

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA
FONSECA – CEFET/RJ**

DEPARTAMENTO DE ENGENHARIA MECÂNICA

FELIPE ANTÔNIO ALMEIDA SERÓDIO

MARIANA MACHADO DA SILVA

**OTIMIZAÇÃO TERMODINÂMICA DE UM CICLO DE
REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS
UTILIZANDO O SIMULADOR DE PROCESSOS IPSEPRO
INTEGRADO AO MATLAB**

TRABALHO DE CONCLUSÃO DE CURSO

Rio de Janeiro

2020

FELIPE ANTÔNIO ALMEIDA SERÓDIO

MARIANA MACHADO DA SILVA

**OTIMIZAÇÃO TERMODINÂMICA DE UM CICLO DE
REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS
UTILIZANDO O SIMULADOR DE PROCESSOS IPSEPRO
INTEGRADO AO MATLAB**

Trabalho de conclusão de curso apresentado em cumprimento às
normas do Departamento de Educação Superior
do CEFET/RJ, como parte dos requisitos para obtenção
do título de Engenheiro Mecânico

Prof. Orientador: Thiago de Oliveira Macedo

Coorientador: Engenheiro Rafael Cardoso Brandão

Rio de Janeiro

2020

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

S486 Seródio, Felipe Antônio Almeida

Otimização termodinâmica de um ciclo de refrigeração por compressão em múltiplos estágios utilizando o simulador de processos IPSEPRO integrado ao MATLAB / Felipe Antônio Almeida Seródio [e] Mariana Machado da Silva — 2020.

118f.+ anexo: il.(algumas color). ; enc.

Projeto Final (Graduação) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca , 2020.

Bibliografia : f. 114-118

Orientador: Thiago de Oliveira Macedo
Coorientador:
Rafael Cardoso Brandão

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

AGRADECIMENTOS

Às nossas famílias por nos darem o suporte necessário para encararmos os desafios que encontramos neste percurso, os conselhos, o amor e o carinho fundamentais para encararmos a vida com coragem e doçura.

Ao CEFET por sua qualidade de ensino e por nos fazer crescer também pessoalmente através de um ambiente repleto de diversidade.

Aos muitos bons professores que tivemos a oportunidade de conviver e adquirir parte de seus conhecimentos, em especial ao nosso professor orientador Thiago Macedo.

Ao Engenheiro Rafael Cardoso, nosso coorientador, pela generosidade em todas as etapas do desenvolvimento deste trabalho.

À COPPE/UFRJ, por ceder a licença do software utilizado (IPSEpro – Chave 9378-021024A2-03656) para desenvolvimento deste estudo e ao Prof. Manuel Ernani de Carvalho Cruz, que mesmo sem nos conhecer viabilizou o acesso a este programa, ampliando o acesso ao conhecimento.

Aos nossos amigos e companheiros, que dividem com a gente muitas das nossas angústias e alegrias.

RESUMO

SERÓDIO, Felipe. SILVA, Mariana. **Otimização Termodinâmica de um Ciclo de Refrigeração por Compressão em Múltiplos Estágios utilizando o Simulador de Processos IPSEpro integrado ao MatLab. 2020.** 163 Folhas. Trabalho de Conclusão de Curso – Centro Federal de Educação Tecnológica Celso Suckow da Fonseca. Rio de Janeiro. Rio de Janeiro, 2020.

O presente trabalho aborda o conceito de otimização energética para um ciclo de refrigeração por compressão de vapor em múltiplos estágios. O trabalho parte da abordagem termodinâmica, trazendo um breve histórico do ciclo de refrigeração por compressão de vapor, elucidando as diferenças entre os ciclos ideais e reais de refrigeração e possibilitando ao leitor noções teóricas básicas que fundamentam as simulações e metodologias aplicadas. Para tanto, foi utilizado um problema de referência e a combinação entre um simulador de processos profissional, IPSEpro, e um pacote computacional, MatLab, com a aplicação de princípios de algoritmos genéticos, baseados na evolução biológica (seleção natural e genética) – método estocástico robusto que traz grandes probabilidades de atingir o ótimo global na convergência do algoritmo. A modelagem e simulação do problema de referência foram realizadas com o IPSEpro. O IPSEpro realiza os balanços de massa e energia e calcula as propriedades termodinâmicas ponto a ponto no ciclo e possui uma biblioteca específica, *Friigo_Lib*, para sistemas de refrigeração. As metodologias de análise e otimização foram implementadas no MatLab, utilizando a ferramenta nativa *Global Optimization Toolbox*. Foi desenvolvida uma interface entre o IPSEpro e o MatLab, de modo que o MatLab executa o IPSEpro, recebe as propriedades termodinâmicas calculadas, o cálculo das transferências de energia (por calor e por trabalho) e realiza o cálculo do Coeficiente de Performance (COP) do ciclo. Com a utilização da ferramenta de otimização, a sequência descrita anteriormente torna-se um processo iterativo, onde as variáveis de decisão escolhidas (pressão do evaporador, do condensador, do separador de líquido, eficiência isentrópica do compressor de baixa pressão e do compressor de alta pressão) são enviadas do MatLab para o IPSEpro, e um novo valor (maximizado) de COP, associado a um novo conjunto (otimizado) de variáveis de decisão, é calculado no final de cada iteração. O processo é interrompido de acordo com as configurações definidas no Algoritmo Genético, e um COP máximo é alcançado, considerando os limites das variáveis de decisão e as restrições do problema de otimização. Por fim, os resultados são apresentados e as devidas validações realizadas de acordo com dados disponíveis e com a literatura.

Palavras-chave: Ciclo de Refrigeração. Simulador de Processos. Otimização Energética. Otimização Termodinâmica. Algoritmo Genético.

ABSTRACT

SERÓDIO, Felipe. SILVA, Mariana. **Thermodynamic Optimization of a Multistage Compression Refrigeration Cycle, using the IPSEpro Process Simulator integrated with MatLab. 2020.** 163 Pages. Trabalho de Conclusão de Curso – Federal Center of Technological Education - Rio de Janeiro. Rio de Janeiro, 2020.

The current study addresses the concept of energy optimization for a multistage compression refrigeration cycle. The work begins with the thermodynamic approach, bringing a brief of the vapor-compression refrigeration cycle history, elucidating the differences between ideal and real refrigeration cycles and providing basic notions to the reader which support to understand the simulation and applied methodologies. For this purpose, a reference problem was used and also the combination of professional process simulator, IPSEpro, and a computational package, MatLab, through the application of genetic algorithm principles, which are based on the biological evolution (natural selection and genetics) - robust stochastic method that brings great probabilities of achieving the global optimum in the convergence of the algorithm. The reference problem modeling and simulation were developed through IPSEpro. The IPSEpro performs the mass and energy balances, calculates the thermodynamic properties point to point in the cycle and has a specific library, Frigo_Lib, for refrigeration systems. The methodologies concerning analysis and optimization were implemented on MatLab, using the native tool *Global Optimization Toolbox*. An interface was developed between IPSEpro and MatLab, so that MatLab executes IPSEpro, receives the calculated thermodynamic properties and energy transfers (by heat and work) and calculates the coefficient of performance (COP) of the cycle. Using the optimization tool, the sequence previously described becomes an iterative process, where the decision variables chosen (evaporator pressure, condenser pressure, flash chamber pressure, low pressure compressor isentropic efficiency and high pressure compressor isentropic efficiency) are sent from MatLab to IPSEpro and a new COP value (maximized), associated to a new set (optimized) of decision variables, is calculated in the end of each iteration. This process is interrupted according to the settings defined on Genetic Algorithm, and a maximum COP is reached, considering the decision variables limits and the optimization problem restrictions. Finally, the results are presented and the validations carried out according to available data and the literature.

Keywords: Refrigeration Cycle. Process Simulator. Energy Optimization. Thermodynamic Optimization. Genetic Algorithm.

