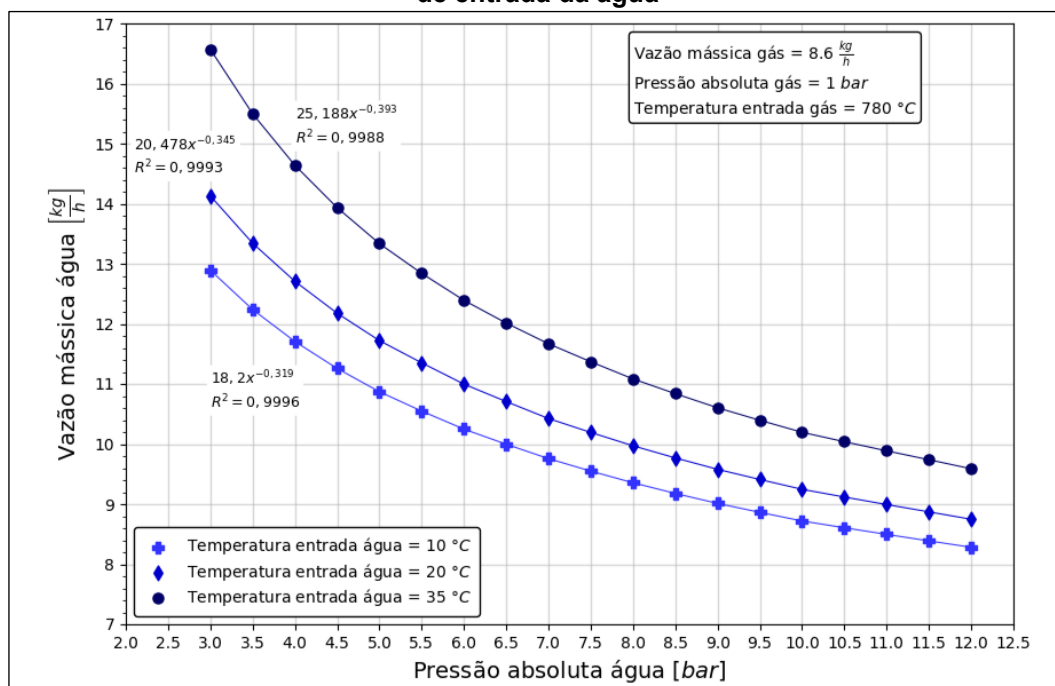


Fonte: Autoria própria

O Gráfico 11 mostra uma relação de decaimento da vazão mássica conforme há um aumento da pressão absoluta da água. As curvas ilustradas apresentam uma tendência polinomial. Comparativamente, a variação percentual entre os valores de vazão mássica encontrados à 3 bar e 12 bar, para uma temperatura de entrada da água constante à 10 °C, foi de 35,8%. Para o mesmo intervalo, mas com temperatura de entrada igual à 20 °C, a variação foi de 38,1%, e com temperatura de entrada igual à 35 °C, a variação foi de 42,1%.

A redução da vazão mássica requerida pelo equipamento, conforme há um aumento da pressão absoluta da água, deve-se ao conseqüente aumento da diferença de temperatura da água na Equação (2), a fim de se manter o equilíbrio nas taxas de transferência de calor.

Gráfico 11 - Vazão mássica da água em função da pressão absoluta da água e da temperatura de entrada da água

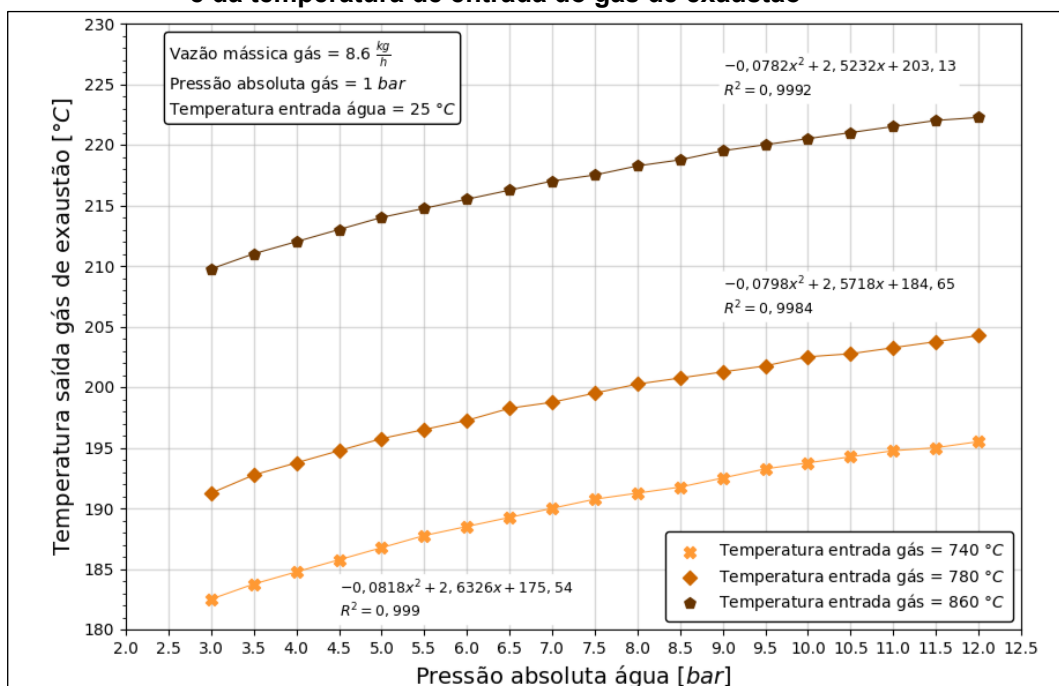


Fonte: Autoria própria

4.6 Efeitos da variação da pressão absoluta da água e temperatura de entrada do gás de exaustão

O Gráfico 12 verifica o aumento da temperatura de saída do gás de exaustão com o aumento da pressão da água, nota-se também um deslocamento vertical positivo da curva com o aumento da temperatura de entrada do gás. O comportamento é semelhante ao apresentado no Gráfico 10. O destaque se dá pelo deslocamento mais acentuado no Gráfico 12, entre as duas primeiras curvas (740 °C e 780 °C), em comparação ao Gráfico 10 (10 °C e 20 °C), uma diferença percentual de 52,4%.

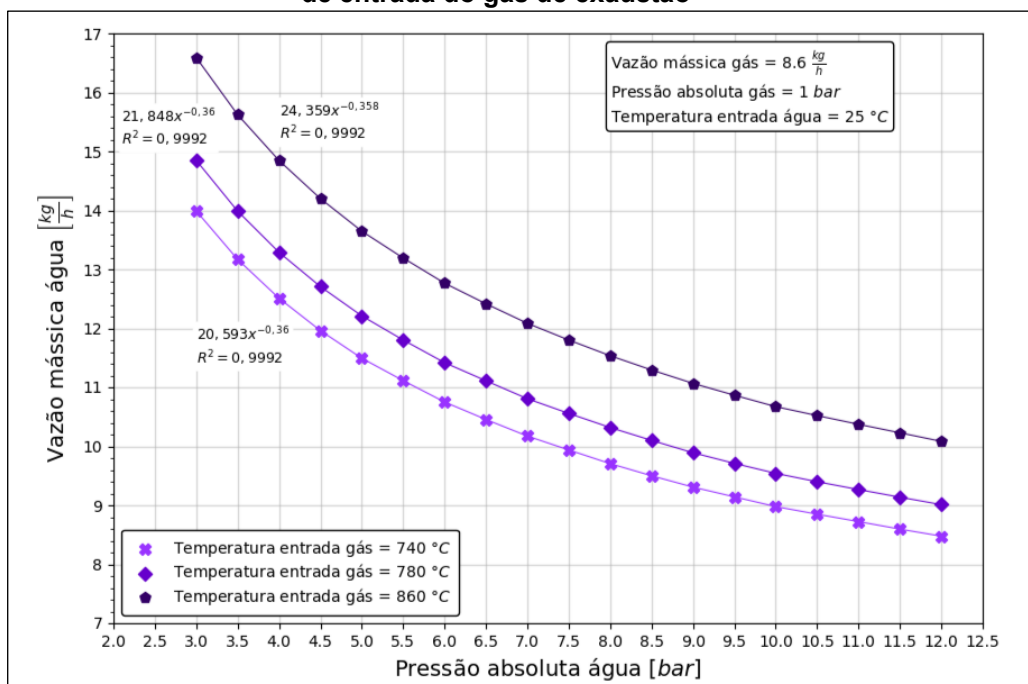
Gráfico 12 - Temperatura de saída do gás de exaustão em função da pressão absoluta da água e da temperatura de entrada do gás de exaustão



Fonte: Autoria própria

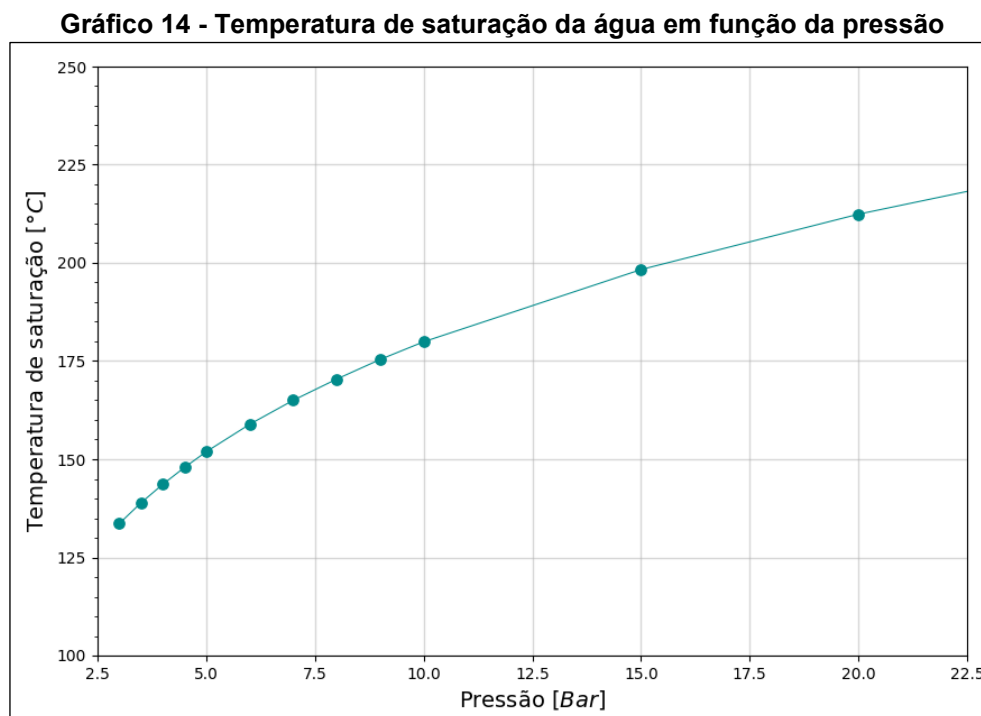
O Gráfico 13 ilustra a vazão mássica em função da pressão absoluta da água. Pode-se observar que há uma queda da vazão conforme o aumento da pressão. Tal comportamento é devido ao mesmo mecanismo presente no Gráfico 11.

Gráfico 13 - Vazão mássica de água em função da pressão absoluta da água e da temperatura de entrada do gás de exaustão



Fonte: Autoria própria

Nos Gráficos 11 e 13, as curvas refletem a variação não-linear da temperatura de saturação da água em relação à pressão, ilustrada no Gráfico 14.



Fonte: Autoria própria

4.7 Correlações matemáticas

Através dos resultados foi possível obter as correlações matemáticas de cada curva e os coeficientes de determinação (R^2). Analisando as correlações para a determinação da temperatura de saída do gás de exaustão, as curvas com variáveis de temperatura de entrada da água e do gás de exaustão geraram correlações lineares e com valores de R^2 entre 0,9992 e 0,9998. Para a variação da pressão absoluta da água, as correlações geradas foram polinomiais e potenciais, e valores de R^2 variando muito entre 0,998 e 0,9992.

É possível ainda notar que as curvas de vazão mássica por temperatura de entrada da água geram correlações polinomiais e, para as curvas de vazão mássica por temperatura de entrada do gás, são obtidas correlações lineares. Em ambos os casos, os valores de R^2 são iguais a 1. Para os últimos gráficos (vazão mássica em função da pressão absoluta da água), é possível notar que há correlações potenciais e os valores de R^2 se encontram entre 0,9988 e 0,9996.

5 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

Os resultados das simulações demonstram que a variável temperatura de entrada do gás de exaustão exerce uma influência maior na determinação da temperatura de saída do gás do que os parâmetros de temperatura de entrada e pressão absoluta da água no sistema.

Em contrapartida, a pressão absoluta da água recebida pelo economizador tem importante influência na determinação da vazão mássica do líquido. Essa influência é maior do que se comparada à produzida pela variação da temperatura de entrada da água.

Dos resultados obtidos foi possível encontrar correlações matemáticas que descrevem cada curva nos gráficos. Desta forma, pode-se estimar valores de temperatura de saída do gás de exaustão e vazão mássica da água para determinado modo de operação do economizador.