LISTA DE ILUSTRAÇÕES

FIGURA 1- DIAGRAMA ESQUEMÁTICO DO SISTEMA EM CASCATA	19
FIGURA 2- DIAGRAMA ESQUEMÁTICO DO SISTEMA DE DUPLO ESTÁGIO DE COMPRESSÃO COM SEPARADOR DE GÁS DE “FLASH”	20
FIGURA 3 - DIAGRAMA ESQUEMÁTICO DO SISTEMA DE COMPRESSÃO COM ESTÁGIO SIMPLES E DUAS TEMPERATURAS DE EVAPORAÇÃO	21
FIGURA 4 - DIAGRAMA ESQUEMÁTICO DO SISTEMA DE COMPRESSÃO EM DUPLO ESTÁGIO E DOIS NÍVEIS DE TEMPERATURA DE EVAPORAÇÃO COM SEPARADOR DE GÁS DE “FLASH”	22
FIGURA 5 – CICLO DE CARNOT REVERSO FONTE: ADAPTADA DE ÇENGEL E BOLES (2011)	30
FIGURA 6 - CICLO IDEAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR	32
FIGURA 7 - CICLO REAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR	33
FIGURA 8 - CICLO IDEAL DE COMPRESSÃO DE VAPOR EM CASCATA	36
FIGURA 9 - DIAGRAMA T-S DO CICLO DE COMPRESSÃO DE VAPOR EM CASCATA	37
FIGURA 10 – SISTEMA DE REFRIGERAÇÃO REAL COM CICLO DE REFRIGERAÇÃO EM CASCATA	37
FIGURA 11 - CICLO IDEAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR EM MÚLTIPLOS ESTÁGIOS	39
FIGURA 12 - CICLO DE REFRIGERAÇÃO COM MÚLTIPLOS PROPÓSITOS E UM ÚNICO COMPRESSOR	42
FIGURA 13 - ILUSTRAÇÃO DO CONCEITO DE CONFIGURAÇÕES ACEITÁVEIS OU “WORKABLE DESIGNS”	44
FIGURA 14 - PONTO ÓTIMO DA FUNÇÃO OBJETIVO U, CONSIDERANDO OS VALORES POSSÍVEIS PARA A VARIÁVEL INDEPENDENTE x_1	45
FIGURA 15 - MÁXIMO E MÍNIMO LOCAL E GLOBAL DENTRO DO DOMÍNIO DE CONFIGURAÇÕES ACEITÁVEIS DE UM PROBLEMA	46
FIGURA 16 – AMBIENTE DE SIMULAÇÃO IPSEPRO – MÓDULO PSE (PROCESS SIMULATION ENVIRONMENT) VERSÃO 6.0	54
FIGURA 17 - VARIÁVEIS DE UM COMPONENTE DEFINIDAS COMO “SET” E “ESTIMATE”	56
FIGURA 18 – COMPONENTES DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR	58
FIGURA 19 – COMPONENTES DO CICLO DE REFRIGERAÇÃO POR ABSORÇÃO DO SISTEMA NH ₃ -H ₂ O	58
FIGURA 20– COMPONENTES DO CICLO DE REFRIGERAÇÃO POR ABSORÇÃO DO SISTEMA LiBr-H ₂ O	59
FIGURA 21 – CICLO DE REFRIGERAÇÃO DO PROBLEMA DE REFERÊNCIA	61
FIGURA 22 - CRIAÇÃO DE UM OBJETO PARA A ESPECIFICAÇÃO DO FLUIDO REFRIGERANTE	65
FIGURA 23– ASSOCIAÇÃO DA CORRENTE A UM FLUIDO REFRIGERANTE	66
FIGURA 24– PRINCÍPIO DA MODELAGEM DO PROBLEMA PELO CONDENSADOR	66
FIGURA 25– CONFIGURAÇÃO DO TÍTULO NA SAÍDA DO COMPRESSOR	67
FIGURA 26– CONFIGURAÇÃO DA PRESSÃO NA ENTRADA DO CONDENSADOR	67
FIGURA 27- CONFIGURAÇÃO DA TRANSFERÊNCIA DE CALOR NO CONDENSADOR	68
FIGURA 28 – JANELA DA VÁLVULA DE EXPANSÃO ONDE OS DADOS DE SAÍDA SERÃO CALCULADOS PELO IPSEPRO	69
FIGURA 29– CONFIGURAÇÃO DA PRESSÃO NA ENTRADA DO SEPARADOR DE LÍQUIDO	69
FIGURA 30- CONFIGURAÇÃO DO SEPARADOR DE LÍQUIDO NO SISTEMA	70
FIGURA 31– CONFIGURAÇÃO DA CORRENTE DE ENTRADA DA SEGUNDA VÁLVULA DE EXPANSÃO	70
FIGURA 32– CONFIGURAÇÃO DO TÍTULO DO REFRIGERANTE NA SAÍDA DO EVAPORADOR	71
FIGURA 33– CONFIGURAÇÃO DOS DADOS DE ENTRADA DO EVAPORADOR	72
FIGURA 34– CONFIGURAÇÃO DA CORRENTE NA ENTRADA DO EVAPORADOR	72
FIGURA 35– CONFIGURAÇÃO DO COMPRESSOR DE BAIXA PRESSÃO	73
FIGURA 36– CONFIGURAÇÃO DO MIXER	73
FIGURA 37– CONFIGURAÇÃO DO COMPRESSOR DE ALTA PRESSÃO	74
FIGURA 38– CONFIGURAÇÃO DO MOTOR ELÉTRICO DO COMPRESSOR DE BAIXA PRESSÃO	74
FIGURA 39– CONFIGURAÇÃO DO MOTOR ELÉTRICO DO COMPRESSOR DE ALTA PRESSÃO	74
FIGURA 40– CONFIGURAÇÃO DO CONECTOR	75
FIGURA 41– MODELAGEM DO SISTEMA E RESULTADOS APÓS A SIMULAÇÃO	76
FIGURA 42– PARTE DO RELATÓRIO GERADO PELO IPSEPRO QUE APRESENTA A DIMENSÃO DO MODELO (85 VARIÁVEIS E 85 EQUAÇÕES)	77

FIGURA 43– RELATÓRIO GERADO PELO IPSEPRO QUE INDICA A QUANTIDADE DE OS ERROS E AVISOS NA SIMULAÇÃO	77
FIGURA 44– DIAGRAMA ESTRUTURAL DO PROGRAMA COMPUTACIONAL DESENVOLVIDO NO MATLAB.....	83
FIGURA 45– FLUXOGRAMA DA FUNÇÃO QUE REALIZA O CÁLCULO DO COP	84
FIGURA 46 - EFEITO DA VARIAÇÃO DA EFICIÊNCIA ISENTRÓPICA DO COMPRESSOR DE BAIXA PRESSÃO SOBRE O COP	91
FIGURA 47– EFEITO DA VARIAÇÃO DA EFICIÊNCIA ISENTRÓPICA DO COMPRESSOR DE ALTA PRESSÃO SOBRE O COP	92
FIGURA 48– EFEITO DA VARIAÇÃO DA PRESSÃO DO EVAPORADOR SOBRE O COP	93
FIGURA 49– EFEITO DA VARIAÇÃO DA PRESSÃO DO CONDENSADOR SOBRE O COP.....	94
FIGURA 50– EFEITO DA VARIAÇÃO DA PRESSÃO DO SEPARADOR DE LÍQUIDO SOBRE O COP.....	95
FIGURA 51– GRÁFICO GERADO DURANTE O PROCESSO DE OTIMIZAÇÃO.....	100
FIGURA 52– GRÁFICO DO VALOR DA FUNÇÃO OBJETIVO (COP) EM FUNÇÃO DO NÚMERO DE CHAMADAS DA FUNÇÃO OBJETIVO	101
FIGURA 53- DIAGRAMA ESTRUTURAL DO PROGRAMA COMPUTACIONAL DESENVOLVIDO NO MATLAB REFERENTE AO ALGORITMO GENÉTICO.....	103
FIGURA 54– FLUXOGRAMA DO ALGORITMO GENÉTICO, BEM COMO SUAS TOMADAS DE DECISÃO	103
FIGURA 55- RESULTADOS OBTIDOS PARA A SIMULAÇÃO DO PROBLEMA DE REFERÊNCIA NO PONTO ÓTIMO	104

LISTA DE TABELAS

TABELA 1 - MÓDULOS DE UTILIZAÇÃO DO IPSE-PRO (SIMTECHNOLOGY)	53
TABELA 2 - OBJETOS DO IPSE-PRO (SIMTECHNOLOGY) UTILIZADOS NA MODELAGEM E SIMULAÇÃO	64
TABELA 3 - COMPARAÇÃO ENTRE OS RESULTADOS DE VAZÃO MÁSSICA, PRESSÃO, TEMPERATURA E ENTALPIA ESPECÍFICA DO IPSEPRO E DO TRABALHO DE REFERÊNCIA.....	78
TABELA 4 - DIFERENÇA ENTRE OS VALORES RELATIVOS DA ENTALPIA ESPECÍFICA PONTO A PONTO	79
TABELA 5 - ERROS PERCENTUAIS DAS PRINCIPAIS INTERAÇÕES ENERGÉTICAS	80
TABELA 6 – PRINCIPAIS COMANDOS UTILIZADOS NA INTERFACE COM	82
TABELA 7 – RESULTADOS OBTIDOS PARA O PONTO OPERACIONAL DO PROBLEMA DE REFERÊNCIA	85
TABELA 8 – VALORES DE REFERÊNCIA DO COP E CALCULADOS ATRAVÉS DO MATLAB	86
TABELA 9 – LIMITES SUPERIOR E INFERIOR DAS VARIÁVEIS DE DECISÃO	98
TABELA 10 – VALORES GERADOS PELO ALGORITMO GENÉTICO A CADA GERAÇÃO.....	102
TABELA 11 – VALORES DAS VARIÁVEIS DE DECISÃO, DAS TAXAS DE TRANSFERÊNCIA DE ENERGIA E DO COP DO PROBLEMA DE REFERÊNCIA OTIMIZADO EM COMPARAÇÃO COM OS RESULTADOS DO PROBLEMA DE REFERÊNCIA NÃO OTIMIZADO.	105
TABELA 12 - COMPARAÇÃO ENTRE OS RESULTADOS DA SIMULAÇÃO COM O IPSEPRO E OS RESULTADOS DO PROBLEMA SIMILAR ENCONTRADO NA LITERATURA.	107
TABELA 13 - COMPARAÇÃO ENTRE OS RESULTADOS DA OTIMIZAÇÃO COM O PROGRAMA DESENVOLVIDO E OS RESULTADOS DO PROBLEMA SIMILAR ENCONTRADO NA LITERATURA OTIMIZADO	108

LISTA DE SÍMBOLOS

C	Constante	
h	Entalpia específica	[kJ/kg]
s	Entropia específica	[kJ/kg.K]
δ	Erro relativo	[%]
U	Função objetivo	
\dot{w}_{total}	Potência total consumida	[W]
H	Restrições de desigualdade	
G	Restrições de igualdade	
U*	Solução ótima da função objetivo	
Q_{cond}	Taxa de calor transferida no condensador	[W]
Q_{evap}	Taxa de calor transferida no evaporador	[W]
T	Temperatura	[°C]
$x_{título}$	Título do refrigerante	[kg/kg]
ε	Tolerância admitida	[%]
\dot{m}	Vazão mássica	[kg/s]
x	Vetor de variáveis de decisão	