Todavia, ainda que seja necessário validar os resultados obtidos das simulações através de dados experimentais, o código e o método abordados nesse trabalho podem contribuir como base para um futuro programa de controle ou automação da caldeira de leito fluidizado circulante presente no Laboratório de Processos Termoquímicos da UTFPR-Campus Ponta Grossa.

Por fim, como sugestão para trabalhos futuros associados ao tema deste estudo, sugere-se:

- Realizar estudos experimentais para validação dos resultados adquiridos através desse trabalho. Desta forma, pode-se aprimorar o modelo desenvolvido;
- Realizar um estudo mais abrangente de operação do equipamento através da variação de suas dimensões, como por exemplo, combinações de diâmetros e comprimento diferentes do equipamento;
- Realizar um estudo comparativo entre os diferentes tipos de trocadores de calor que o economizador pode possuir dentro do sistema de caldeira de leito fluidizado circulante.

REFERÊNCIAS

- BASU, Prabir. **Combustion and gasification in fluidized beds**. CRC press, 2006.
- DAVIDSON, J. F. Circulating fluidised bed hydrodynamics, **Powder technology**, v. 113, n. 3, p. 249-260, 2000.
- SINGH, Ravi Inder; BRINK, Anders; HUPA, Mikko. CFD modeling to study fluidized bed combustion and gasification. **Applied Thermal Engineering**, v. 52, n. 2, p. 585-614, 2013.
- NOSRATI, Masoud. Python: An appropriate language for real world programming. **World Applied Programming**, v. 1, n. 2, p. 110-117, 2011.
- The Python Programming Language. **TIOBE**. 06 out, 2021. Disponível em: <<https://www.tiobe.com/tiobe-index/python/>>. Acesso em: 24 out, 2021.
- KUNII, Daizo; LEVENSPIEL, Octave. **Fluidization engineering**. 2. Ed. Butterworth-Heinemann, 1991.
- GRACE, John R.; BI, Xiaotao; ELLIS, Naoko. **Essentials of fluidization technology**. John Wiley & Sons, 2020.
- GELDART, Derek. **Gas fluidization technology**. John Wiley & Sons, 1986.
- BERGMAN, Theodore L. et al. **Fundamentos de Transferência de Calor e de Massa**. 8 ed. Rio de Janeiro: Grupo GEN-LTC, 2019. 9788521636656. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788521636656/>. Acesso em: 12 nov, 2021.
- ÇENGEL, Yunus. A. **Transferência de Calor e Massa: Uma Abordagem Prática**. 4. ed. Porto Alegre: MC Grall Hill, 2012. 9788580551280. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788580551280/>. Acesso em: 12 nov, 2021.
- KIUSALAAS, Jaan. **Numerical methods in engineering with Python**. New York: Cambridge university press, 2005.
- NUDDS, Donald. Conversion of programs between computers: Interpreters, simulators and compilers. **Journal of Computer and System Sciences**, v. 7, n. 6, p. 597-614, 1973.
- KNUTH, Donald Ervin. Fundamental algorithms. **The art of computer programming**, v. 1, p. 261-268, 1997.
- KREITH, Frank; MANGLIK, Raj M.; BOHN, Mark S. **Princípios de transferência de calor - Tradução da 7ª edição norte-americana**. São Paulo: Cengage Learning Brasil, 2015. 9788522122028. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788522122028/>. Acesso em: 04 abr, 2022.

NGUYEN, Huy H.; MARTIN, Michael A. An Interpolation Method for Obtaining Thermodynamic Properties Near Saturated Liquid and Saturated Vapor Lines. In: **52nd JANNAF Propulsion Meeting**. 2003. Disponível em: <https://ntrs.nasa.gov/api/citations/20040075672/downloads/20040075672.pdf>. Acesso em: 04 abr, 2022.

MORAN, Michael J. et al. **Princípios de Termodinâmica para Engenharia**, 8ª edição. Rio de Janeiro: Grupo GEN, 2018. 9788521634904. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788521634904/>. Acesso em: 03 mai. 2022.

APÊNDICE A - Código em Python completo para as simulações

```

1  ##-----Importando bibliotecas-----
   -----##
2  from math import log, pow, pi
3  import pandas as pd
4  import numpy as np
9
10 ##-----Tabelas-----
   ##
11     #Propriedades termofísicas água
12     agua_array = [ ["Temperatura", "Pressão", "Volume específico", " ", "Calor
de Vaporização", "Calor específico", " ", "Viscosidade dinâmica", "
", "Condutividade térmica", " ", "Numero de Prandtl", " ", "Tensão
superficial", "Coeficiente de expansão"],
13     ["T", "p", "vf*10^3", "vg", "hfg", "cpf", "cpg", "uf*10^6", "ug*10^6", "kf
*10^3", "kg*10^3", "Prf", "Prg", "Sigmaf*10^3", "Betaf*10^6"],
14     ["[K]", "[bars]", "[m³/kg]", "[m³/kg]", "[kJ/kg]", "[kJ/(kg*K)]", "[kJ/
(kg*K)]", "[N*s/m²]", "[N*s/m²]", "[W/(m*K)]", "[W/(m*K)]", " ", "
", "[N/m]", "[K^-1]"],
15     [273.15, 0.00611, 1.000, 206.3, 2502, 4.217, 1.854, 1750, 8.02, 569, 18.20,
12.99, 0.815, 75.5, -68.05],
16     [275, 0.00697, 1.000, 181.7, 2497, 4.211, 1.855, 1652, 8.09, 574, 18.30, 12.
22, 0.817, 75.3, -32.74],
17     [280, 0.00990, 1.000, 130.4, 2485, 4.198, 1.858, 1422, 8.29, 582, 18.60, 10.
26, 0.825, 74.8, 46.04],
18     [285, 0.01387, 1.000, 99.4, 2473, 4.189, 1.861, 1225, 8.49, 590, 18.90, 8.81
, 0.833, 74.3, 114.1],
19     [290, 0.01917, 1.001, 69.7, 2461, 4.184, 1.864, 1080, 8.69, 598, 19.30, 7.56
, 0.841, 73.7, 174.0],
20     [295, 0.02617, 1.002, 51.94, 2449, 4.181, 1.868, 959, 8.89, 606, 19.50, 6.62
, 0.849, 72.7, 227.5],
21     [300, 0.03531, 1.003, 39.13, 2438, 4.179, 1.872, 855, 9.09, 613, 19.60, 5.83
, 0.857, 71.7, 276.1],
22     [305, 0.04712, 1.005, 29.74, 2426, 4.178, 1.877, 769, 9.29, 620, 20.10, 5.20
, 0.865, 70.9, 320.6],
23     [310, 0.06221, 1.007, 22.93, 2414, 4.178, 1.882, 695, 9.49, 628, 20.40, 4.62
, 0.873, 70.00, 361.9],
24     [315, 0.08132, 1.009, 17.82, 2402, 4.179, 1.888, 631, 9.69, 634, 20.70, 4.16
, 0.883, 69.2, 400.4],
25     [320, 0.10530, 1.011, 13.98, 2390, 4.180, 1.895, 577, 9.89, 640, 21.00, 3.77
, 0.894, 68.3, 436.7],
26     [325, 0.13510, 1.013, 11.06, 2378, 4.182, 1.903, 528, 10.09, 645, 21.30, 3.4
2, 0.901, 67.5, 471.2],
27     [330, 0.17190, 1.016, 8.82, 2366, 4.184, 1.911, 489, 10.29, 650, 21.70, 3.15
, 0.908, 66.6, 504.0],
28     [335, 0.21670, 1.018, 7.09, 2354, 4.186, 1.920, 453, 10.49, 656, 22.00, 2.88
, 0.916, 65.8, 535.5],
29     [340, 0.27130, 1.021, 5.74, 2342, 4.188, 1.930, 420, 10.69, 660, 22.30, 2.66
, 0.925, 64.9, 566.0],
30     [345, 0.33720, 1.024, 4.683, 2329, 4.191, 1.941, 389, 10.89, 664, 22.60, 2.4
5, 0.933, 64.1, 595.4],
31     [350, 0.41630, 1.027, 3.846, 2317, 4.195, 1.954, 365, 11.09, 668, 23.00, 2.2
9, 0.942, 63.2, 624.2],

```


32	[355,0.51000,1.03,3.180,2304,4.199,1.968,343,11.29,671,23.30,2.14,0.951,62.3,652.3],
33	[360,0.62090,1.034,2.645,2291,4.203,1.983,324,11.49,674,23.70,2.02,0.960,61.4,697.9],
34	[365,0.75140,1.038,2.212,2278,4.209,1.999,306,11.69,677,24.10,1.91,0.969,60.5,707.1],
35	[370,0.90400,1.041,1.861,2265,4.214,2.017,289,11.89,679,24.50,1.80,0.978,59.5,728.7],
36	[373.15,1.01330,1.044,1.679,2257,4.217,2.029,279,12.02,680,24.80,1.76,0.984,58.9,750.1],
37	[375,1.08150,1.045,1.574,2252,4.220,2.036,274,12.09,681,24.90,1.70,0.987,58.6,761],
38	[380,1.28690,1.049,1.337,2239,4.226,2.057,260,12.29,683,25.40,1.61,0.999,57.6,788],
39	[385,1.52330,1.053,1.142,2225,4.232,2.080,248,12.49,685,25.80,1.53,1.004,56.6,814],
40	[390,1.79400,1.058,0.980,2212,4.239,2.104,237,12.69,686,26.30,1.47,1.013,55.6,841],
41	[400,2.45500,1.067,0.731,2183,4.256,2.158,217,13.05,688,27.20,1.34,1.033,53.6,896],
42	[410,0.33020,1.077,0.553,2153,4.278,2.221,200,13.42,688,28.20,1.24,1.054,51.5,952],
43	[420,0.43700,1.088,0.425,2123,4.302,2.291,185,13.79,688,29.80,1.12,1.075,49.4,1010],
44	[430,0.56990,1.099,0.331,2001,4.331,2.369,173,14.14,685,30.40,1.09,1.100,47.2," "],
45	[440,7.33300,1.11,0.261,2059,4.36,2.46,162,14.5,682,31.70,1.04,1.12,45.1," "],
46	[450,9.31900,1.123,0.208,2024,4.4,2.56,152,14.85,678,33.10,0.99,1.14,42.9," "],
47	[460,11.71000,1.137,0.167,1989,4.44,2.68,143,15.19,673,34.60,0.95,1.17,40.7," "],
48	[470,14.55000,1.152,0.136,1951,4.48,2.79,136,15.54,667,36.30,0.92,1.20,38.5," "],
49	[480,17.90000,1.167,0.111,1912,4.53,2.94,129,15.88,660,38.10,0.89,1.23,36.2," "],
50	[490,21.83000,1.184,0.0922,1870,4.59,3.1,124,16.23,651,40.10,0.87,1.25,33.9," "],
51	[500,26.40000,1.203,0.0766,1825,4.66,3.27,118,16.59,642,42.30,0.86,1.28,31.6," "],
52	[510,31.66000,1.222,0.0631,1779,4.74,3.47,113,16.95,631,44.70,0.85,1.31,29.3," "],
53	[520,37.70000,1.244,0.0525,1730,4.84,3.70,108,17.33,621,47.50,0.84,1.35,26.9," "],
54	[530,44.58000,1.268,0.0445,1679,4.95,3.96,104,17.72,608,50.60,0.85,1.39,24.5," "],
55	[540,52.38000,1.294,0.0375,1622,5.08,4.27,101,18.1,594,54.00,0.86,1.43,22.1," "],
56	[550,61.19000,1.323,0.0000,1564,5.24,4.64,97,18.6,580,58.30,0.87,1.47,19.7," "],
57	[560,71.08000,1.355,0.0269,1499,5.43,5.09,94,19.1,563,63.70,0.90,1.52,17.3," "],
58	[570,82.16000,1.392,0.0228,1429,5.68,5.67,91,19.7,548,76.70,0.94,1.59,15.0," "],

```

59     [580,94.51000,1.433,0.0193,1353,6.00,6.40,88,20.4,528,76.70,0.99,
60     1.68,12.8," "],
61     [590,108.30000,1.482,0.0163,1274,6.41,7.35,84,21.5,513,84.10,1.05
62     ,1.84,10.5," "],
63     [600,123.50000,1.541,0.0137,1176,7.00,8.75,81,2.27,497,92.90,1.14
64     ,2.15,8.4," "],
65     [610,137.30000,1.612,0.0115,1068,7.85,11.1,77,24.1,467,103,1.3,2.
66     60,6.3," "],
67     [620,159.10000,1.705,0.0094,941,9.15,15.4,72,25.9,444,114,1.52,3.
68     46,4.5," "],
69     [625,169.10000,1.778,0.0085,858,10.6,18.3,70,27,430,121,1.65,4.20
70     ,3.5," "],
71     [630,179.70000,1.856,0.0075,781,12.6,22.1,67,28,412,130,2.0,4.80,
72     2.6," "],
73     [635,190.90000,1.935,0.0066,683,16.4,27.6,64,30,392,141,2.7,6.00,
74     1.5," "],
75     [640,202.70000,2.075,0.0057,560,26,42.0,59,32,367,155,4.2,9.60,0.
76     8," "],
77     [645,215.20000,2.351,0.0045,361,90,"
78     ",54,37,331,178,12,26.00,0.1," "],
79     [647.3,221.20000,3.17,0.0032,0," ","",45,45,238,238," ",""
80     ",0.0," "]]
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