SUMÁRIO

1	INTRODUÇÃO	11
1.1.	MOTIVAÇÃO	14
1.2.	JUSTIFICATIVA.....	14
1.3.	OBJETIVO DO TRABALHO	15
1.4.	METODOLOGIA	15
1.5.	ORGANIZAÇÃO DO TRABALHO	17
2	REVISÃO BIBLIOGRÁFICA.....	18
2.1.	REVISÃO TÉCNICA DOS CICLOS DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS	18
2.1.1.	SISTEMAS EM CASCATA.....	18
2.1.2.	COMPRESSÃO EM DUPLO ESTÁGIO E UMA ÚNICA TEMPERATURA DE EVAPORAÇÃO	19
2.1.3.	COMPRESSÃO EM DUPLO ESTÁGIO E DUAS TEMPERATURAS DE EVAPORAÇÃO COM SEPARADOR DE GÁS DE “FLASH”	20
2.2.	MODELAGEM E SIMULAÇÃO DE CICLOS DE REFRIGERAÇÃO UTILIZANDO SIMULADORES DE PROCESSOS	22
2.3.	OTIMIZAÇÃO DE CICLOS DE REFRIGERAÇÃO	23
2.4.	ALGORITMOS GENÉTICOS APLICADOS NA OTIMIZAÇÃO DE SISTEMAS TÉRMICOS.....	24
3	REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR	27
3.1.	BREVE HISTÓRICO	27
3.2.	REFRIGERADORES E BOMBAS DE CALOR	28
3.3.	O CICLO DE CARNOT REVERSO	29
3.4.	O CICLO IDEAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR	31
3.5.	O CICLO REAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR	32
3.6.	SELEÇÃO DO FLUIDO REFRIGERANTE.....	33
3.7.	CICLOS DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR MODIFICADOS	35
3.7.1.	CICLOS DE REFRIGERAÇÃO EM CASCATA	35
3.7.2.	CICLOS DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS	38
3.7.3.	MODELAGEM ENERGÉTICA DE UM CICLO DE DOIS ESTÁGIOS COM UM SEPARADOR DE LÍQUIDO	39
3.7.4.	CICLOS DE REFRIGERAÇÃO COM MÚLTIPLOS PROPÓSITOS EM UM ÚNICO COMPRESSOR.....	41
4	INTRODUÇÃO À OTIMIZAÇÃO	43
4.1.	IDEIAS BÁSICAS	43
4.2.	FORMULAÇÃO GERAL DE UM PROBLEMA DE OTIMIZAÇÃO.....	45
4.2.1.	FORMULAÇÃO GERAL DE UM PROBLEMA DE OTIMIZAÇÃO.....	47
4.3.	CLASSIFICAÇÃO DOS MÉTODOS DE OTIMIZAÇÃO.....	48
5	O SIMULADOR DE PROCESSOS IPSEPRO.....	52

5.1.	PRINCIPAIS CARACTERÍSTICAS DO IPSEPRO	52
5.2.	O MÓDULO PSE (<i>PROCESS SIMULATION ENVIRONMENT</i>)	53
5.2.1.	MÉTODOS DE CÁLCULO UTILIZADOS PELO MÓDULO PSE	56
5.3.	A BIBLIOTECA FRIGO_LIB (<i>REFRIGERATION PROCESS LIBRARY</i>)	57
6	MODELAGEM E SIMULAÇÃO DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR EM MÚLTIPLOS ESTÁGIOS.....	60
6.1.	O PROBLEMA DE REFERÊNCIA	60
6.2.	CRITÉRIO PARA VALIDAÇÃO	61
6.3.	DETALHAMENTO DA MODELAGEM E SIMULAÇÃO DO PROBLEMA DE REFERÊNCIA NO IPSEPRO	62
6.3.1.	OBJETOS UTILIZADOS NA MODELAGEM E SIMULAÇÃO DO PROBLEMA	63
6.3.2.	FLUIDO REFRIGERANTE	65
6.3.3.	CONDENSADOR.....	66
6.3.4.	VÁLVULA DE EXPANSÃO NA SAÍDA DO CONDENSADOR	68
6.3.5.	SEPARADOR DE LÍQUIDO	69
6.3.6.	VÁLVULA DE EXPANSÃO NA SAÍDA DO SEPARADOR.....	70
6.3.7.	EVAPORADOR.....	71
6.3.8.	COMPRESSOR DE BAIXA PRESSÃO	72
6.3.9.	<i>MIXER</i>	73
6.3.10.	COMPRESSOR DE ALTA PRESSÃO	73
6.3.11.	MOTORES.....	74
6.3.12.	CONECTOR	75
6.4.	RESULTADOS E DISCUSSÃO DA SIMULAÇÃO.....	75
6.5.	COMPARAÇÃO COM O PONTO DE PROJETO	77
6.5.1.	CONSIDERAÇÕES SOBRE OS ERROS RELATIVOS	78
6.6.	VALIDAÇÃO DOS RESULTADOS DA SIMULAÇÃO.....	79
7	ANÁLISE ENERGÉTICA DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS	81
7.1.	INTEGRAÇÃO ENTRE O SIMULADOR DE PROCESSOS IPSEPRO E O MATLAB.....	81
7.1.1.	PRINCIPAIS COMANDOS UTILIZADOS NA INTERFACE COM.....	81
7.2.	DIAGRAMA ESTRUTURAL DO PROGRAMA COMPUTACIONAL DESENVOLVIDO NO MATLAB	83
7.3.	FLUXOGRAMA DO CÓDIGO COMPUTACIONAL DESENVOLVIDO.....	83
7.4.	CRITÉRIO PARA VALIDAÇÃO	84
7.5.	RESULTADOS E DISCUSSÃO DA ANÁLISE ENERGÉTICA	85
7.6.	VALIDAÇÃO DOS RESULTADOS DA ANÁLISE ENERGÉTICA	85
8	OTIMIZAÇÃO ENERGÉTICA DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS.....	87
8.1.	CRITÉRIO PARA A VALIDAÇÃO DA OTIMIZAÇÃO	87
8.1.1.	ESTUDOS SIMILARES ENCONTRADOS NA LITERATURA	88

8.2.	FUNÇÃO OBJETIVO DO PROBLEMA DE OTIMIZAÇÃO	88
8.3.	RESTRIÇÕES DO PROBLEMA DE OTIMIZAÇÃO	89
8.4.	VARIÁVEIS DE DECISÃO.....	89
8.4.1.	ESTUDOS PARAMÉTRICOS	90
8.4.1.1.	EFEITO DA EFICIÊNCIA ISENTRÓPICA DO COMPRESSOR NO COP.....	91
8.4.1.2.	EFEITO DA PRESSÃO DO EVAPORADOR NO COP	92
8.4.1.3.	EFEITO DA PRESSÃO DO CONDENSADOR NO COP	93
8.4.1.4.	EFEITO DA PRESSÃO DO SEPARADOR DE LÍQUIDO NO COP	94
8.4.1.5.	RESULTADOS E DISCUSSÃO DAS ANÁLISES PARAMÉTRICAS	95
8.4.1.6.	VALIDAÇÃO DOS RESULTADOS DAS ANÁLISES PARAMÉTRICAS	96
8.5.	METODOLOGIA	96
8.5.1	A FERRAMENTA DE OTIMIZAÇÃO DO MATLAB.....	96
8.6.	DIAGRAMA ESTRUTURAL DO PROGRAMA COMPUTACIONAL DESENVOLVIDO NO MATLAB	102
8.7.	FLUXOGRAMA DOS CÓDIGO COMPUTACIONAIS DESENVOLVIDOS	103
8.8.	RESULTADOS E DISCUSSÃO DA OTIMIZAÇÃO	104
8.9.	VALIDAÇÃO DOS RESULTADOS DA OTIMIZAÇÃO	106
9	CONCLUSÃO	110
	REFERÊNCIAS	114
	APÊNDICE A - MEMÓRIA DE CÁLCULO DA SIMULAÇÃO (GERADA PELO IPSEPRO) 119	
	APÊNDICE B – CÓDIGO COMPUTACIONAL DO MATLAB PARA CÁLCULO DO COP ATRAVÉS DE INTEGRAÇÃO COM O IPSEPRO	141
	ANEXO A - MODELOS UTILIZADOS NOS OBJETOS DA SIMULAÇÃO, DESENVOLVIDOS PELO IPSEPRO.....	145
	ANEXO B - CONDIÇÕES OPERACIONAIS NÃO OTIMIZADAS DO ESTUDO SIMILAR DE OTIMIZAÇÃO DESENVOLVIDO POR DINCER, ROSEN E AHMADI (2017)	157
	ANEXO C – EFEITOS DAS VARIÁVEIS DE DECISÃO SOBRE O COP DE ACORDO COM ESTUDO SIMILAR DE OTIMIZAÇÃO (DINCER, ROSEN E AHMADI, 2017).....	158

1 INTRODUÇÃO

O Balanço Energético Nacional (EPE, 2020), elaborado pela Empresa de Pesquisa Energética, ligada ao Ministério de Minas e Energia, indica que a energia final produzida é composta pela soma da energia útil e da energia perdida, que por sua vez é contabilizada considerando a energia não recuperável e o potencial de economia de energia.

Ainda de acordo com o Balanço Energético Nacional (EPE, 2020), a energia útil é estimada com base nas eficiências médias das instalações de cada setor de atividade e o potencial de economia de energia é estimado com base nas eficiências das instalações mais modernas de cada setor de atividade existentes no ano do estudo. A Energia Não Recuperável é calculada pela diferença entre eles.

O referido relatório indica que a indústria é responsável por cerca de 36% do consumo de energia elétrica do país e a Confederação Nacional da Indústria (Portal da Indústria, 2018) aponta que a energia total gasta com refrigeração, motores elétricos, ar comprimido e iluminação pode representar mais de 50% dos custos totais relacionados à energia elétrica. Estes números mostram a importância do setor industrial e do segmento de refrigeração para a energia elétrica do Brasil.

O Relatório Eficiência Energética em Sistemas de Refrigeração Industrial e Comercial (2005), desenvolvido pela Eletrobrás, explica que diferente do condicionamento de ar para conforto térmico, o condicionamento de ar na indústria tem por objetivo satisfazer as condições requeridas pelo processo. O relatório apresenta a indústria de manufatura, construção e a indústria química e de processo como grandes demandantes de sistemas de refrigeração.

O Relatório Eficiência Energética em Sistemas de Refrigeração Industrial e Comercial (2005) define também o sistema multipressão como sendo um sistema de refrigeração, por compressão de vapor, que possui dois ou mais níveis de baixa pressão (pressão entre o dispositivo de expansão e o compressor). Como principais vantagens desse sistema, o relatório cita a redução de potência requerida pelo compressor; a redução da vazão de refrigerante que circula pelo evaporador e as linhas de líquido, possibilitando a redução de suas dimensões; a redução da capacidade (deslocamento volumétrico) do compressor e o aumento de sua vida útil, já que a diferença de pressão em que trabalha o compressor é reduzida, diminuindo também o desgaste nas superfícies dos mancais.

Tal relatório exemplifica que um sistema multipressão pode ser encontrado em uma indústria de laticínios, em matadouros frigoríficos e em qualquer processo industrial no qual um arranjo de dois ou três estágios de compressão sirva um evaporador com temperaturas abaixo de -20°C .

O Balanço Energético Nacional (EPE, 2020) aponta que os rendimentos energéticos aumentaram nos últimos anos por meio de equipamentos mais eficientes e pela mudança da matriz energética adotada. Sola e Mota (2015) afirmam que para além da redução do consumo de recursos energéticos não renováveis, o uso eficiente de energia acarreta ganhos sociais, ambientais e também ganhos de competitividade no mercado produtivo.