```

97     [3,133.6],
98     [3.5,138.9],
99     [4,143.6],
100    [4.5,147.9],
101    [5,151.9],
102    [6,158.9],
103    [7,165],
104    [8,170.4],
105    [9,175.4],
106    [10,179.9],
107    [15,198.3],
108    [20,212.4],
109    [25,224],
110    [30,233.9],
111    [35,242.6],
112    [40,250.4],
113    [45,257.5],
114    [50,264],
115    [60,275.6],
116    [70,285.9],
117    [80,295.1],
118    [90,303.4],
119    [100,311.1],
120    [110,318.2],
121    [120,324.8],
122    [130,330.9],
123    [140,336.8],
124    [150,342.2],
125    [160,347.4],
126    [170,352.4],
127    [180,357.1],
128    [190,361.5],
129    [200,365.8],
130    [220.9,374.1]]
131 agua_sat_df = pd.DataFrame(agua_sat_array)
132 agua_sat_df = agua_sat_df.loc[3:]
133 agua_sat_df = agua_sat_df.reset_index()
134 agua_sat_df.index = range(0, len(agua_sat_df))
135 agua_sat_df = agua_sat_df.replace(" ",0)
136 agua_sat_df = agua_sat_df.astype(float, errors = 'raise')
137
138     #Propriedades termofísicas ar
139 ar_array = [ ["Temperatura", "Densidade", "Calor
    específico", "Viscosidade dinâmica", "Viscosidade
    cinemática", "Condutividade térmica", "Viscosidade", "Prandtl"],
140     ["T", "p", "cp", "u*10^7", "v*10^6", "k*10^3", "u*10^6", "Pr"],

```

```

141 ["[K]", "[Kg/m³]", "[kJ/(kg*K)]", "[N*s/m²]", "[m²/s]", "[W/(m*K)]", "[
142 m²/s]", " "],
143 [100, 3.5562, 1.032, 71.1, 2.000, 9.34, 2.54, 0.786],
144 [150, 2.3364, 1.012, 103.4, 4.426, 13.8, 5.84, 0.758],
145 [200, 1.7458, 1.007, 132.5, 7.590, 18.1, 10.3, 0.737],
146 [250, 1.3947, 1.006, 159.6, 11.44, 22.3, 15.9, 0.72],
147 [300, 1.1614, 1.007, 184.6, 15.89, 26.3, 22.5, 0.707],
148 [350, 0.995, 1.009, 208.2, 20.92, 30, 29.9, 0.7],
149 [400, 0.8711, 1.014, 230.1, 26.41, 33.8, 38.3, 0.69],
150 [450, 0.774, 1.021, 250.7, 32.39, 37.3, 47.2, 0.686],
151 [500, 0.6964, 1.030, 270.1, 38.79, 40.7, 56.7, 0.684],
152 [550, 0.6329, 1.040, 288.4, 45.57, 43.9, 66.7, 0.683],
153 [600, 0.5804, 1.051, 305.8, 52.69, 46.9, 76.9, 0.685],
154 [650, 0.5356, 1.063, 322.5, 60.21, 49.7, 87.3, 0.69],
155 [700, 0.4975, 1.075, 338.8, 68.10, 52.4, 98, 0.695],
156 [750, 0.4643, 1.087, 354.6, 76.37, 54.9, 109, 0.702],
157 [800, 0.4354, 1.099, 369.8, 84.93, 57.3, 120, 0.709],
158 [850, 0.4097, 1.110, 384.3, 93.80, 59.6, 131, 0.716],
159 [900, 0.3868, 1.121, 398.1, 102.9, 62, 143, 0.72],
160 [950, 0.3666, 1.131, 411.3, 112.2, 64.3, 155, 0.723],
161 [1000, 0.3482, 1.141, 424.4, 121.9, 66.7, 168, 0.726],
162 [1100, 0.3166, 1.159, 449, 141.8, 71.5, 195, 0.728],
163 [1200, 0.2902, 1.175, 473, 162.9, 76.3, 224, 0.728],
164 [1300, 0.2679, 1.189, 496, 185.1, 82, 257, 0.719],
165 [1400, 0.2488, 1.207, 530, 213, 91, 303, 0.703],
166 [1500, 0.2322, 1.230, 557, 240, 100, 350, 0.685],
167 [1600, 0.2177, 1.248, 584, 268, 106, 390, 0.688],
168 [1700, 0.2049, 1.267, 611, 298, 113, 435, 0.685],
169 [1800, 0.1935, 1.286, 637, 329, 120, 482, 0.683],
170 [1900, 0.1833, 1.307, 663, 362, 128, 534, 0.677],
171 [2000, 0.1741, 1.337, 689, 396, 137, 589, 0.672],
172 [2100, 0.1658, 1.372, 715, 431, 147, 646, 0.667],
173 [2200, 0.1582, 1.417, 740, 468, 160, 714, 0.655],
174 [2300, 0.1513, 1.478, 766, 506, 175, 783, 0.647],
175 [2400, 0.1448, 1.558, 792, 547, 196, 869, 0.63],
176 [2500, 0.1389, 1.665, 818, 589, 222, 960, 0.613],
177 [3000, 0.1135, 2.726, 955, 841, 486, 1570, 0.536]]
177 ar_df = pd.DataFrame(ar_array)
178 ar_df = ar_df.loc[3:]
179 ar_df = ar_df.reset_index()
180 ar_df.index = range(0, len(ar_df))
181 ar_df = ar_df.replace(" ", 0)
182 ar_df = ar_df.astype(float, errors = 'raise')
183
184 #Números de Nusselt região Anular e taxa de transferência de
185 calor constante
185 Nu_array = [["Razão diâmetros", "Nusselt"],

```

```

186     [0,0],
187     [0.05,17.81],
188     [0.1,11.91],
189     [0.2,8.499],
190     [0.4,6.583],
191     [0.6,5.912],
192     [0.8,5.58],
193     [1,5.385]]
194 Nu_df = pd.DataFrame(Nu_array)
195 Nu_df = Nu_df.loc[1:]
196 Nu_df = Nu_df.reset_index()
197 Nu_df.index = range(0, len(Nu_df))
198 Nu_df = Nu_df.replace(" ",0)
199 Nu_df = Nu_df.astype(float, errors = 'raise')
200
201 ##-----Variáveis gerais-----
202     ##
203     #Listas de operação de tipo de fluido, estado e escoamento
204 tipo_fluido = ["Agua", "Ar"]
205 escoamento = ["Interno-Circular","Interno-Anelar"]
206
207     #Definindo array que serão armazenados os resultados
208 resultados_array = []
209
210     #Definindo array para método iterativo
211 n_m_1,n_temp_4 = ([] for i in range(2))
212
213     #Nome das abas dos arquivos Excel
214 nome_tab =
215     [['Temp_1_1','Temp_1_2','Temp_1_3','Temp_1_4','Temp_1_5','Temp_1_6'],
216      ['Temp_3_1','Temp_3_2','Temp_3_3','Temp_3_4','Temp_3_5','Temp
217      _3_6'],
218      ['Press_1_1','Press_1_2','Press_1_3','Press_1_4','Press_1_5',
219      'Press_1_6']]
220
221     #Nome dos arquivos Excel para cada simulação em função da
222     #variável de entrada
223 nome_result =
224     ["Resultados_1.xlsx","Resultados_2.xlsx","Resultados_3.xlsx"]
225
226     #Definindo dicionário das colunas
227 dict_col = {0:'Temperatura entrada água',
228             1:'Temperatura saída água',
229             2:'Temperatura entrada ar',
230             3:'Temperatura saída ar',
231             4:'Vazão água',
232             5:'Vazão do ar',
233             6:'Delta_TmL',

```

```

228         7:'Reynolds ar',
229         8:'Reynolds água',
230         9:'Nusselt ar',
231        10:'Nusselt água',
232        11:'Coef. convectivo ar',
233        12:'Coef. convectivo água',
234        13:'Calor ar',
235        14:'Calor água',
236        15:'Calor coef. global',
237        16:'Pressão ar',
238        17:'Pressão água'}
239
240 #Definindo dicionário das unidades
241 dict_unity = {'Temperatura entrada água':['[°C]'],
242             'Temperatura saída água':['[°C]'],
243             'Temperatura entrada ar':['[°C]'],
244             'Temperatura saída ar':['[°C]'],
245             'Vazão água':['[kg/h]'],
246             'Vazão do ar':['[kg/h]'],
247             'Delta_Tml':['[ ]'],
248             'Reynolds ar':['[ ]'],
249             'Reynolds água':['[ ]'],
250             'Nusselt ar':['[ ]'],
251             'Nusselt água':['[ ]'],
252             'Coef. convectivo ar':['[W/m²*K]'],
253             'Coef. convectivo água':['[W/m²*K]'],
254             'Calor ar':['[W]'],
255             'Calor água':['[W]'],
256             'Calor coef. global':['[W]'],
257             'Pressão ar':['[bar]'],
258             'Pressão água':['[bar]'])}
259
260 ##-----Variáveis dimensionais-----
261 -----##
262     #Diâmetros do tubo interno
263     de_i = 33.40*(pow(10,-3))
264     esp_i = 3.38*(pow(10,-3))
265     di_i = de_i - 2*(esp_i)      #26.64 mm
266
267     #Diâmetros do tubo externo
268     de_e = 48.30*(pow(10,-3))
269     esp_e = 3.38*(pow(10,-3))
270     di_e = de_e - 2*(esp_e)
271
272     #Comprimento efetivo em [m]
273     lef_i = 2400*(pow(10,-3))

```

```

273
274     #Razão entre diâmetros interno e externo da região anular
275 raz_diam = de_i/di_e
276
277     #Diâmetro hidráulico da região anular
278 d_h_e = di_e - de_i
279     #Área de troca interno anular
280 a_troca = (pi*de_i*lef_i)
281     #Área de troca interno gás
282 a_troca_q = (pi*di_i*lef_i)
283     #Área de troca interno água
284 a_troca_f = a_troca
285     #Área transversal gás
286 a_transv_q = (pi*pow(di_i,2)/4)
287     #Área transversal água
288 a_transv_f = (pi*(pow(di_e,2)-pow(de_i,2))/4)
289
290 ##-----Funções-----
291 ##
292 #Função para criar lista com intervalo de valores (aceita float como
293 #passo)
294 """ Detalhes dos parâmetros:
295     inicio = Valor inicial do intervalo
296     fim = Valor final do intervalo
297     passo = Valor do passo do intervalo (n deve ser 0)
298     n = Quantidade de divisões do intervalo
299 """
300 def list_range(inicio,fim,passo,n):
301     i = 0
302     list = []
303
304     if n != 0:
305         end = n
306         passo = (fim - inicio) / (n)
307     elif n == 0:
308         end = (fim - inicio) / passo
309     else:
310         return 0
311
312     while i <= end:
313         s = (inicio) + (passo * i)
314         list.append(s)
315         i += 1
316     return list
317 #Função para busca e interpolação:

```

```

318 """ Detalhes dos parâmetros:
319     df = Dataframe para realizar as buscas
320     valor = Valor a ser encontrado ou interpolado
321 """
322 def interpolacao(df,valor):
323     for i in range(0,len(df.index)):
324         valor_teste = df.loc[i,0]
325         #Teste
326         #Se o valor desejado se encontra na tabela, retorna a
linha referente a ele
327         if valor_teste == valor:
328             x0 = 0
329             x1 = 0
330             y = pd.DataFrame(df.loc[i]).T
331             break
332         #Se o valor não for encontrado, realiza a interpolação
333         elif valor_teste > valor:
334             x1 = valor_teste
335             y1 = pd.DataFrame(df.loc[i]).T
336             y = pd.DataFrame(y0.loc[j] + (y1.loc[i] - y0.loc[j]) *
((valor - x0) / (x1 - x0))).T
337             break
338         #Definindo valores para a sequencia do loop
339         x0 = valor_teste
340         y0 = pd.DataFrame(df.loc[i]).T
341         j = i
342         #Zera o índice da linha
343         y = y.set_axis([0],axis=0)
344         return y
345
346 #Função para encontrar a temperatura de saturação em função da
pressão de operação da água (Interpolação)
347 """ Detalhes dos parâmetros:
348     pressao = Valor em bar da pressão que se quer retornar a
temperatura de saturação
349 """
350 def temp_sat(pressao):
351     df = agua_sat_df
352     y = interpolacao(df,pressao)
353     y = y.drop(["index"],axis=1)
354     y = y.loc[0].tolist()
355     return y[1]
356
357 #Função para os cálculos necessários para encontrar resultados
358 """ Detalhes dos parâmetros:
359     vazq_entra = Vazão de entrada do fluido quente em [kg/s]
360     vazf_entra = Vazão de entrada do fluido frio em [kg/s]

```



```

361     Tq_entra = Temperatura de entrada do fluido quente em [°C]
362     Tq_sai = Temperatura de saída do fluido quente em [°C]
363     Tf_entra = Temperatura de entrada do fluido frio em [°C]
364     Tf_sai = Temperatura de saída do fluido frio em [°C]
365     p_3 = Pressão absoluta do fluido quente [bar]
366     p_1 = Pressão absoluta do fluido frio [bar]
367     """
368     def
369         calculo(vazq_entra,vazf_entra,Tq_entra,Tq_sai,Tf_entra,Tf_sai,p_q,p_f
370         ):
371             #Função para encontrar as propriedades termofísicas dos fluidos:
372             """ Detalhes dos parâmetros:
373                 tipo_fluido = Qual fluido de interesse (CONSULTAR VARIÁVEL
374                 "tipo_fluido" PARA VER OPÇÕES)
375                 temp_med = Temperatura média/fluido a se retornar as
376                 propriedades termofísicas em [K]
377             """
378             #Ordem da lista propriedades termofísicas retornado:
379             #[densidade,calor específico,viscosidade absoluta,viscosidade
380             cinemática,condutividade térmica,Prandtl,difusividade térmica]
381             def prop_term(tipo_fluido,temp_med):
382                 p = list()
383                 u = 0
384                 #Seleção da tabela
385                 if tipo_fluido == "Agua":
386                     df = agua_df
387
388                 elif tipo_fluido == "Ar":
389                     df = ar_df
390                 #Procura e interpolação (quando necessário)
391                 y = interpolacao(df,temp_med)
392                 #Organização e conversão das propriedades para o SI
393                 if tipo_fluido == "Agua":
394                     y = y.drop(["index",0,1,3,4,6,8,10,12,13,14], axis = 1)
395                     y = y.set_axis([0,1,2,3,4], axis = 1)
396                     for i in range(0,len(y.columns)):
397                         p.append(y.at[0,i])
398                         p.insert(3,0)
399                         p.insert(7,0)
400                         p[0] = 1 / (p[0] * pow(10,-3))
401                         p[1] = p[1] * pow(10,3)
402                         p[2] = p[2] * pow(10,-6)
403                         p[4] = p[4] * pow(10,-3)
404
405                 elif tipo_fluido == "Ar":
406                     y = y.drop(["index",0],axis = 1)
407                     y = y.set_axis([0,1,2,3,4,5,6],axis = 1)

```

```

403         for i in range(0,len(y.columns)):
404             p.append(y.at[0,i])
405             u = p[6]
406             p.remove(u)
407             p.insert(5,u)
408             p[1] = p[1] * pow(10,3)
409             p[2] = p[2] * pow(10,-7)
410             p[3] = p[3] * pow(10,-6)
411             p[4] = p[4] * pow(10,-3)
412             p[6] = p[6] * pow(10,-6)
413         return p
414
415     #Cálculo MDLT
416     """ Detalhes dos parâmetros:
417         Tq_entra = Temperatura de entrada do fluido quente em [°C]
418         Tq_sai = Temperatura de saída do fluido quente em [°C]
419         Tf_entra = Temperatura de entrada do fluido frio em [°C]
420         Tf_sai = Temperatura de saída do fluido frio em [°C]
421     """
422     def MDLT(Tq_entra,Tq_sai,Tf_entra,Tf_sai):
423         #Configuração contracorrente
424         delta_T1 = Tq_entra - Tf_sai
425         delta_T2 = Tq_sai - Tf_entra
426         delta_Tml = (delta_T2 - delta_T1) / log(delta_T2 / delta_T1)
427         return delta_Tml
428
429     #Cálculo Reynolds
430     """ Detalhes dos parâmetros:
431         densidade = Densidade mássica do fluido em [kg/m³]
432         visc_abs = Viscosidade absoluta do fluido em [N*s/m²]
433         diametro = Diâmetro da seção transversal ou diâmetro
434         hidáulico do escoamento em [m]
435         a_transv = Área transversal do escoamento em [m²]
436         taxa_massica = Vazão mássica do escoamento em [kg/s]
437         vel_media = Velocidade média do escoamento em [m/s] (se 0, a
438         função realiza o cálculo do mesmo através dos parâmetros anteriores)
439     """
440     def reynolds(densidade, visc_abs, diametro, a_transv,
441                 taxa_massica, vel_media):
442         #Calcula a velocidade média do fluido
443         if vel_media == 0:
444             vel_media = (taxa_massica) / (densidade * a_transv)
445         #Cálculo do numero de reynolds
446         Re = (densidade * vel_media * diametro) / (visc_abs)
447         return Re
448
449     #Cálculo Nusselt (Laminar ou Turbulento)

```

```

447     """ Detalhes dos parâmetros:
448         escoamento = Região de escoamento (CONSULTAR VARIÁVEL
"escoamento" PARA VER OPÇÕES)
449         Re = Número de Reynolds do escoamento
450         Pr = Número de Prandtl do fluido em escoamento
451         raz_diam = Razão dos diâmetros interno e externo da seção
anular
452     """
453     def nusselt(escoamento,Re,Pr,raz_diam):
454         #Escoamento Laminar
455         if Re <= 2300.00:
456             if escoamento == "Interno-Circular":
457                 Nud = 4.36
458
459             elif escoamento == "Interno-Anelar":
460                 Nud = float(interpolacao(Nu_df,raz_diam).iloc[0][1])
461
462         #Escoamento Turbulento
463         elif Re > 2300.00:
464             #Correlação Gnieslinski
465             #0.5<=Pr<=200 \ 3000<=Re<=5000000
466             f = pow((0.790*np.log(Re)-1.64),-2)
467             Nud = ((f/8)*(Re-
1000)*Pr)/(1+12.7*pow((f/8),(1/2))*(pow(Pr,(2/3))-1))
468             return Nud
469
470         #Cálculo coeficiente convectivo
471         """ Detalhes dos parâmetros:
472             Nud = Número de Nusselt do escoamento
473             condutividade_termica = Condutividade térmica do fluido em
[W/(m*K)]
474             diametro = Diâmetro da seção transversal ou diâmetro
hidráulico do escoamento em [m]
475         """
476         def coef_convect(Nud,condutividade_termica,diametro):
477             h = Nud*condutividade_termica/diametro
478             return h
479
480         #Cálculo resistência Convecção + resistencia por incrustação
481         """ Detalhes dos parâmetros:
482             tipo_fluido = Qual fluido do escoamento (CONSULTAR VARIÁVEL
"tipo_fluido" PARA VER OPÇÕES)
483             h = Coeficiente convectivo médio do escoamento em [W/(m²*K)]
484             a_troca = Área de troca térmica em [m²]
485         """
486         def res_conv(tipo_fluido,h,a_troca):
487             if tipo_fluido == "Agua":
488                 R_f = 0.0001

```

```

489
490     elif tipo_fluido == "Ar":
491         R_f = 0.0004
492         res_conv = R_f/a_troca + 1/(h*a_troca)
493         return res_conv
494
495     #Cálculo resistencia Condução
496     """ Detalhes dos parâmetros:
497         k_mat = Condutividade térmica do material em [W/(m*K)]
498         di = Diâmetro interno do material em [m]
499         de = Diâmetro externo do material em [m]
500         L = Comprimento de troca térmica em [m]
501     """
502     def res_cond(k_mat,di,de,L):
503         res_cond = log(de/di)/(2*pi*k_mat*L)
504         return res_cond
505
506     #Definição de variáveis
507         #Array onde ficarão os resultados da simulação
508     resultados = []
509
510     #Calculando temperaturas médias dos fluidos e o deltaTml
511         #Temperaturas médias dos fluidos
512     temp_med_i = ((Tq_entra+Tq_sai)/2)+273
513     temp_med_e = ((Tf_entra+Tf_sai)/2)+273
514         #deltaTml
515     delta_Tml = MDLT(Tq_entra,Tq_sai,Tf_entra,Tf_sai)
516
517     #Propriedades termofísicas:
518         #Gás de exaustão na temperatura média fluido (propriedades do
519     ar)
520     p_i = prop_term(tipo_fluido[1],temp_med_i)
521         #Água na temperatura média fluido
522     p_e = prop_term(tipo_fluido[0],temp_med_e)
523         #Material na temperatura de superfície
524     k_mat = 15.5
525
526     #Número de Reynolds:
527         #Tubo circular
528     Re_i = reynolds(p_i[0],p_i[2],di_i,a_transv_q,vazq_entra,0)
529         #Anular
530     Re_e = reynolds(p_e[0],p_e[2],d_h_e,a_transv_f,vazf_entra,0)
531
532     #Número de Nusselt:
533         #Tubo circular
534     Nud_i = nusselt(escoamento[0],Re_i,p_i[5],0)

```

```

534     #Anular
535     Nud_e = nusselt(escoamento[1],Re_e,p_e[5],raz_diam)
536
537     #Coeficiente convectivo
538     #Tubo circular
539     h_i = coef_convect(Nud_i,p_i[4],di_i)
540     #Anular
541     h_e = coef_convect(Nud_e,p_e[4],d_h_e)
542
543     #Resistências:
544     #Convectiva tubo circular
545     R_conv_i = res_conv(tipo_fluido[1],h_i,a_troca_q)
546     #Condução do material do tubo
547     R_cond = res_cond(k_mat,di_i,de_i,lef_i)
548     #Convectiva anular
549     R_conv_e = res_conv(tipo_fluido[0],h_e,a_troca_f)
550     #Total (configuração em série)
551     Rtot = R_conv_i + R_cond + R_conv_e
552
553     #Taxa de transferência de calor através das resistências total
554     q = (1/Rtot)*delta_Tml
555
556     #Taxa de transferência de calor através das vazões mássicas e
    capacidade térmicas
557     #Tubo circular
558     q_i = vazq_entra*p_i[1]*abs(Tq_entra-Tq_sai)
559     #Anular
560     q_e = vazf_entra*p_e[1]*abs(Tf_sai-Tf_entra)
561
562     #Criando lista e retornando o vetor com os resultados e as
    propriedades da região anular
563     resultados.extend([Tf_entra,Tf_sai,Tq_entra,Tq_sai,vazf_entra*360
    0,vazq_entra*3600,delta_Tml,Re_i,Re_e,Nud_i,Nud_e,h_i,h_e,q_i,q_e,q,p
    _q,p_f])
564     return resultados, p_e
565
566     ##-----SIMULAÇÃO-----
    --##
567
568     #Definindo intervalos de operação:
569     #Temperatura de entrada da água em [°C]
570     temp_1_range = list_range(10,30,1,0)
571     #Temperatura de entrada do gás em [°C]
572     temp_3_range = list_range(740,860,5,0)
573     #Pressão de operação da água em [bar]
574     p_1_range = list_range(3,12,0.5,0)
575     #Variação de estados de entrada para simulação

```

```
576     #Temperatura de entrada da água em [°C]
577 temp1_lim = [10,20,35]
578     #Temperatura de entrada do gás em [°C]
579 temp3_lim = [740,780,860]
580     #Pressão de operação da água em [bar]
581 p1_lim = [3,5,12]
582     #Criando lista a ser percorrida para a simulação (interativo)
583 temp_4_range = list_range(250,150,0,200)
584
585 #Laço para as variáveis principais
586 for cond in range(0,3,1):
587     #Criando arquivo Excel para armazenar resultados
588     writer = pd.ExcelWriter(nome_result[cond], engine='xlsxwriter')
589
590     #Variando temperatura de entrada da água
591     if cond == 0:
592         var_prim = temp_1_range
593
594     #Variando temperatura de entrada do gás
595     elif cond == 1:
596         var_prim = temp_3_range
597
598     #Variando pressão da água
599     elif cond == 2:
600         var_prim = p_1_range
601
602     #Laço para as variáveis secundárias
603     for sub_cond in range(0,6,1):
604
605         ##-----Definindo variáveis de estado padrão-----##
606         #Ponto 1 - Alimentação de água
607         temp_1 = 25 #°C
608         p_1 = 5 #bar
609         #Valor para chute inicial da vazão mássica de água
610         m_1 = 14 / (3600) #kg/s
611
612         #Ponto 2 - Saída de água
613         t_sat = 151.9 #°C (Tabela Shapiro p/ P = 5 bar)
614         temp_2 = t_sat - 20 #°C
615         p_2 = p_1 #bar
616         m_2 = m_1 #kg/s
617
618         #Ponto 3 - Alimentação de gás de exaustão
619         temp_3 = 780 #°C
620         p_3 = 1 #bar
621         m_3 = 8.6 / (3600) #kg/s
```

```

622
623     #Ponto 4 - Saída de gás de exaustão
624     temp_4 = 400 #temp_3 #°C (valor arbitrário)
625     p_4 = p_3 #bar
626     m_4 = m_3 #kg/s
627
628     #Limpando array de resultados
629     resultados_array.clear()
630
631     #Laço para a percorrer a lista de variáveis principais
    escolhidas:
632     for j in var_prim:
633         ##-----Modos de operação-----##
634         #Variando temperatura de entrada da água
635         if cond == 0:
636             temp_1 = j
637             #Variando pressão absoluta da água
638             if sub_cond < 3:
639                 p_1 = p1_lim[sub_cond]
640                 t_sat = temp_sat(p_1)
641                 temp_2 = t_sat - 20 #[°C]
642
643             #Variando temperatura de entrada do gás de
    exaustão
644             elif sub_cond >= 3:
645                 temp_3 = temp3_lim[sub_cond-3]
646
647             #Variando temperatura de entrada do gás
648             elif cond == 1:
649                 temp_3 = j
650                 #Variando pressão absoluta da água
651                 if sub_cond < 3:
652                     p_1 = p1_lim[sub_cond]
653                     t_sat = temp_sat(p_1)
654                     temp_2 = t_sat - 20 #[°C]
655
656                 #Variando temperatura de entrada da água
657                 elif sub_cond >= 3:
658                     temp_1 = temp1_lim[sub_cond-3]
659
660                 #Variando pressão da água
661             elif cond == 2:
662                 p_1 = j
663                 t_sat = temp_sat(p_1)
664                 temp_2 = t_sat - 20 #[°C]
665                 #Variando temperatura de entrada da água
666                 if sub_cond < 3:

```