Desta forma, fica evidente que dedicar esforços visando obter a melhor solução viável para o projeto de componentes, equipamentos e sistemas de conversão de energia resulta em inúmeros benefícios para a sociedade. Os avanços científicos e tecnológicos fomentaram as possibilidades de desenvolvimento de soluções mais eficientes para todos os setores produtivos, sendo possível ainda otimizar sistemas de conversão de energia já existentes, utilizando ferramentas computacionais.

De acordo com Jaluria (2007), o conceito de otimização envolve a maximização ou minimização de características ou quantidades de parâmetros de interesse, considerando uma série de restrições, dada uma condição inicial. Jaluria (2007) descreve que para se alcançar o modelo ótimo, todas as possibilidades aceitáveis de arranjos entre as variáveis devem ser consideradas e é fundamental que se tenha conhecimento sobre como identificar as possíveis configurações de modelos aplicáveis ao sistema.

Segundo Jaluria (2007) qualquer processo de otimização exige que se tenha uma função que deve ser maximizada ou minimizada. Essa função é denominada função objetivo. Em geral a função objetivo visa atingir o ponto máximo ou mínimo de parâmetros como custo, lucro, peso, volume, taxa de consumo energético, taxa de transferência de calor, durabilidade, nível de segurança, performance de um sistema, produção, qualidade etc.

Jaluria (2007) esclarece ainda que as restrições de um problema de otimização são as limitações impostas pelas variáveis do sistema. Elas podem ser parâmetros físicos pelos quais os princípios básicos de conservação devem ser respeitados, como também podem envolver limitações de operação como espaço, parâmetros de funcionamento do equipamento, faixa de temperatura aceitável, materiais empregados etc.

Diante da complexidade da otimização de um sistema térmico, torna-se uma boa estratégia a utilização integrada de simuladores de processos e pacotes computacionais comerciais de otimização robustos. A importância do uso de simuladores de processo consolidados deve-se aos códigos validados pelas comunidades científica e industrial com modelos termodinâmicos que atendem às leis de conservação de energia e massa e à demanda por menor custo computacional. Isso atenua parte da complexidade do problema de otimização¹ por tratar as restrições do problema, como será visto em maior detalhe na quarta seção, possibilitando a ampliação do escopo e o aprofundamento do trabalho.

Entre os métodos de otimização, é possível destacar o algoritmo genético como sendo um exemplo de método estocástico, inspirado nos mecanismos de seleção natural e recombinação genética em que são empregadas técnicas de inteligência computacional de modo a encontrar soluções aproximadas em problemas de otimização e busca.

Importante dizer que muitas vezes o resultado ótimo de uma variável não é encontrado padronizado no mercado, fazendo com que na prática a escolha final dos parâmetros do sistema esteja a mais próxima possível do resultado ótimo. Assim, o projeto associado ao sistema não será o projeto ótimo e será denominado projeto viável. Jaluria (2007) exemplifica que uma dimensão ótima de 4,65 metros pode ser aproximada para uma dimensão padronizada de 5 metros que viabilize a construção de um sistema; um resultado que aponte para um motor de 8,34 kW pode implicar em uma escolha realística de 10 kW, pois é a opção disponível no mercado.

Assim, uma vez expostos os conceitos básicos de otimização, deve-se ressaltar as particularidades da otimização de sistemas térmicos. Jaluria (2007) cita alguns exemplos de parâmetros de sistemas térmicos para os quais o objetivo é a maximização: a eficiência, coeficiente de performance (COP) e taxa de calor transferido no evaporador para o caso de refrigeradores. Por outro lado, existem alguns exemplos de parâmetros para os quais o objetivo é a minimização: desperdício de energia e destruição de exergia.

Desta forma, fica claro que projetar sistemas que atendam às demandas térmicas com o mínimo consumo possível de energia elétrica, reduz custos e torna a utilização dos recursos naturais mais racional, reduzindo assim os impactos ambientais.

¹ As restrições relativas aos princípios de conservação (balanços de massa e energia) do problema de otimização são atendidas pelo IPSEpro. Portanto, isso diminui o problema de otimização.

1.1. MOTIVAÇÃO

Tendo em vista as demandas da sociedade moderna com o mínimo de custos e impactos ambientais, o engenheiro, através da promoção de tecnologias funcionais e seguras, consegue viabilizar soluções que atinjam o resultado ótimo, ou seja, o melhor resultado em algum sistema específico.

Os sistemas de refrigeração são sistemas altamente relevantes em diversos setores da indústria, como a indústria alimentícia, plástica, têxtil, química, dentre outras. Para além disso, tais sistemas estão presentes no dia a dia da sociedade moderna, no contexto residencial – geladeiras, freezers, climatizadores – e comercial, essencialmente relacionado à conforto térmico e conservação de produtos a baixas temperaturas.

Assim, compreendê-los e torná-los mais eficientes são formas de adotar um comportamento alinhado com as expectativas de racionalização dos recursos e visão integralista da sociedade moderna, que busca não apenas analisar fenômenos isoladamente, mas compreende-os na sua totalidade, considerando seus efeitos associados, como redução do consumo dos recursos energéticos não renováveis, impacto nos custos e efeitos residuais no meio ambiente (como na camada de ozônio), objetivando a sustentabilidade.

Neste contexto, destaca-se o desafio representado pela viabilização de decisões estratégicas alinhadas às demandas sociais, por exigir domínio em diversos campos do conhecimento, como a programação, a utilização de um simulador de processos termodinâmicos profissional e a otimização de sistemas térmicos, incluindo sistemas térmicos complexos. Portanto, a execução e a redação de um trabalho que integra essas áreas visam contribuir com a difusão dessas competências no setor produtivo.

1.2. JUSTIFICATIVA

A refrigeração é uma área relevante de aplicação da termodinâmica. Os processos de refrigeração e congelamento são importantes na indústria e, em alguns casos, o intervalo de temperatura com o qual trabalham pode ter um alto valor, de modo que a compressão em um único estágio se torna inviável.

Macagnan (2018) explicita que os sistemas de refrigeração com um único estágio são adequados para operar quando as relações de temperatura do condensador e evaporador são

relativamente pequenas, entre 50 °C e 60 °C. Isso porque quando a diferença entre as temperaturas de condensação e evaporação é muito elevada o sistema torna-se ineficiente, há redução da capacidade de refrigeração, além da diminuição da vida útil do compressor.

Segundo Macagnan (2018), aplicações que envolvem temperaturas de vaporização de baixo valor - como sistemas de congelamento de alimentos, aplicações na indústria química e liquefação de gás natural – ou temperaturas de condensação de alto valor, como aplicações de bomba de calor, podem fazer com que a diferença de valores de temperaturas de condensação e evaporação atinja faixas superiores ao limite adequado para ciclos de refrigeração em um único estágio.

Para solucionar estes problemas, Macagnan (2018) aponta que são utilizados sistemas de compressão em múltiplos estágios que, apesar do custo inicial mais alto, requerem menor potência (devido à refrigeração intermediária) do que um único compressor e frequentemente a economia na potência justifica o custo extra do equipamento.

A elaboração deste trabalho se justifica uma vez que há um interesse crescente em aliar metodologias de otimização à termodinâmica para o desenvolvimento de sistemas de conversão de energia (como os sistemas de refrigeração) mais eficientes. Isto porque tais sistemas podem ser aplicados no desenvolvimento de novos projetos de engenharia e na avaliação de projetos existentes, de modo a atingir níveis cada vez maiores de eficiências globais, e níveis cada vez menores de custos de energia e de impactos ambientais.

1.3. OBJETIVO DO TRABALHO

Este trabalho se propõe a realizar a modelagem e a simulação de um ciclo de refrigeração por compressão em múltiplos estágios de um problema de referência e posterior aplicação das metodologias de análise energética e otimização energética ao ciclo em questão, esta última utilizando algoritmos genéticos.

1.4. METODOLOGIA

Para o desenvolvimento do trabalho foi utilizada uma metodologia em que se realizou a modelagem e simulação do problema de referência utilizando o simulador de processos IPSEpro, módulo PSE (Process Simulation Environment), biblioteca Frigo_Lib (Refrigeration Process Library). Para garantir a validade da simulação, comparou-se os resultados nela

encontrados com os resultados do problema de referência, com o cálculo do erro relativo percentual.

Em seguida, desenvolveu-se um programa principal no MatLab para comunicação com o IPSEpro, utilizando a interface COM. Nesta etapa, o bloco inicial do programa principal estabeleceu a interface entre o IPSEpro, arquivo que contém o ciclo de refrigeração do problema de referência, e o MatLab, arquivo que contém o programa principal. Ainda nesta fase do desenvolvimento, o programa principal também viabilizou a execução do IPSEpro e obtenção das propriedades termodinâmicas ponto a ponto do ciclo de refrigeração do problema de referência.

Na sequência, uma função foi programada no MatLab de modo a viabilizar uma análise energética do ciclo de refrigeração do problema de referência, calculando as principais interações e o Coeficiente de Performance (COP). Para garantir a validade dos resultados encontrados na análise energética, os mesmos foram comparados com os resultados do problema de referência, com cálculo do erro relativo percentual. Uma vez validada, a função foi implementada no programa principal.

Dando continuidade ao trabalho, buscou-se verificar o efeito da variação dos parâmetros do ciclo sobre o COP. Para tanto, estudos paramétricos foram realizados e um programa a parte desenvolvido. Importante mencionar que os parâmetros escolhidos estão de acordo com estudos similares encontrados na literatura e que os resultados obtidos foram validados através da comparação com os resultados de estudos similares encontrados na literatura.

Por fim, foi programada uma função no MatLab para a otimização energética do ciclo de refrigeração do problema de referência. Essa função contempla a ferramenta de otimização do MatLab que utiliza o algoritmo genético como método de otimização. Assim como em todas as etapas do projeto, realizou-se a validação dos resultados da otimização energética pela comparação com os resultados da otimização energética dos estudos similares encontrados na literatura.

1.5. ORGANIZAÇÃO DO TRABALHO

O presente trabalho está estruturado em nove seções, sendo a primeira delas dedicada à introdução e esclarecimento ao leitor sobre a metodologia aplicada, à estruturação do trabalho como um todo e aos fatores que fundamentaram a elaboração deste estudo.