```

667         temp_1 = temp1_lim[sub_cond]
668
669         #Variando temperatura de entrada do gás de
exaustão
670         elif sub_cond >= 3:
671             temp_3 = temp3_lim[sub_cond-3]
672
673         #Laço para a variação da temperatura de saída do gás:
674         for k in range(0,201,1):
675             temp_4 = float(temp_4_range[k])
676             #Realizando os cálculos e colhendo os resultados para
determinada operação
677             resultado, prop_e =
calculo(m_3,m_1,temp_3,temp_4,temp_1,temp_2,p_3,p_1)
678
679             #Condição de 1% de erro entre as taxas de
transferência de calor do coef. global e pelo gás
680             if ((abs(resultado[15]-
resultado[13])/resultado[15])*100<=1):
681                 #Cálculo da vazão mássica da água para a solução
encontrada
682                 m_1_novo = (resultado[15] / (prop_e[1] * (temp_2
- temp_1))) #kg/s
683
684                 #Adicionando em listas os valores de vazão
mássica da água e temperatura de saída do gás, respectivamente:
685                 n_m_1.append(m_1_novo)
686                 n_temp_4.append(temp_4) #°C
687
688             #Reset das variáveis de temperatura e vazão mássica final
encontradas
689             temp_4_f = 0
690             m_1_f = 0
691
692             #Cálculo dos valores de temperatura e vazão mássica
finais (Média simples dos valores encontrados e adicionados nas
listas)
693             temp_4_f = sum(n_temp_4) / len(n_temp_4)
694             m_1_f = (sum(n_m_1) / len(n_m_1))
695
696             #Cálculo final do resultado com temperatura e vazão
mássica finais:
697             resultado, prop_e =
calculo(m_3,m_1_f,temp_3,temp_4_f,temp_1,temp_2,p_3,p_1)
698
699             #Adição do resultado na tabela final:
700             resultados_array.append(resultado)
701
702             #Limpendo listas de valores de soluções encontradas:
703             n_m_1.clear()

```



```

704         n_temp_4.clear()
705
706         #Criando tabela com resultados e adicionando em arquivo Excel
707         resultados_df = pd.DataFrame(resultados_array)
708         resultados_df = resultados_df.rename(columns = dict_col,
inplace = False)
709         df_un = pd.DataFrame(data=dict_unity)
710         resultados_df_f = pd.concat([df_un, resultados_df])
711         resultados_df_f.reset_index(drop=True, inplace=True)
712         resultados_df_f.to_excel(writer, sheet_name =
nome_tab[cond][sub_cond])
713
714         #Salvando arquivo Excel
715         writer.save()
716
717         #Laço para a variação da temperatura de saída do gás:
718         for j in range(0,201,1):
719             temp_4 = float(temp_4_range[j])
720             #Realizando os cálculos e colhendo os resultados para
determinada operação
721             resultado, prop_e =
calculo(m_3,m_1,temp_3,temp_4,temp_1,temp_2,p_3,p_1)
722
723             #Condição de 1% de erro entre as taxas de
transferência de calor do coef. global e pelo gás
724             if (0.99<=resultado[15]/resultado[13]<=1.01):
725                 #Cálculo da vazão mássica da água para a solução
encontrada
726                 m_1_novo = (resultado[15] / (prop_e[1] * (temp_2
- temp_1)))*3600
727                 #print(cond, sub_cond, m_1_novo)
728                 #Adicionando em listas os valores de vazão
mássica da água e temperatura de saída do gás, respectivamente:
729                 n_m_1.append(m_1_novo)
730                 n_temp_4.append(temp_4)
731
732                 #Reset das variáveis de temperatura e vazão mássica final
encontradas
733                 temp_4_f = 0
734                 m_1_f = 0
735
736                 #Cálculo dos valores de temperatura e vazão mássica
finais (Média simples dos valores encontrados e adicionados nas
listas)
737                 temp_4_f = sum(n_temp_4) / len(n_temp_4)
738                 m_1_f = (sum(n_m_1) / len(n_m_1)) / 3600
739

```

```
740         #Cálculo final do resultado com temperatura e vazão
mássica finais:
741         resultado, prop_e =
calculo(m_3,m_1_f,temp_3,temp_4_f,temp_1,temp_2,p_3,p_1)
742
743         #Adição do resultado na tabela final:
744         resultados_array.append(resultado)
745
746         #Reset das listas com valores de todas as soluções
encontradas:
747         n_m_1.clear()
748         n_temp_4.clear()
749         #Criando tabela com resultados e adicionando em arquivo Excel
750         resultados_df = pd.DataFrame(resultados_array)
751         resultados_df = resultados_df.rename(columns = dict_col,
inplace = False)
752         df_un = pd.DataFrame(data=dict_unity)
753         resultados_df_f = pd.concat([df_un, resultados_df])
754         resultados_df_f.reset_index(drop=True, inplace=True)
755         resultados_df_f.to_excel(writer,sheet_name =
nome_tab[cond][sub_cond])
756         #Salvando arquivo Excel
757         writer.save()
```

APÊNDICE B - Código em Python para criação dos gráficos

```

1  ##-----Importando bibliotecas-----
   -----##
2  import matplotlib.pyplot as plt
3  import pandas as pd
4  from matplotlib.ticker import (AutoMinorLocator, MultipleLocator)
5
6  ##-----Definições de listas e parâmetros---
   -----##
7  filename =
8  ['Resultados_1.xlsx', 'Resultados_2.xlsx', 'Resultados_3.xlsx']
9
9  nome_tab =
10 ['Temp_1_1', 'Temp_1_2', 'Temp_1_3', 'Temp_1_4', 'Temp_1_5', 'Temp_1_6']
11 ,
12     ['Temp_3_1', 'Temp_3_2', 'Temp_3_3', 'Temp_3_4', 'Temp_3_5', 'Temp
13     p_3_6'],
14     ['Press_1_1', 'Press_1_2', 'Press_1_3', 'Press_1_4', 'Press_1_5'
15     , 'Press_1_6']]
16
17 x_label = ['Temperatura entrada água ' + r'$\left [ °C \right ]$',
18            'Temperatura entrada gás ' + r'$\left [ °C \right ]$',
19            'Pressão absoluta água ' + r'$\left [ bar \right ]$']
20
21 y_label = ['Temperatura saída gás de exaustão ' + r'$\left [ °C
22            \right ]$',
23            'Vazão mássica água ' + r'$\left [ \frac{kg}{h} \right ]$']
24
25 x_scale = [[9, 31],[730, 870],[2.5,12.5]]
26 y_scale_1 = [[170, 220],[175, 225],[180, 230]]
27 y_scale_2 = [[7, 16],[7, 17],[7, 17]]
28
29 major_tick_x = [1,10,0.5]
30 major_tick_y_1 = [5,5,5]
31 major_tick_y_2 = [1,1,1]
32
33 txt_px_1 =
34 [x_scale[0][0]+major_tick_x[0]*1,x_scale[1][0]+major_tick_x[1]*0.5,x
35 _scale[2][0]+major_tick_x[2]*1]
36 txt_px_2 =
37 [x_scale[0][0]+major_tick_x[0]*1,x_scale[1][0]+major_tick_x[1]*0.5,x
38 _scale[2][0]+major_tick_x[2]*12]
39 eq_e1_px =
40 [[10,27.8,10],[736,856,738],[4.5,11,11],[10,10,27.8],[750,856,739],[
41 4.5,11,11]]
42 eq_e2_px =
43 [[26,10,10],[740,740,740],[3,12.2,4],[10,27,10],[740,855,740],[3,12.
44 2,4]]
45
46 txt_py_1 = [y_scale_1[0][1]-major_tick_y_1[0]*1.5,y_scale_1[1][1]-
47 major_tick_y_1[1]*1.5,y_scale_1[2][1]-major_tick_y_1[2]*1.5]

```

```

35 txt_py_2 = [y_scale_2[0][1]-major_tick_y_2[0]*1.5,y_scale_2[1][1]-
major_tick_y_2[1]*1.5,y_scale_2[2][1]-major_tick_y_2[2]*1.5]
36 eq_e1_py =
[[179,201,199],[178,216,200],[181,203,212],[174,190,211],[176,213,19
8],[181,206,224]]
37 eq_e2_py =
[[14.1,11.5,8.8],[13.2,10.6,7.6],[10.6,8.5,15],[9.3,13,12.8],[9.4,13
.4,13.2],[11.4,8.7,15.1]]
38
39 texto = ['Vazão mássica gás = 8.6 '+r'\frac{kg}{h}$'+'\nPressão
absoluta gás = 1 '+r'$\bar{p}$'+'\nTemperatura entrada gás = 780
'+r'$°C$',
40 'Vazão mássica gás = 8.6 '+r'\frac{kg}{h}$'+'\nPressão
absoluta gás = 1 '+r'$\bar{p}$'+'\nTemperatura entrada água = 25
'+r'$°C$',
41 'Vazão mássica gás = 8.6 '+r'\frac{kg}{h}$'+'\nPressão
absoluta gás = 1 '+r'$\bar{p}$'+'\nTemperatura entrada gás = 780
'+r'$°C$',
42 'Vazão mássica gás = 8.6 '+r'\frac{kg}{h}$'+'\nPressão
absoluta gás = 1 '+r'$\bar{p}$'+'\nPressão absoluta água = 5 '+r'$\bar{p}$',
43 'Vazão mássica gás = 8.6 '+r'\frac{kg}{h}$'+'\nPressão
absoluta gás = 1 '+r'$\bar{p}$'+'\nPressão absoluta água = 5 '+r'$\bar{p}$',
44 'Vazão mássica gás = 8.6 '+r'\frac{kg}{h}$'+'\nPressão
absoluta gás = 1 '+r'$\bar{p}$'+'\nTemperatura entrada água = 25
'+r'$°C$']
45
46 legenda = [['Pressão absoluta água = 3 '+r'$\bar{p}$','Pressão absoluta
água = 5 '+r'$\bar{p}$','Pressão absoluta água = 12 '+r'$\bar{p}$'],
47 ['Pressão absoluta água = 3 '+r'$\bar{p}$','Pressão absoluta
água = 5 '+r'$\bar{p}$','Pressão absoluta água = 12 '+r'$\bar{p}$'],
48 ['Temperatura entrada água = 10 '+r'$°C$','Temperatura
entrada água = 20 '+r'$°C$','Temperatura entrada água = 35
'+r'$°C$'],
49 ['Temperatura entrada gás = 740 '+r'$°C$','Temperatura
entrada gás = 780 '+r'$°C$','Temperatura entrada gás = 860
'+r'$°C$'],
50 ['Temperatura entrada água = 10 '+r'$°C$','Temperatura
entrada água = 20 '+r'$°C$','Temperatura entrada água = 35
'+r'$°C$'],
51 ['Temperatura entrada gás = 740 '+r'$°C$','Temperatura
entrada gás = 780 '+r'$°C$','Temperatura entrada gás = 860
'+r'$°C$']]
52
53 eq_e1 = [[r'${0,5854x+176,78}$'+'\n'+r'${R^2 =
0,9993}$',r'${0,5831x+181,09}$'+'\n'+r'${R^2 =
0,9993}$',r'${0,5727x+189,91}$'+'\n'+r'${R^2 = 0,9995}$'],
54 [r'${0,2267x+14,667}$'+'\n'+r'${R^2 =
0,9998}$',r'${0,2265x+18,989}$'+'\n'+r'${R^2 =
0,9998}$',r'${0,2236x+29,908}$'+'\n'+r'${R^2 = 0,9997}$'],
55 [r'${-0,79x^2+2,559x+190,58}$'+'\n'+r'${R^2 =
0,9987}$',r'${178,38x^{0,0484}}$'+'\n'+r'${R^2 = 0,998}$',r'${-
0,0771x^2+2,5653x+175,9}$'+'\n'+r'${R^2 = 0,9991}$'],

```

```

56      [r'${0,5802x+172,31}$'+'\n'+r'${R^2 =
0,9992}$',r'${0,5831x+181,09}$'+'\n'+r'${R^2 =
0,9993}$',r'${0,5786x+199,40}$'+'\n'+r'${R^2 = 0,9996}$'],
57      [r'${0,2254x+11,202}$'+'\n'+r'${R^2 =
0,9998}$',r'${0,2254x+16,952}$'+'\n'+r'${R^2 =
0,9998}$',r'${0,2259x+25,392}$'+'\n'+r'${R^2 = 0,9998}$'],
58      [r'${-0,0818x^2+2,6326x+175,54}$'+'\n'+r'${R^2 =
0,999}$',r'${-0,0798x^2+2,5718x+184,65}$'+'\n'+r'${R^2 =
0,9984}$',r'${-0,0782x^2+2,5232x+203,13}$'+'\n'+r'${R^2 =
0,9992}$']]
59
60  eq_e2 = [[r'${0,0015x^2+0,0796x+11,952}$'+'\n'+r'${R^2 =
1}$',r'${0,0008x^2+0,0597x+10,19}$'+'\n'+r'${R^2 =
1}$',r'${0,0004x^2+0,0359x+7,8862}$'+'\n'+r'${R^2 = 1}$'],
61      [r'${0,0216x-2,0336}$'+'\n'+r'${R^2 = 1}$',r'${0,0179x-
1,7773}$'+'\n'+r'${R^2 = 1}$',r'${0,0134x-1,4406}$'+'\n'+r'${R^2 =
1}$'],
62      [r'${18,2x^{-0,319}}$'+'\n'+r'${R^2 =
0,9996}$',r'${20,478x^{-0,345}}$'+'\n'+r'${R^2 =
0,9993}$',r'${25,188x^{-0,393}}$'+'\n'+r'${R^2 = 0,9988}$'],
63      [r'${0,0008x^2+0,0559x+9,6072}$'+'\n'+r'${R^2 =
1}$',r'${0,0008x^2+0,0597x+10,19}$'+'\n'+r'${R^2 =
1}$',r'${0,0009x^2+0,0679x+11,361}$'+'\n'+r'${R^2 = 1}$'],
64      [r'${0,0157x-1,4055}$'+'\n'+r'${R^2 = 1}$',r'${0,0171x-
1,6267}$'+'\n'+r'${R^2 = 1}$',r'${0,0197x-2,055}$'+'\n'+r'${R^2 =
1}$'],
65      [r'${20,593x^{-0,36}}$'+'\n'+r'${R^2 =
0,9992}$',r'${21,848x^{-0,36}}$'+'\n'+r'${R^2 =
0,9992}$',r'${24,359x^{-0,358}}$'+'\n'+r'${R^2 = 0,9992}$']]
66
67  leng_loc = [4,4,3]
68
69  size = 40
70  f_size = 10
71  pad = 8
72  alpha = 1
73
74  ##-----Função para importar os resultados-----
-----##
75  def import_df(filename,sheetname):
76      df =
pd.DataFrame(pd.read_excel(r"C:\\Users\\gusta\\OneDrive\\UTFPR\\TCC\\
\\Programas\\GitHub-TCC\\"+filename,sheet_name=sheetname))
77      df = df.loc[1:]
78      df = df.reset_index()
79      df = df.drop(['Unnamed: 0','index'], axis='columns',
inplace=False)
80
81      if filename == 'Resultados_3.xlsx':
82          df = df[['Temperatura entrada água','Temperatura saída
água','Temperatura entrada ar','Pressão água','Temperatura saída
ar','Vazão água','Vazão do ar','Delta_Tml','Reynolds ar','Reynolds

```

```

água', 'Nusselt ar', 'Nusselt água', 'Coef. convectivo ar', 'Coef.
convectivo água', 'Calor ar', 'Calor água', 'Calor coef.
global', 'Pressão ar']]
83     return df
84
85     ##-----Agrupar as colunas e calcular valor
médio da temperatura de saída do gás e da vazão da água-----
-----##
86     def group_1(df,order):
87         df = df.groupby(by=order, as_index=False).mean()
88         return df
89
90     ##-----Escolher qual dado para os gráficos-
-----##
91     modo = int(input('Selecione o modo do gráfico:\n0:Temperatura de
entrada água\n1:Temperatura de entrada gás\n2:Pressão da água\n'))
92
93     ##-----Salvar ou não os gráficos-----
-----##
94     ger_graf = str(input('Deseja salvar o gráfico? [S/N]'))
95
96     ##-----Criando DFs para cada simulação-----
-----##
97     df_1 = import_df(filename[modo],nome_tab[modo][0])
98     df_2 = import_df(filename[modo],nome_tab[modo][1])
99     df_3 = import_df(filename[modo],nome_tab[modo][2])
100    df_4 = import_df(filename[modo],nome_tab[modo][3])
101    df_5 = import_df(filename[modo],nome_tab[modo][4])
102    df_6 = import_df(filename[modo],nome_tab[modo][5])
103
104    #-----Obtendo o index das colunas e
definindo listas de colunas para cada modo-----
----##
105    columns = df_1.columns.tolist()
106    ordem_gb = [[columns[0],columns[1],columns[2]],
107                [columns[2],columns[0],columns[1]],
108                [columns[3],columns[0],columns[1],columns[2]]]
109
110    if modo == 0:
111        col_x = columns[0]
112        col_y1 = columns[3]
113        col_y2 = columns[4]
114    elif modo == 1:
115        col_x = columns[2]
116        col_y1 = columns[3]
117        col_y2 = columns[4]
118    elif modo == 2:
119        col_x = columns[3]
120        col_y1 = columns[4]
121        col_y2 = columns[5]

```

```

122
123  ##-----Agrupando dados e calculando os
      valores médios para os resultados encontrados-----
      -----##
124  df_1 = group_1(df_1,ordem_gb[modo])
125  df_2 = group_1(df_2,ordem_gb[modo])
126  df_3 = group_1(df_3,ordem_gb[modo])
127  df_4 = group_1(df_4,ordem_gb[modo])
128  df_5 = group_1(df_5,ordem_gb[modo])
129  df_6 = group_1(df_6,ordem_gb[modo])
130
131  ##-----Criar figura e eixos com subplots()
      em gráficos separados (Grid 2X2)-----##
132  fig,axs = plt.subplots(2,2,figsize=(22,15),dpi=100)
133
134  axs[0,0].scatter(df_1[col_x], df_1[col_y1], color =
      '#FF3333',marker="P", s = size+5, label=legenda[modo][0])
135  axs[0,0].plot(df_1[col_x], df_1[col_y1],color =
      '#FF3333',linewidth=0.75)
136  axs[0,0].text(eq_e1_px[modo][0],eq_e1_py[modo][0],eq_e1[modo][0],fon
      tsize=f_size-1,bbox={'facecolor':
      '#FFFFFF','edgecolor':'none','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
137
138  axs[1,0].scatter(df_1[col_x], df_1[col_y2], color =
      '#3333FF',marker="P", s = size+5,label=legenda[modo][0])
139  axs[1,0].plot(df_1[col_x], df_1[col_y2],color =
      '#3333FF',linewidth=0.75)
140  axs[1,0].text(eq_e2_px[modo][0],eq_e2_py[modo][0],eq_e2[modo][0],fon
      tsize=f_size-1,bbox={'facecolor':
      '#FFFFFF','edgecolor':'none','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
141
142  axs[0,1].scatter(df_4[col_x], df_4[col_y1], color =
      '#FF9933',marker="X", s = size+12, label=legenda[modo+3][0])
143  axs[0,1].plot(df_4[col_x], df_4[col_y1],color =
      '#FF9933',linewidth=0.75)
144  axs[0,1].text(eq_e1_px[modo+3][0],eq_e1_py[modo+3][0],eq_e1[modo+3][
      0],fontsize=f_size-1,bbox={'facecolor':
      '#FFFFFF','edgecolor':'none','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
145
146  axs[1,1].scatter(df_4[col_x], df_4[col_y2], color =
      '#9933FF',marker="X", s = size+12,label=legenda[modo+3][0])
147  axs[1,1].plot(df_4[col_x], df_4[col_y2],color =
      '#9933FF',linewidth=0.75)
148  axs[1,1].text(eq_e2_px[modo+3][0],eq_e2_py[modo+3][0],eq_e2[modo+3][
      0],fontsize=f_size-1,bbox={'facecolor':
      '#FFFFFF','edgecolor':'none','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
149

```



```

150  axs[0,0].scatter(df_2[col_x], df_2[col_y1], color =
      '#CC0000',marker="d", s = size+2, label=legenda[modo][1])
151  axs[0,0].plot(df_2[col_x], df_2[col_y1],color =
      '#CC0000',linewidth=0.75)
152  axs[0,0].text(eq_e1_px[modo][1],eq_e1_py[modo][1],eq_e1[modo][1],fon
      tsize=f_size-1,bbox={'facecolor':
      '#FFFFFF', 'edgecolor': 'none', 'alpha': alpha, 'pad':
      pad, 'boxstyle': 'round, pad=0.25'})
153
154  axs[1,0].scatter(df_2[col_x], df_2[col_y2], color =
      '#0000CC',marker="d", s = size+2, label=legenda[modo][1])
155  axs[1,0].plot(df_2[col_x], df_2[col_y2],color =
      '#0000CC',linewidth=0.75)
156  axs[1,0].text(eq_e2_px[modo][1],eq_e2_py[modo][1],eq_e2[modo][1],fon
      tsize=f_size-1,bbox={'facecolor':
      '#FFFFFF', 'edgecolor': 'none', 'alpha': alpha, 'pad':
      pad, 'boxstyle': 'round, pad=0.25'})
157
158  axs[0,1].scatter(df_5[col_x], df_5[col_y1], color =
      '#CC6600',marker="D", s = size-2, label=legenda[modo+3][1])
159  axs[0,1].plot(df_5[col_x], df_5[col_y1],color =
      '#CC6600',linewidth=0.75)
160  axs[0,1].text(eq_e1_px[modo+3][1],eq_e1_py[modo+3][1],eq_e1[modo+3][
      1],fontsize=f_size-1,bbox={'facecolor':
      '#FFFFFF', 'edgecolor': 'none', 'alpha': alpha, 'pad':
      pad, 'boxstyle': 'round, pad=0.25'})
161
162  axs[1,1].scatter(df_5[col_x], df_5[col_y2], color =
      '#6600CC',marker="D", s = size-2, label=legenda[modo+3][1])
163  axs[1,1].plot(df_5[col_x], df_5[col_y2],color =
      '#6600CC',linewidth=0.75)
164  axs[1,1].text(eq_e2_px[modo+3][1],eq_e2_py[modo+3][1],eq_e2[modo+3][
      1],fontsize=f_size-1,bbox={'facecolor':
      '#FFFFFF', 'edgecolor': 'none', 'alpha': alpha, 'pad':
      pad, 'boxstyle': 'round, pad=0.25'})
165
166  axs[0,0].scatter(df_3[col_x], df_3[col_y1], color =
      '#660000',marker="o", s = size+3, label=legenda[modo][2])
167  axs[0,0].plot(df_3[col_x], df_3[col_y1],color =
      '#660000',linewidth=0.75)
168  axs[0,0].text(eq_e1_px[modo][2],eq_e1_py[modo][2],eq_e1[modo][2],fon
      tsize=f_size-1,bbox={'facecolor':
      '#FFFFFF', 'edgecolor': 'none', 'alpha': alpha, 'pad':
      pad, 'boxstyle': 'round, pad=0.25'})
169
170  axs[1,0].scatter(df_3[col_x], df_3[col_y2], color =
      '#000066',marker="o", s = size+3, label=legenda[modo][2])
171  axs[1,0].plot(df_3[col_x], df_3[col_y2],color =
      '#000066',linewidth=0.75)
172  axs[1,0].text(eq_e2_px[modo][2],eq_e2_py[modo][2],eq_e2[modo][2],fon
      tsize=f_size-1,bbox={'facecolor':
      '#FFFFFF', 'edgecolor': 'none', 'alpha': alpha, 'pad':
      pad, 'boxstyle': 'round, pad=0.25'})

```

```

173
174  axs[0,1].scatter(df_6[col_x], df_6[col_y1], color =
      '#663300',marker="p", s = size+10, label=legenda[modo+3][2])
175  axs[0,1].plot(df_6[col_x], df_6[col_y1],color =
      '#663300',linewidth=0.75)
176  axs[0,1].text(eq_e1_px[modo+3][2],eq_e1_py[modo+3][2],eq_e1[modo+3][
      2],fontsize=f_size-1,bbox={'facecolor':
      '#FFFFFF','edgecolor':'none','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
177
178  axs[1,1].scatter(df_6[col_x], df_6[col_y2], color =
      '#330066',marker="p", s = size+10,label=legenda[modo+3][2])
179  axs[1,1].plot(df_6[col_x], df_6[col_y2],color =
      '#330066',linewidth=0.75)
180  axs[1,1].text(eq_e2_px[modo+3][2],eq_e2_py[modo+3][2],eq_e2[modo+3][
      2],fontsize=f_size-1,bbox={'facecolor':
      '#FFFFFF','edgecolor':'none','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
181
182  ##-----Limites eixos-----
      -----##
183  axs[0,0].set_xlim([x_scale[modo][0],x_scale[modo][1]])
184  axs[0,0].set_ylim([y_scale_1[modo][0],y_scale_1[modo][1]])
185  axs[1,0].set_xlim([x_scale[modo][0],x_scale[modo][1]])
186  axs[1,0].set_ylim([y_scale_2[modo][0],y_scale_2[modo][1]])
187  axs[0,1].set_xlim([x_scale[modo][0],x_scale[modo][1]])
188  axs[0,1].set_ylim([y_scale_1[modo][0],y_scale_1[modo][1]])
189  axs[1,1].set_xlim([x_scale[modo][0],x_scale[modo][1]])
190  axs[1,1].set_ylim([y_scale_2[modo][0],y_scale_2[modo][1]])
191
192  ##-----Texto de condições da simulação-----
      -----##
193  axs[0,0].text(txt_px_1[modo],txt_py_1[modo],texto[modo],fontsize=f_s
      ize,bbox={'facecolor': '#FFFFFF','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
194  axs[1,0].text(txt_px_2[modo],txt_py_2[modo],texto[modo],fontsize=f_s
      ize,bbox={'facecolor': '#FFFFFF','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
195  axs[0,1].text(txt_px_1[modo],txt_py_1[modo],texto[modo+3],fontsize=f
      _size,bbox={'facecolor': '#FFFFFF','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
196  axs[1,1].text(txt_px_2[modo],txt_py_2[modo],texto[modo+3],fontsize=f
      _size,bbox={'facecolor': '#FFFFFF','alpha': alpha,'pad':
      pad,'boxstyle':'round, pad=0.25'})
197
198  ##-----Configuração do grid-----
      -----##
199  axs[0,0].xaxis.set_major_locator(MultipleLocator(major_tick_x[modo]
      ))
200  axs[0,0].yaxis.set_major_locator(MultipleLocator(major_tick_y_1[modo
      ]))
201  axs[0,0].yaxis.set_minor_locator(AutoMinorLocator())

```