A seção dois apresenta uma revisão bibliográfica, levantando alguns trabalhos acadêmicos realizados, semelhantes à proposta deste estudo, indicando a relevância da proposta aqui desenvolvida.

A seção três apresenta uma síntese teórica dos ciclos de compressão de vapor, apresentando um breve histórico sobre conceitos básicos dos ciclos de compressão de vapor, o ciclo de Carnot e alguns dos ciclos de refrigeração – ideal, real, em cascata e em múltiplos estágios – bem como a seleção do fluido refrigerante a ser utilizado.

A seção quatro se propõe a apresentar as ideias básicas relacionadas a otimização, bem como as suas possíveis classificações e a formulação geral de um problema de otimização. Este capítulo apresenta ainda os conceitos básicos de algoritmo genético.

Em seguida, a seção cinco ilustra o simulador de processos IPSEpro, suas principais características e os módulos utilizados para elaboração deste estudo.

A seção seis apresenta o problema de referência, e nela são realizadas a modelagem e a simulação do ciclo estudado no programa IPSEpro, apresentando e discutindo ainda os resultados encontrados, bem como suas validações.

Na sequência, a sétima seção apresenta a análise energética do ciclo de refrigeração por compressão em múltiplos estágios através da integração entre o simulador de processos IPSEpro e o software de computação científica de otimização adotado, o MatLab.

Na seção oito é aplicada a metodologia de otimização do ciclo de refrigeração por compressão em múltiplos estágios, adotando estudos similares encontrados na literatura para obtenção dos critérios de validação e destacando a função objetivo e as restrições do problema para sua solução. A oitava seção apresenta também os estudos de sensibilidade para a avaliação do impacto de alguns parâmetros de interesse do ciclo sobre o coeficiente de performance.

Por fim, a nona seção apresenta as conclusões do trabalho, promovendo as discussões e considerações finais, bem como sugestões para trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

Esta seção está dividida em uma revisão técnica dos ciclos de refrigeração por compressão em múltiplos estágios, seguida de uma revisão de trabalhos sobre os seguintes temas: Modelagem e Simulação de Ciclos de Refrigeração; Otimização de Ciclos de Refrigeração e Algoritmos Genéticos Aplicados na Otimização de Sistemas Térmicos.

2.1. Revisão técnica dos ciclos de refrigeração por compressão em múltiplos estágios

Stoecker e Jabardo (2002) afirmam que uma parcela significativa das instalações na área da Refrigeração Industrial opera com temperaturas de evaporação e condensação que diferem entre si 50 a 80°C. Dossat (2007) afirma que a utilização de sistema de refrigeração simples estágio requerem a utilização de refrigerantes que possuam temperaturas de evaporação baixas o suficiente para prover as temperaturas de refrigeração exigidas e ainda possuam pressões de condensação razoáveis.

As exigências da indústria adicionadas às limitações existentes nos sistemas simples estágio, aumentaram a importância e a utilização de sistemas de refrigeração com múltiplos estágios de compressão. A revisão técnica dos ciclos de refrigeração por compressão em duplo estágio será feita a seguir.

2.1.1. Sistemas em Cascata

Nos sistemas em cascata, são utilizados refrigerantes diferentes para o estágio (ou ciclo) de alta e de baixa pressão. Nesse modelo, a conexão entre os estágios é realizada por meio de um trocador de calor que opera como condensador para o estágio (ciclo) de baixa pressão e como evaporador para o estágio (ciclo) de alta pressão. A Figura 1 apresenta um sistema em cascata.

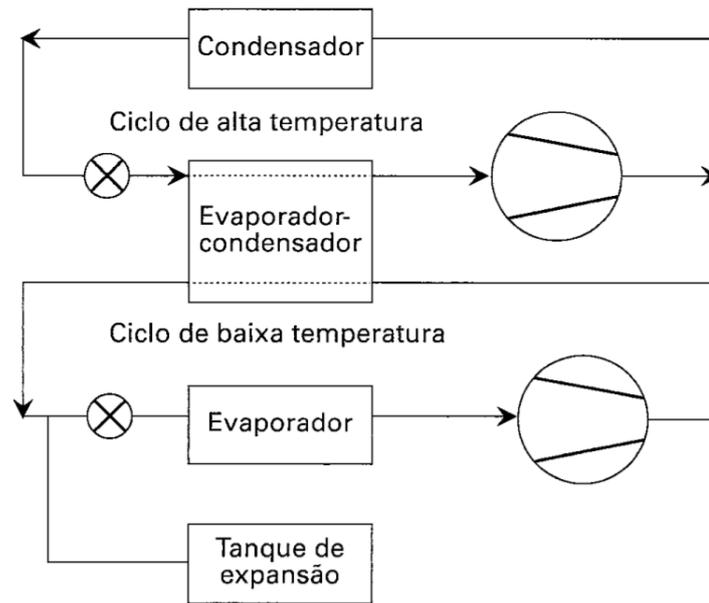


Figura 1- Diagrama Esquemático do Sistema em cascata

Fonte: Stoecker e Jabardo (2002)

2.1.2. Compressão em Duplo Estágio e uma Única Temperatura de Evaporação

Stoecker e Jabardo (2002) informam que os sistemas de duplo estágio e uma única temperatura utilizam o separador líquido-vapor para separar o refrigerante líquido do vapor saturado proveniente do condensador. Esses sistemas servem a um ou mais evaporadores que operam na mesma temperatura de evaporação. A Figura 2 representa o diagrama esquemático desse ciclo.

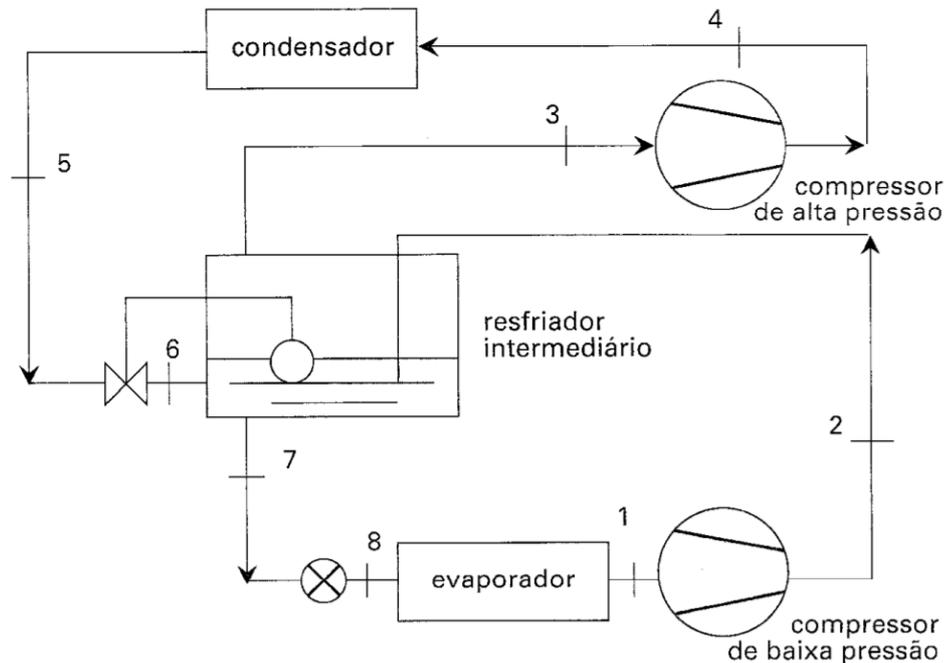


Figura 2- Diagrama Esquemático do Sistema de Duplo Estágio de compressão com separador de gás de “flash”

Fonte: Stoecker e Jabardo (2002)

Nesse sistema, o refrigerante que sai do condensador passa pela válvula controladora de nível (válvula entre os pontos 5 e 6) e entra no separador de gás de “flash” que também faz o papel de resfriador intermediário. Nesse caso, a parte líquida do refrigerante que é separada do vapor é enviada ao evaporador através do dispositivo de expansão, além de resfriar o vapor superaquecido que sai do compressor de baixa pressão.

Enquanto o vapor, por outro lado, é comprimido até a pressão de condensação no compressor de alta pressão. Os sistemas que utilizam o separador de gás de “flash” reduzem a potência de compressão ao retirar vapor saturado, que não produz qualquer efeito de refrigeração, antes de passar pelo evaporador.

É importante destacar que esse resfriamento não necessariamente ocorre dentro do separador de gás de “flash”, podendo ocorrer na linha, quando o vapor que sai do compressor de baixa pressão encontra o vapor proveniente do separador.

2.1.3. Compressão em Duplo Estágio e Duas Temperaturas de Evaporação Com Separador de gás de “Flash”

A compressão em duplo estágio com duas temperaturas de evaporação é uma aplicação alternativa aos sistemas de compressão em estágio simples com duas temperaturas de evaporação (Figura 3).

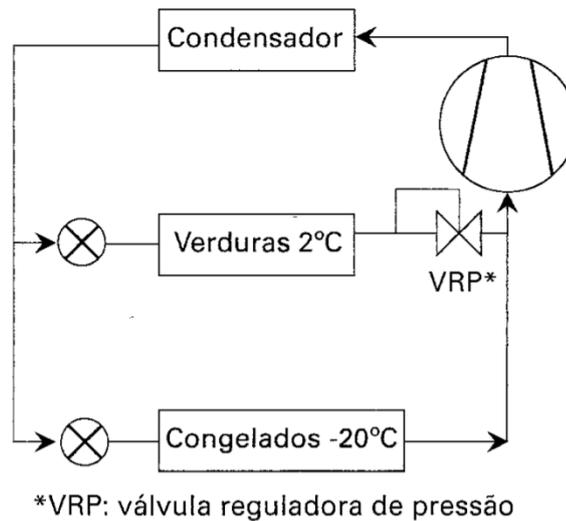


Figura 3 - Diagrama Esquemático do Sistema de compressão com estágio simples e duas temperaturas de evaporação

Fonte: Stoecker e Jabardo (2002)

Um dos problemas dos sistemas de compressão com estágio simples e duas temperaturas de evaporação é que o sistema impõe um estrangulamento na válvula reguladora de pressão (VRP) do vapor proveniente do evaporador de maior temperatura até a pressão do evaporador de baixa temperatura, o que não traz qualquer vantagem em relação à potência de compressão. Para solucionar esse problema, utiliza-se um sistema com compressão em duplo estágio e separador de gás de “*flash*”.

Nesse sistema, o vapor proveniente do evaporador de temperatura intermediária (maior temperatura de evaporação) é enviado para o separador de gás de “*flash*”, de onde o vapor saturado é enviado para o compressor de alta pressão e o líquido saturado continua para ser utilizado pelo evaporador de baixa temperatura. Dessa maneira, o sistema é eficiente em utilizar a potência de compressão já que não precisa estrangular o vapor proveniente do evaporador de temperatura intermediária até a pressão do evaporador de baixa temperatura, direcionando-o direto para o compressor de alta pressão. A Figura 4 apresenta um diagrama esquemático desse sistema.

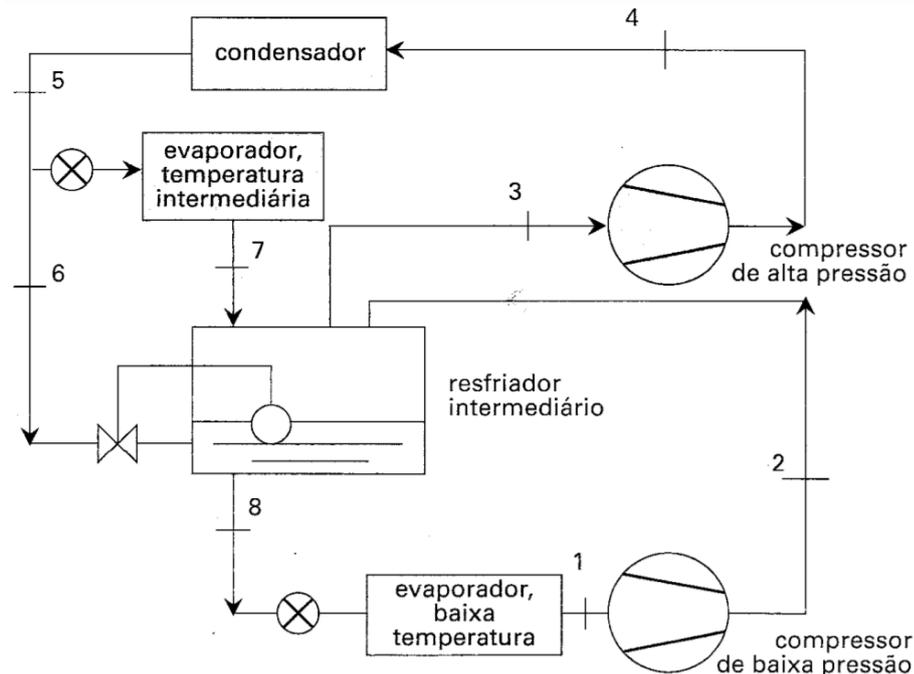


Figura 4 - Diagrama Esquemático do Sistema de compressão em duplo estágio e dois níveis de temperatura de evaporação com separador de gás de "flash"
Fonte: Stoecker e Jabardo (2002)

Existem casos ainda onde a temperatura de evaporação é extremamente baixa, exigindo sistema de compressão com mais de dois estágios. É o caso dos sistemas de compressão em triplo estágio. Nesse caso, Stoecker e Jabardo (2002) afirmam que o procedimento desenvolvido para os sistemas duplo estágio pode ser utilizado para o entendimento desses.

2.2. MODELAGEM E SIMULAÇÃO DE CICLOS DE REFRIGERAÇÃO UTILIZANDO SIMULADORES DE PROCESSOS

Brandão (2018) realizou a análise exergoeconômica e a simulação através do software IPSEpro 6.0 de um *chiller* de absorção duplo efeito LiBr-H₂O de um sistema de resfriamento distrital visando a possível implementação deste fornecimento centralizado na cidade do Rio de Janeiro. O cálculo da Carga Térmica foi realizado aos moldes de um shopping center localizado na Barra da Tijuca no Rio de Janeiro.

O IPSEpro 6.0 foi utilizado para o cálculo das entalpias de fornecimento e retorno da água, seguindo algumas hipóteses de modelagem, entre elas: temperatura constante do fornecimento e retorno da água, troca térmica com o solo desprezível, incompressibilidade da água entre outros. Entre outros passos, foi realizada uma simulação no IPSEpro 6.0 para cálculo

dos coeficientes globais de troca de calor da operação interna do *chiller*. Os resultados do estudo mostraram um aumento do custo específico da água gelada (U\$/kJ) e a diminuição da sua exergia (kW) com a diminuição da carga de operação (TR) do sistema de resfriamento distrital.

Ghaddar, Shihab e Bdeir (1997) realizaram a modelagem e simulação de um sistema de absorção solar para Beirute. Os resultados mostraram que para cada tonelada de refrigeração é necessária uma área coletora de 23,3 m² no mínimo, com uma capacidade de armazenamento de água variando de 1000 a 1500 litros, no ponto ótimo, quando o sistema opera exclusivamente com energia solar por cerca de 7 horas por dia. A fração solar mensal da energia total usada para resfriamento é determinada como uma função da energia solar por área do coletor e da capacidade do tanque de armazenamento.

Ahmadu, Folayan e Anafi (2016) realizaram a modelagem, simulação e otimização de um sistema de ar condicionado por absorção solar para um bloco de escritório cobrindo uma área total de 90 m² usando o software TRNSYS. A demanda de energia de resfriamento por hora do bloco de escritórios para todo o ano foi simulada usando o subprograma do TRNSYS, TRNbuild. Os resultados indicam que um *chiller* de absorção de 12 kW de capacidade, área de 40 m² de placa coletora e volume do tanque de armazenamento de água quente de 0,9 m³, o sistema atinge um COP máximo de 0,62, SCOP (coeficiente de performance solar) máximo de 0,116 e FS (Fração Solar) média anual de 0,79.

2.3. OTIMIZAÇÃO DE CICLOS DE REFRIGERAÇÃO

Mosaffa et.al (2016) realizaram a otimização de dois sistemas de refrigeração em cascata utilizando como fluídos de trabalho o dióxido de carbono e amônia (CO₂/NH₃). O primeiro com dois separadores de líquido e o segundo com dois separadores de líquido e um trocador de calor intermediário no circuito de baixa temperatura. A otimização, realizada com o objetivo de maximizar o COP, mostrou um aumento de 16,9% e 20,8% no COP para os sistemas 1 e 2, respectivamente, ao se aumentar em 10 K a temperatura de evaporação do CO₂. Além disso, o COP máximo foi alcançado com uma temperatura de condensação do CO₂ de 1,9 °C e 2,1 °C para os sistemas 1 e 2, respectivamente.

Bellos, Tzivanidis e Antonopoulos (2016) realizaram a otimização e comparação de 4 tipos de coletores solares de um sistema de refrigeração por absorção do tipo LiBr-H₂O. A otimização energética foi realizada para duas funções objetivo, que visou a maximização do COP Solar (SCOP) e minimização da área do coletor. O sistema com a melhor configuração

encontrada foi a do coletor de calhas parabólicas, que obteve um SCOP de 0,5647 e uma área de 295,2 m².

Baraldi (2015) realizou a otimização de um ciclo de refrigeração por compressão de uma Unidade de Processamento de Gás Natural. O ciclo foi previamente simulado no software Excel® utilizando a biblioteca Coolprop (Coolprop, 2010). A otimização gerou um aumento de 35% no COP (Coeficiente de Performance), passando de 2,66 para um COP de 3,61. A otimização foi realizada considerando a restrição imposta de que a temperatura final do gás natural, ou temperatura do gás natural após passar pelo ciclo de refrigeração, não sofresse variação maior que 2%.

Sadeghi, Mahmoudi e Saray (2015) realizaram a análise e otimização de um ciclo de refrigeração por jato-compressão acionado por um motor HCCI (*Homogeneous Charge Compression Ignition*). Foi realizada uma otimização multi-objetivo a fim de maximizar a eficiência exergética e minimizar o custo unitário do produto do sistema. As temperaturas encontradas após a otimização para as variáveis de decisão, gerador, condensador e evaporador foram respectivamente 94,54°C, 33,44°C e 0,03°C. Além disso, um aumento de 15,18% na eficiência exergética foi alcançado com somente 0,85% de aumento no custo total do produto.

Wang, Dai e Gao (2008) realizaram a análise paramétrica e otimização de um ciclo Rankine combinado com ciclo de refrigeração por absorção e fluido de trabalho amônia-água. A otimização dos parâmetros no ciclo é obtida utilizando a eficiência exergética como função objetivo por meio de algoritmo genético. A otimização atingiu uma eficiência exergética máxima de 43,06% quando a pressão de entrada da turbina, a temperatura de entrada da turbina e a concentração de amônia da solução básica foram iguais a 1,706 MPa, 280,4 °C e 0,305 kg de amônia/kg de mistura.

2.4. ALGORITMOS GENÉTICOS APLICADOS NA OTIMIZAÇÃO DE SISTEMAS TÉRMICOS

Wu et. al (2020) realizaram a otimização de um condensador de placa de um sistema de conversão de energia térmica oceânica (OTEC). Foi utilizada uma função composta (FC) a fim de otimizar simultaneamente a Taxa de geração de entropia e a Potência de Bombeamento Total (Psum) através de um algoritmo genético multi-objetivo, duas funções inversamente relacionadas em um condensador de placa. Como variáveis de decisão foram utilizados o

comprimento e a largura eficazes e o número eficaz de placas de transferência de calor. O resultado obtido para a FC apresenta uma redução de 7,8% na FC após a primeira otimização e de 9,9% após a segunda otimização, em comparação com o FC inicial.

Lee (2016) realizou a otimização de um ciclo de Rankine em cascata a fim de recuperar a energia descartada no processo de regaseificação de Gás Natural líquido (GNL) utilizando algoritmo genético. Os fluidos de trabalho etano (C_2H_6) e propano (C_3H_8) foram escolhidos como variáveis de decisão e a Potência Líquida (potência das turbinas menos potência das bombas) foi a função objetivo escolhida. Apesar do ciclo simples etano ter produzido levemente mais potência nas turbinas do que o ciclo etano-propano para a regaseificação de 1kg/s de GNL, o segundo foi o que apresentou a maior potência líquida (96,3 kW) devido ao menor consumo de energia nas bombas.