```

202  axs[0,0].grid(which='major',linestyle = '-', linewidth = 1,alpha =
    0.5)
203  axs[0,0].set_axisbelow(True)
204
205  axs[1,0].xaxis.set_major_locator(MultipleLocator(major_tick_x[modo]
    ))
206  axs[1,0].yaxis.set_major_locator(MultipleLocator(major_tick_y_2[modo
    ]))
207  axs[1,0].yaxis.set_minor_locator(AutoMinorLocator())
208  axs[1,0].grid(which='major',linestyle = '-', linewidth = 1,alpha =
    0.5)
209  axs[1,0].set_axisbelow(True)
210
211  axs[0,1].xaxis.set_major_locator(MultipleLocator(major_tick_x[modo]
    ))
212  axs[0,1].yaxis.set_major_locator(MultipleLocator(major_tick_y_1[modo
    ]))
213  axs[0,1].yaxis.set_minor_locator(AutoMinorLocator())
214  axs[0,1].grid(which='major',linestyle = '-', linewidth = 1,alpha =
    0.5)
215  axs[0,1].set_axisbelow(True)
216
217  axs[1,1].xaxis.set_major_locator(MultipleLocator(major_tick_x[modo]
    ))
218  axs[1,1].yaxis.set_major_locator(MultipleLocator(major_tick_y_2[modo
    ]))
219  axs[1,1].yaxis.set_minor_locator(AutoMinorLocator())
220  axs[1,1].grid(which='major',linestyle = '-', linewidth = 1,alpha =
    0.5)
221  axs[1,1].set_axisbelow(True)
222
223  ##-----Etiquetas dos gráficos-----
    -----##
224
225  axs[0,0].set_ylabel(y_label[0], fontsize = 14, fontweight="medium")
226  axs[0,0].set_xlabel(x_label[modo], fontsize = 14,
    fontweight="medium")
227  axs[1,0].set_ylabel(y_label[1], fontsize = 14, fontweight="medium")
228  axs[1,0].set_xlabel(x_label[modo], fontsize = 14,
    fontweight="medium")
229  axs[0,1].set_ylabel(y_label[0], fontsize = 14, fontweight="medium")
230  axs[0,1].set_xlabel(x_label[modo], fontsize = 14,
    fontweight="medium")
231  axs[1,1].set_ylabel(y_label[1], fontsize = 14, fontweight="medium")
232  axs[1,1].set_xlabel(x_label[modo], fontsize = 14,
    fontweight="medium")
233
234  axs[0,0].legend(loc=4,frameon=True,framealpha=alpha,edgecolor='inher
    it',labelspacing = 0.25,borderpad=0.5)
235  axs[1,0].legend(loc=leng_loc[modo],frameon=True,framealpha=alpha,edg
    ecolor='inherit',labelspacing = 0.25,borderpad=0.5)

```