Rodrigues (2013) em seu projeto, realizou a otimização de um sistema complexo real de cogeração de energia, localizado na cidade do Rio de Janeiro, utilizando o simulador de processos IPSEpro e o seu módulo PSOptimize. A otimização foi realizada a partir de duas diferentes vertentes, a otimização Termodinâmica (Caso I) e a otimização termoeconômica (Caso II). A função objetivo do Caso I (eficiência global da planta) obteve ganho percentual de 8,3% para condições de 400 ton/h de vapor de processo exportado. Já no Caso II, função objetivo (Custo de Energia) obteve ganho percentual de 1,4%, para a condição de 100 ton/h de vapor de processo exportado. As demais condições termodinâmicas não apresentaram divergências significativas entre os dois casos devido “às restrições impostas pelo problema, às próprias condições operacionais que fixavam determinados valores e aos valores ótimos que eram intrínsecos ao sistema, independente do objetivo em questão.” (Rodrigues, 2013, pág 55).

Cunha (2016) utilizou o simulador de processos IPSEpro e o seu módulo PSOptimize para otimização de um ciclo de turbina a vapor que trabalha em superaquecimento com uma turbina de três estágios, dois pré-aquecedores e duas válvulas de expansão. O ciclo foi otimizado para duas funções objetivo diferentes, o trabalho líquido e o rendimento térmico e para sete pontos operacionais (casos I a VII). Em todos os casos, o ganho percentual no trabalho líquido ficou acima de 7%, atingindo o seu máximo (8,84%) no caso II (Pressão na saída da caldeira = 7500,00 kPa; Temperatura na saída da caldeira = 773 K; Pressão na saída da última turbina de expansão = 29,55 kPa; $\eta_{\text{turbina}} = 90\%$). Já para o rendimento térmico, houve mais variação, com ganho percentual na otimização acima de 4% em todos os casos, chegando à 7,74% no caso III (Pressão na saída da caldeira = 10000,00 kPa; Temperatura na saída da caldeira = 773 K; Pressão na saída da última turbina de expansão = 59,85 kPa; $\eta_{\text{turbina}} = 90\%$).

Maudonet (2017) realizou um estudo de caso de otimização através de algoritmos genéticos, utilizando o módulo PSEoptimize do IPSEpro e análise exergética como software MatLab em três diferentes sistemas térmicos: um ciclo Brayton, um ciclo Rankine com cogeração e um ciclo combinado com cogeração a partir da combinação dos dois primeiros sistemas. A otimização energética do Ciclo Brayton atingiu 56880 kW e 0,6101 contra 43200 kW e 0,5868 no caso padrão, para os parâmetros respectivos “Potência da turbina à gás” e “Rendimento Exergético global”.

Já para o Ciclo Rankine com Cogeração, a otimização atingiu 25500 kW e 0,2642 contra 18800 kW e 0,2566 no caso padrão para os respectivos parâmetros, “Potência da turbina de alta pressão” e “Rendimento exergético global”. Para o Ciclo Combinado com Cogeração, a otimização exergética atingiu 0,4527 contra 0,4209 no caso padrão para o parâmetro “Rendimento exergético global”.

3 REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR

Boles e Çengel (2011) definem refrigeração como uma das grandes áreas de aplicação da termodinâmica que corresponde à transferência de calor de uma região com menor temperatura para uma região com maior temperatura.

Boles e Çengel (2011) explicam ainda que os dispositivos que produzem refrigeração são os chamados refrigeradores e os ciclos nos quais eles operam são chamados de ciclos de refrigeração. O ciclo de refrigeração usado com mais frequência é o ciclo de refrigeração por compressão de vapor. O ciclo de refrigeração por compressão de vapor simples envolve quatro processos conhecidos como expansão, vaporização, compressão e condensação.

Esta seção abordará um breve histórico do sistema de refrigeração por compressão de vapor, bem como o ciclo de refrigeração associado ao problema de referência a ser estudado.

3.1. BREVE HISTÓRICO

Boles e Çengel (2011) consideram Jacob Perkins - um físico, inventor e engenheiro mecânico americano - como o precursor do sistema de refrigeração por compressão de vapor e destacam a patente recebida em 1834 pela criação de uma máquina de gelo que utilizava éter como refrigerante, mas que nunca chegou a ser comercializada.

O Portal Exame (2018) descreve ainda que os conceitos básicos da refrigeração como conhecemos hoje foram fundamentados anos antes pelo cientista e médico escocês William Cullen e indica sua primeira invenção documentada de 1756, um vácuo parcial sobre um recipiente com éter etílico.

Já em 1850, Alexander Twinnig começou a projetar e construir sistemas de refrigeração utilizando éter etílico e tais sistemas eram utilizados especialmente em máquinas de gelo e armazenamento a baixas temperaturas (Boles e Çengel, 2011). Quarenta anos depois, as máquinas passaram a deixar de ser movidas por motores a vapor e passaram a ser elétricas, tornaram-se mais leves e começaram a aparecer em açougues e residências.

Finalmente, em 1930, surgiram equipamentos de refrigeração por compressão de vapor baratos, leves e eficientes.

3.2. REFRIGERADORES E BOMBAS DE CALOR

A transferência de calor ocorre espontaneamente da região de maior temperatura para a região de menor temperatura. O processo inverso não ocorre de maneira espontânea e, para que este processo forçado possa ser realizado, dispositivos especiais devem ser utilizados. Esses dispositivos são chamados de refrigeradores.

Os refrigeradores trabalham com um fluido refrigerante que absorve o calor do espaço refrigerado. Outro dispositivo que transfere o calor de uma região de menor temperatura para uma região de maior temperatura é a chamada bomba de calor. A forma de operação das bombas de calor e dos refrigeradores é essencialmente igual, a finalidade de cada um desses dispositivos é o que os diferencia.

Boles e Çengel (2011) explicam que a finalidade dos refrigeradores é manter um espaço refrigerado, a uma baixa temperatura e, para tanto, promove a remoção de calor deste ambiente. A rejeição desse calor em um ambiente de temperatura mais alta é apenas parte necessária da operação dos refrigeradores. Por outro lado, as bombas de calor têm por finalidade manter um espaço aquecido a uma temperatura alta, o que é realizado através da absorção de calor de uma fonte fria - como o ar exterior do inverno, por exemplo - e fornecimento de calor a um ambiente mais quente, como uma casa.

Boles e Çengel (2011) apresentam ainda as formulações para o cálculo do desempenho dos refrigeradores e bombas de calor, que são representadas pelas Equações 1 e 2, respectivamente. Neste índice de desempenho, quanto maior o resultado, maior será a performance do equipamento.

$$\text{COP}_R = \frac{\text{Efeito desejado}}{\text{Entrada Necessária}} = \frac{\text{Resfriamento, por remoção de calor}}{\text{Entrada de trabalho}} \quad (1)$$

$$\text{COP}_{BC} = \frac{\text{Efeito desejado}}{\text{Entrada Necessária}} = \frac{\text{Aquecimento, por fornecimento de calor}}{\text{Entrada de trabalho}} \quad (2)$$

Na Equação 1, COP_R representa o Coeficiente de Performance de um refrigerador e na Equação 2 COP_{BC} representa o Coeficiente de Performance de uma bomba de calor.

3.3. O CICLO DE CARNOT REVERSO

Moran e Shapiro (2012) discorrem que o Ciclo de Carnot Reverso ou Ciclo de Refrigeração de Carnot possui basicamente 4 componentes – compressor, condensador, turbina e evaporador - formando um sistema no qual o fluido refrigerante circula de maneira contínua. Nos próximos parágrafos, serão apresentados os processos do Ciclo de Carnot Reverso de acordo com a Figura 5.

O evaporador absorve o calor do ambiente a ser refrigerado (processo 4→1 no diagrama T-s) e o fluido refrigerante entra no evaporador com um título variando entre 0 e 1, ou seja, uma mistura de líquido-vapor. Não há variação de temperatura e pressão entre a entrada e saída do evaporador, mas parte do refrigerante muda de fase de líquido para vapor, em função da transferência de calor do ambiente a ser refrigerado para o refrigerante.

Em seguida, o refrigerante é comprimido de forma isentrópica pelo compressor (processo 1→2 no diagrama T-s), e sai deste componente na forma de vapor saturado, provocando um aumento de temperatura no fluido.

O refrigerante passa então pelo condensador (processo 2→3 no diagrama T-s), onde muda de fase passando de vapor saturado para líquido saturado, em função da rejeição de calor para o ambiente externo. Neste processo temperatura e pressão permanecem constantes e o fluido é direcionado então para a turbina, sofrendo uma expansão isentrópica (processo 3→4 no diagrama T-s). A Figura 5 ilustra o Ciclo de Carnot Reverso e seu respectivo Diagrama T-s.

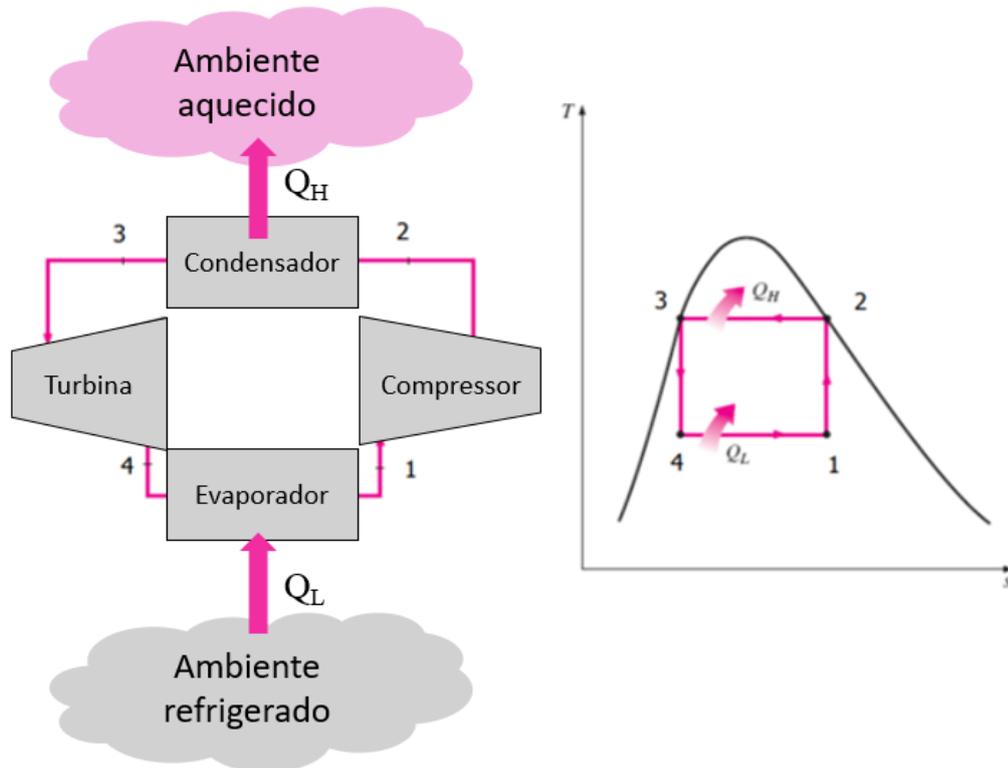


Figura 5 – Ciclo de Carnot Reverso
Fonte: Adaptada de Çengel e Boles (2011)