```
236  axs[0,1].legend(loc=4,frameon=True,framealpha=alpha,edgecolor='inherit',labelspacing = 0.25,borderpad=0.5)
237  axs[1,1].legend(loc=leng_loc[modo],frameon=True,framealpha=alpha,edgecolor='inherit',labelspacing = 0.2,borderpad=0.5)
238
239  ##-----Salvar o gráfico em arquivo .png-----
240  -----##
241  if ger_graf == 'S' and ger_graf != "":
242      plt.savefig('Grafico_final_'+str(modo+1)+'_e',pad_inches=0.1)
243  ##-----Mostar o gráfico-----
244  -----##
244  plt.show()
```

ANEXO A - Tabela de propriedades termofísicas da água

Temperatura	Pressão	Volume específico		Calor de Vaporização	Calor específico		Viscosidade dinâmica		Condutividade térmica		Número de Prandtl		Tensão superficial	Coeficiente de expansão
		$\vartheta_f \cdot 10^3$	ϑ_g		$c_{p,f}$	$c_{p,g}$	$\mu_f \cdot 10^6$	$\mu_g \cdot 10^6$	$k_f \cdot 10^3$	$k_g \cdot 10^3$	Pr_f	Pr_g		
[K]	[bars]	[m ³ /kg]		[kJ/kg]	[kJ/(kg · K)]		[N · s/m ²]		[W/(m · K)]	-	-	[N/m]		[K ⁻¹]
273,15	0,00611	1,0000	206,30	2502	4,217	1,854	1750	8,02	569	18,2	12,99	0,815	75,5	-68,05
275	0,00697	1,0000	181,70	2497	4,211	1,855	1652	8,09	574	18,3	12,22	0,817	75,3	-32,74
280	0,00990	1,0000	130,40	2485	4,198	1,858	1422	8,29	582	18,6	10,26	0,825	74,8	46,04
285	0,01387	1,0000	99,40	2473	4,189	1,861	1225	8,49	590	18,9	8,81	0,833	74,3	114,1
290	0,01917	1,0010	69,70	2461	4,184	1,864	1080	8,69	598	19,3	7,56	0,841	73,7	174
295	0,02617	1,0020	51,94	2449	4,181	1,868	959	8,89	606	19,5	6,62	0,849	72,7	227,5
300	0,03531	1,0030	39,13	2438	4,179	1,872	855	9,09	613	19,6	5,83	0,857	71,7	276,1
305	0,04712	1,0050	29,74	2426	4,178	1,877	769	9,29	620	20,1	5,20	0,865	70,9	320,6
310	0,06221	1,0070	22,93	2414	4,178	1,882	695	9,49	628	20,4	4,62	0,873	70,0	361,9
315	0,08132	1,0090	17,82	2402	4,179	1,888	631	9,69	634	20,7	4,16	0,883	69,2	400,4
320	0,10530	1,0110	13,98	2390	4,180	1,895	577	9,89	640	21	3,77	0,894	68,3	436,7
325	0,13510	1,0130	11,06	2378	4,182	1,903	528	10,09	645	21,3	3,42	0,901	67,5	471,2
330	0,17190	1,0160	8,82	2366	4,184	1,911	489	10,29	650	21,7	3,15	0,908	66,6	504
335	0,21670	1,0180	7,09	2354	4,186	1,920	453	10,49	656	22	2,88	0,916	65,8	535,5
340	0,27130	1,0210	5,74	2342	4,188	1,930	420	10,69	660	22,3	2,66	0,925	64,9	566
345	0,33720	1,0240	4,68	2329	4,191	1,941	389	10,89	664	22,6	2,45	0,933	64,1	595,4
350	0,41630	1,0270	3,846	2317	4,195	1,954	365	11,09	668	23	2,29	0,942	63,2	624,2
355	0,51000	1,0300	3,18	2304	4,199	1,968	343	11,29	671	23,3	2,14	0,951	62,3	652,3
360	0,62090	1,0340	2,645	2291	4,203	1,983	324	11,49	674	23,7	2,02	0,960	61,4	697,9
365	0,7514	1,0380	2,212	2278	4,209	1,999	306	11,69	677	24,1	1,91	0,969	60,5	707,1
370	0,904	1,0410	1,861	2265	4,214	2,017	289	11,89	679	24,5	1,80	0,978	59,5	728,7
373,15	1,0133	1,0440	1,679	2257	4,217	2,029	279	12,02	680	24,8	1,76	0,984	58,9	750,1

Temperatura	Pressão	Volume específico		Calor de Vaporização	Calor específico		Viscosidade dinâmica		Condutividade térmica		Número de Prandtl		Tensão superficial	Coeficiente de expansão
		$\vartheta_f \cdot 10^3$	ϑ_g		$c_{p,f}$	$c_{p,g}$	$\mu_f \cdot 10^6$	$\mu_g \cdot 10^6$	$k_f \cdot 10^3$	$k_g \cdot 10^3$	Pr_f	Pr_g		
[K]	[bars]	[m ³ /kg]		[kJ/kg]	[kJ/(kg · K)]		[N · s/m ²]		[W/(m · K)]	-	-	[N/m]		[K ⁻¹]
375	1,0815	1,0450	1,574	2252	4,22	2,036	274	12,09	681	24,9	1,70	0,987	58,6	761
380	1,2869	1,0490	1,337	2239	4,226	2,057	260	12,29	683	25,4	1,61	0,999	57,6	788
385	1,5233	1,0530	1,142	2225	4,232	2,080	248	12,49	685	25,8	1,53	1,004	56,6	814
390	1,7940	1,0580	0,98	2212	4,239	2,104	237	12,69	686	26,3	1,47	1,013	55,6	841
400	2,4550	1,0670	0,731	2183	4,256	2,158	217	13,05	688	27,2	1,34	1,033	53,6	896
410	0,3302	1,0770	0,553	2153	4,278	2,221	200	13,42	688	28,2	1,24	1,054	51,5	952
420	0,4370	1,0880	0,425	2123	4,302	2,291	185	13,79	688	29,8	1,12	1,075	49,4	1010
430	0,5699	1,0990	0,331	2001	4,331	2,369	173	14,14	685	30,4	1,09	1,100	47,2	-
440	7,333	1,1100	0,261	2059	4,36	2,46	162	14,50	682	31,7	1,04	1,120	45,1	-
450	9,319	1,1230	0,208	2024	4,40	2,56	152	14,85	678	33,1	0,99	1,140	42,9	-
460	11,71	1,1370	0,167	1989	4,44	2,68	143	15,19	673	34,6	0,95	1,170	40,7	-
470	14,55	1,1520	0,136	1951	4,48	2,79	136	15,54	667	36,3	0,92	1,200	38,5	-
480	17,90	1,1670	0,111	1912	4,53	2,94	129	15,88	660	38,1	0,89	1,230	36,2	-
490	21,83	1,1840	0,0922	1870	4,59	3,10	124	16,23	651	40,1	0,87	1,250	33,9	-
500	26,40	1,2030	0,0766	1825	4,66	3,27	118	16,59	642	42,3	0,86	1,280	31,6	-
510	31,66	1,2220	0,0631	1779	4,74	3,47	113	16,95	631	44,7	0,85	1,310	29,3	-
520	37,70	1,2440	0,0525	1730	4,84	3,70	108	17,33	621	47,5	0,84	1,350	26,9	-
530	44,58	1,2680	0,0445	1679	4,95	3,96	104	17,72	608	50,6	0,85	1,390	24,5	-
540	52,38	1,2940	0,0375	1622	5,08	4,27	101	18,1	594	54	0,86	1,430	22,1	-
550	61,19	1,323	0,0317	1564	5,24	4,64	97	18,6	580	58,3	0,87	1,470	19,7	-
560	71,08	1,355	0,0269	1499	5,43	5,09	94	19,1	563	63,7	0,90	1,520	17,3	-
570	82,16	1,392	0,0228	1429	5,68	5,67	91	19,7	548	76,7	0,94	1,590	15	-

Temperatura	Pressão	Volume específico		Calor de Vaporização	Calor específico		Viscosidade dinâmica		Condutividade térmica		Número de Prandtl		Tensão superficial	Coeficiente de expansão
		$\vartheta_f \cdot 10^3$	ϑ_g		$c_{p,f}$	$c_{p,g}$	$\mu_f \cdot 10^6$	$\mu_g \cdot 10^6$	$k_f \cdot 10^3$	$k_g \cdot 10^3$	Pr_f	Pr_g		
[K]	[bars]	[m ³ /kg]		[kJ/kg]	[kJ/(kg · K)]		[N · s/m ²]		[W/(m · K)]	-	-	[N/m]		[K ⁻¹]
580	94,51	1,433	0,0193	1353	6	6,4	88	20,4	528	76,7	0,99	1,68	12,8	-
590	108,3	1,482	0,0163	1274	6,41	7,35	84	21,5	513	84,1	1,05	1,84	10,5	-
600	123,5	1,541	0,0137	1176	7	8,75	81	2,27	497	92,9	1,14	2,15	8,4	-
610	137,3	1,612	0,0115	1068	7,85	11,1	77	24,1	467	103	1,3	2,6	6,3	-
620	159,1	1,705	0,0094	941	9,15	15,4	72	25,9	444	114	1,52	3,46	4,5	-
625	169,1	1,778	0,0085	858	10,6	18,3	70	27	430	121	1,65	4,2	3,5	-
630	179,7	1,856	0,0075	781	12,6	22,1	67	28	412	130	2	4,8	2,6	-
635	190,9	1,935	0,0066	683	16,4	27,6	64	30	392	141	2,7	6	1,5	-
640	202,7	2,075	0,0057	560	26	42	59	32	367	155	4,2	9,6	0,8	-
645	215,2	2,351	0,0045	361	90	-	54	37	331	178	12	26	0,1	-
647,3	221,2	3,17	0,0032	0	∞	∞	45	45	238	238	∞	∞	0	-

Fonte: Adaptado de Bergman et al. (2019, p. 599-600)

ANEXO B - Tabela de propriedades termofísicas do ar

Temperatura	Densidade	Calor específico	Viscosidade dinâmica	Viscosidade cinemática	Condutividade térmica	Viscosidade	Prandtl
T	ρ	c_p	$\mu \cdot 10^7$	$\nu \cdot 10^6$	$k \cdot 10^3$	$\alpha \cdot 10^6$	Pr
[K]	[kg/m ³]	[kJ/(kg · K)]	[N · s/m ²]	[m ² /s]	[W/(m · K)]	[m ² /s]	-
100	3,5562	1,032	71,1	2	9,34	2,54	0,786
150	2,3364	1,012	103,4	4,426	13,8	5,84	0,758
200	1,7458	1,007	132,5	7,59	18,1	10,3	0,737
250	1,3947	1,006	159,6	11,44	22,3	15,9	0,720
300	1,1614	1,007	184,6	15,89	26,3	22,5	0,707
350	0,9950	1,009	208,2	20,92	30	29,9	0,700
400	0,8711	1,014	230,1	26,41	33,8	38,3	0,690
450	0,7740	1,021	250,7	32,39	37,3	47,2	0,686
500	0,6964	1,030	270,1	38,79	40,7	56,7	0,684
550	0,6329	1,040	288,4	45,57	43,9	66,7	0,683
600	0,5804	1,051	305,8	52,69	46,9	76,9	0,685
650	0,5356	1,063	322,5	60,21	49,7	87,3	0,690
700	0,4975	1,075	338,8	68,1	52,4	98	0,695
750	0,4643	1,087	354,6	76,37	54,9	109	0,702
800	0,4354	1,099	369,8	84,93	57,3	120	0,709
850	0,4097	1,110	384,3	93,8	59,6	131	0,716
900	0,3868	1,121	398,1	102,9	62	143	0,720
950	0,3666	1,131	411,3	112,2	64,3	155	0,723
1000	0,3482	1,141	424,4	121,9	66,7	168	0,726
1100	0,3166	1,159	449	141,8	71,5	195	0,728
1200	0,2902	1,175	473	162,9	76,3	224	0,728
1300	0,2679	1,189	496	185,1	82	257	0,719
1400	0,2488	1,207	530	213	91	303	0,703
1500	0,2322	1,230	557	240	100	350	0,685
1600	0,2177	1,248	584	268	106	390	0,688
1700	0,2049	1,267	611	298	113	435	0,685
1800	0,1935	1,286	637	329	120	482	0,683
1900	0,1833	1,307	663	362	128	534	0,677
2000	0,1741	1,337	689	396	137	589	0,672
2100	0,1658	1,372	715	431	147	646	0,667
2200	0,1582	1,417	740	468	160	714	0,655
2300	0,1513	1,478	766	506	175	783	0,647
2400	0,1448	1,558	792	547	196	869	0,630
2500	0,1389	1,665	818	589	222	960	0,613
3000	0,1135	2,726	955	841	486	1570	0,536

Fonte: Adaptado de Bergman et al. (2019, p. 593)

ANEXO C - Tabela de saturação da água

Temperatura		Pressão		Volume específico		Energia interna		Entalpia			Entropia	
T	P	$\vartheta_f \cdot 10^3$	ϑ_g	u_f	u_g	h_f	h_{fg}	h_g	s_f	s_g		
[°C]	[bar]	[m ³ /kg]				[kJ/kg]						
0,01	0,00611	1,0002	206,136	0,00	2375,3	0,01	2501,3	2501,4	0	9,1562		
4	0,00813	1,0001	157,232	16,77	2380,9	16,78	2491,9	2508,7	0,061	9,0514		
5	0,00872	1,0001	147,120	20,97	2382,3	20,98	2489,6	2510,6	0,0761	9,0257		
6	0,00935	1,0001	137,734	25,19	2383,6	25,20	2487,2	2512,4	0,0912	9,0003		
8	0,01072	1,0002	120,917	33,59	2386,4	33,60	2482,5	2516,1	0,1212	8,9501		
10	0,01228	1,0004	106,379	42,00	2389,2	42,01	2477,7	2519,8	0,151	8,9008		
11	0,01312	1,0004	99,857	46,20	2390,5	46,20	2475,4	2521,6	0,1658	8,8765		
12	0,01402	1,0005	93,784	50,41	2391,9	50,41	2473,0	2523,4	0,1806	8,8524		
13	0,01497	1,0007	88,124	54,60	2393,3	54,60	2470,7	2525,3	0,1953	8,8285		
14	0,01598	1,0008	82,848	58,79	2394,7	58,80	2468,3	2527,1	0,2099	8,8048		
15	0,01705	1,0009	77,926	62,99	2396,1	62,99	2465,9	2528,9	0,2245	8,7814		
16	0,01818	1,0011	73,333	67,18	2397,4	67,19	2463,6	2530,8	0,239	8,7582		
17	0,01938	1,0012	69,044	71,38	2398,8	71,38	2461,2	2532,6	0,2535	8,7351		
18	0,02064	1,0014	65,038	75,57	2400,2	75,58	2458,8	2534,4	0,2679	8,7123		
19	0,02198	1,0016	61,293	79,76	2401,6	79,77	2456,5	2536,2	0,2823	8,6897		
20	0,02339	1,0018	57,791	83,95	2402,9	83,96	2454,1	2538,1	0,2966	8,6672		
21	0,02487	1,0020	54,514	88,14	2404,3	88,14	2451,8	2539,9	0,3109	8,645		
22	0,02645	1,0022	51,447	92,32	2405,7	92,33	2449,4	2541,7	0,3251	8,6229		
23	0,02810	1,0024	48,574	96,51	2407,0	96,52	2447,0	2543,5	0,3393	8,6011		
24	0,02985	1,0027	45,883	100,70	2408,4	100,70	2444,7	2545,4	0,3534	8,5794		
25	0,03169	1,0029	43,360	104,88	2409,8	104,89	2442,3	2547,2	0,3674	8,558		
26	0,03363	1,0032	40,994	109,06	2411,1	109,07	2439,9	2549,0	0,3814	8,5367		
27	0,03567	1,0035	38,774	113,25	2412,5	113,25	2437,6	2550,8	0,3954	8,5156		
28	0,03782	1,0037	36,690	117,42	2413,9	117,43	2435,2	2552,6	0,4093	8,4946		
29	0,04008	1,0040	34,733	121,60	2415,2	121,61	2432,8	2554,5	0,4231	8,4739		
30	0,04246	1,0043	32,894	125,78	2416,6	125,79	2430,5	2556,3	0,4369	8,4533		
31	0,04496	1,0046	31,165	129,96	2418,0	129,97	2428,1	2558,1	0,4507	8,4329		
32	0,04759	1,0050	29,540	134,14	2419,3	134,15	2425,7	2559,9	0,4644	8,4127		
33	0,05034	1,0053	28,011	138,32	2420,7	138,33	2423,4	2561,7	0,4781	8,3927		
34	0,05324	1,0056	26,571	142,50	2422,0	142,50	2421,0	2563,5	0,4917	8,3728		
35	0,05628	1,0060	25,216	146,67	2423,4	146,68	2418,6	2565,3	0,5053	8,3531		
36	0,05947	1,0063	23,940	150,85	2424,7	150,86	2416,2	2567,1	0,5188	8,3336		
38	0,06632	1,0071	21,602	159,20	2427,4	159,21	2411,5	2570,7	0,5458	8,295		
40	0,07384	1,0078	19,523	167,56	2430,1	167,57	2406,7	2574,3	0,5725	8,257		

Temperatura		Pressão		Volume específico		Energia interna		Entalpia			Entropia	
T	P	$\vartheta_f \cdot 10^3$	ϑ_g	u_f	u_g	h_f	h_{fg}	h_g	s_f	s_g		
[°C]	[bar]	[m ³ /kg]				[kJ/kg]						
45	0,09593	1,0099	15,258	188,44	2436,8	188,45	2394,8	2583,2	0,6387	8,1648		
50	0,1235	1,0121	12,032	209,32	2443,5	209,33	2382,7	2592,1	0,7038	8,0763		
55	0,1576	1,0146	9,568	230,21	2450,1	230,23	2370,7	2600,9	0,7679	7,9913		
60	0,1994	1,0172	7,671	251,11	2456,6	251,13	2358,5	2609,6	0,8312	7,9096		
65	0,2503	1,0199	6,197	272,02	2463,1	272,06	2346,2	2618,3	0,8935	7,831		
70	0,3119	1,0228	5,042	292,95	2469,6	292,98	2333,8	2626,8	0,9549	7,7553		
75	0,3858	1,0259	4,131	313,90	2475,9	313,93	2321,4	2635,3	1,0155	7,6824		
80	0,4739	1,0291	3,407	334,86	2482,2	334,91	2308,8	2643,7	1,0753	7,6122		
85	0,5783	1,0325	2,828	355,84	2488,4	355,90	2296,0	2651,9	1,1343	7,5445		
90	0,7014	1,0360	2,361	376,85	2494,5	376,92	2283,2	2660,1	1,1925	7,4791		
95	0,8455	1,0397	1,982	397,88	2500,6	397,96	2270,2	2668,1	1,25	7,4159		
100	1,014	1,0435	1,673	418,94	2506,5	419,04	2257,0	2676,1	1,3069	7,3549		
110	1,433	1,0516	1,210	461,14	2518,1	461,30	2230,2	2691,5	1,4185	7,2387		
120	1,985	1,0603	0,8919	503,50	2529,3	503,71	2202,6	2706,3	1,5276	7,1296		
130	2,701	1,0697	0,6685	546,02	2539,9	546,31	2174,2	2720,5	1,6344	7,0269		
140	3,613	1,0797	0,5089	588,74	2550,0	589,13	2144,7	2733,9	1,7391	6,9299		
150	4,758	1,0905	0,3928	631,68	2559,5	632,20	2114,3	2746,5	1,8418	6,8379		
160	6,178	1,1020	0,3071	674,86	2568,4	675,55	2082,6	2758,1	1,9427	6,7502		
170	7,917	1,1143	0,2428	718,33	2576,5	719,21	2049,5	2768,7	2,0419	6,6663		
180	10,02	1,1274	0,1941	762,09	2583,7	763,22	2015,0	2778,2	2,1396	6,5857		
190	12,54	1,1414	0,1565	806,19	2590,0	807,62	1978,8	2786,4	2,2359	6,5079		
200	15,54	1,1565	0,1274	850,65	2595,3	852,45	1940,7	2793,2	2,3309	6,4323		
210	19,06	1,1726	0,1044	895,53	2599,5	897,76	1900,7	2798,5	2,4248	6,3585		
220	23,18	1,1900	0,08619	940,87	2602,4	943,62	1858,5	2802,1	2,5178	6,2861		
230	27,95	1,2088	0,07158	986,74	2603,9	990,12	1813,8	2804,0	2,6099	6,2146		
240	33,44	1,2291	0,05976	1033,2	2604,0	1037,3	1766,5	2803,8	2,7015	6,1437		
250	39,73	1,2512	0,05013	1080,4	2602,4	1085,4	1716,2	2801,5	2,7927	6,073		
260	46,88	1,2755	0,04221	1128,4	2599,0	1134,4	1662,5	2796,6	2,8838	6,0019		
270	54,99	1,3023	0,03564	1177,4	2593,7	1184,5	1605,2	2789,7	2,9751	5,9301		
280	64,12	1,3321	0,03017	1227,5	2586,1	1236,0	1543,6	2779,6	3,0668	5,8571		
290	74,36	1,3656	0,02557	1278,9	2576,0	1289,1	1477,1	2766,2	3,1594	5,7821		
300	85,81	1,4036	0,02167	1332,0	2563,0	1344,0	1404,9	2749,0	3,2534	5,7045		
320	112,7	1,4988	0,01549	1444,6	2525,5	1461,5	1238,6	2700,1	3,448	5,5362		
340	145,9	1,6379	0,0108	1570,3	2464,6	1594,2	1027,9	2622,0	3,6594	5,3357		

Temperatura		Pressão		Volume específico		Energia interna		Entalpia		Entropia	
T	P	$\vartheta_f \cdot 10^3$	ϑ_g	u_f	u_g	h_f	h_{fg}	h_g	s_f	s_g	
[°C]	[bar]	[m ³ /kg]				[kJ/kg]					
360	186,5	1,8925	0,006945	1725,2	2351,5	1760,5	720,5	2481,0	3,9147	5,0526	
374,14	220,9	3,155	0,003155	2029,6	2029,6	2099,3	0	2099,3	4,4298	4,4298	

Fonte: Adaptado de Moran (2018, p. 751-752)