Contudo, os autores explicam que na prática os sistemas reais desviam-se do ciclo de Carnot reverso, em função de três razões principais. A primeira delas está relacionada às transferências de calor entre o fluido refrigerante e as regiões refrigerada e aquecida, uma vez que para que seja possível atingir a temperatura desejada no ambiente em sistemas reais, o fluido refrigerante precisa estar vários graus abaixo da temperatura do ciclo de Carnot e, de forma análoga, o refrigerante no condensador precisa estar muitos graus acima da temperatura do ambiente aquecido para alcançar a taxa de transferência de calor necessária para um desempenho aceitável do sistema.

O segundo motivo pelo qual o ciclo de Carnot reverso é inviável na prática está associado ao processo de compressão na região saturada (também conhecida como compressão molhada), quando o refrigerante entra no compressor como mistura de líquido-vapor. Isso porque a presença de gotas de refrigerante (fase líquida) pode danificar um compressor que lida apenas com vapor. Além disso, a utilização de compressores que lidam com duas fases é inviável do ponto de vista econômico, devido ao alto custo desse tipo de compressor.

Por fim, a terceira característica que torna o ciclo de Carnot reverso impraticável seria a baixa eficiência durante o processo de expansão, uma vez que as turbinas não operam de forma eficiente com vapor de baixo título, ou seja, alto conteúdo de umidade. Çengel e Boles (2011) relatam que a operação com vapor com título abaixo de 90% não é tolerada, uma vez que a colisão das gotas de líquido nas pás da turbina é uma importante fonte de desgaste do componente em questão. Além disso, a turbina é um componente caro e sua manutenção é complexa, o que inviabiliza a sua utilização de forma escalável por um valor acessível.

3.4. O CICLO IDEAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR

Uma vez identificadas as limitações práticas do ciclo reverso de Carnot, foi possível eliminá-los substituindo a turbina por uma válvula de expansão, reduzindo ainda os custos de implementação e manutenção. Moran e Shapiro (2011) desprezam a irreversibilidade no evaporador e no condensador, fazendo com que não se considere a queda de pressão por atrito. Considerando também a compressão isentrópica, a Figura 6 apresenta o esquema e o diagrama do ciclo ideal de refrigeração por compressão de vapor. Abaixo são apresentados os processos:

1-2: compressão isentrópica no compressor até a pressão do condensador;

2-3: transferência de calor no condensador do refrigerante para o ambiente aquecido, a pressão constante, até que o fluido saia na forma de líquido saturado do condensador;

3-4: estrangulamento em um dispositivo; o refrigerante sai como uma mistura líquido-vapor;

4-1: transferência de calor no evaporador a partir do ambiente refrigerado para o refrigerante a pressão constante. O fluido sai na forma de vapor saturado em direção ao compressor.

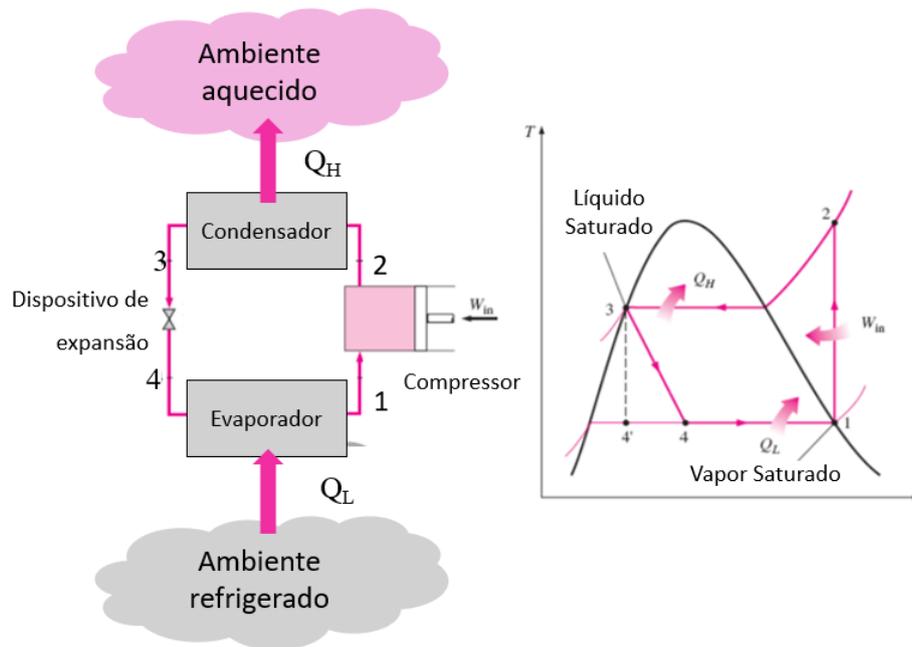


Figura 6 - Ciclo ideal de refrigeração por compressão de vapor
 Fonte: adaptada de Çengel e Boles (2011)

3.5. O CICLO REAL DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR

Çengel e Boles (2011) apontam que o atrito (que causa quedas de pressão) e a transferência de calor de ou para a vizinhança são duas fontes comuns de irreversibilidade. Além disso, diferente do que foi considerado para o ciclo ideal de refrigeração por compressão de vapor, na prática não é possível desconsiderar as irreversibilidades, assim como não é possível controlar de forma tão precisa como no modelo ideal o estado do refrigerante após sair do evaporador e do condensador.

Desta forma, é necessário incluir esses fatores na modelagem deste ciclo para representar melhor as condições reais de operação do ciclo de refrigeração por compressão de vapor. Portanto, neste modelo termodinâmico mais próximo à operação real, considera-se que o refrigerante fica ligeiramente superaquecido na entrada do compressor, o que garante uma segurança de que o fluido entrará totalmente vaporizado em tal componente, evitando assim a compressão molhada. De forma análoga, faz-se necessário também o sub-resfriamento antes do refrigerante entrar na válvula de expansão. Vale ressaltar que as quedas de pressão causadas pelo atrito no condensador e no evaporador provocam as pequenas quedas da temperatura de saturação do refrigerante notadas no diagrama T-s do ciclo real de refrigeração por compressão

de vapor apresentado na Figura 7, que também mostra o esquema do ciclo real de refrigeração por compressão de vapor.

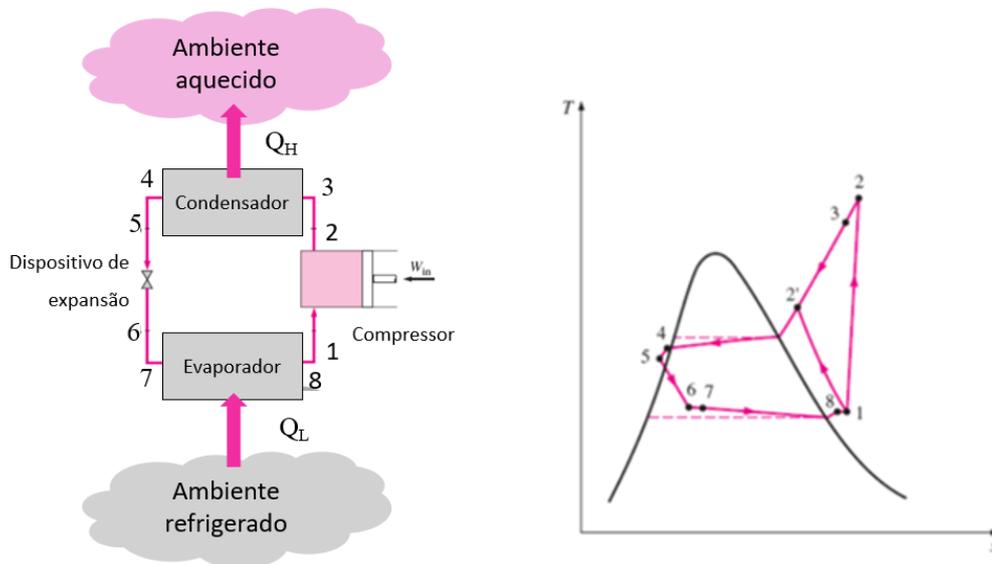


Figura 7 - Ciclo real de refrigeração por compressão de vapor

Fonte: adaptada de Çengel e Boles (2011)

3.6. SELEÇÃO DO FLUIDO REFRIGERANTE

Entre os diversos tipos de fluidos refrigerantes existentes no mercado – HCFCs (hidroclorofluorcarbonos), HFCs (hidrofluorcarbonos), amônia, dióxido de carbono, hidrocarbonos (propano, isobutano, etano, etileno) etc. – é importante compreender em que contexto a utilização do refrigerante está inserida para que seja realizada a escolha apropriada do fluido.

Boles e Çengel (2011) afirmam que o éter etílico foi o primeiro refrigerante a ser utilizado comercialmente em sistemas de compressão de vapor em 1850. No dias atuais, os autores discorrem que, embora tóxica, a amônia é amplamente utilizada nos setores industriais, isso porque quando comparada a outros refrigerantes seu custo é baixo; suas propriedades termodinâmicas e de transporte são mais favoráveis, promovendo coeficientes de troca de calor mais altos e possibilitando, assim, trocadores de calor menores; os coeficientes de performance são mais altos, logo o consumo de energia elétrica é menor; a facilidade de detecção de

vazamentos é maior quando comparada a outros fluidos; e ela não traz impactos negativos à camada de ozônio. Comparativamente, Boles e Çengel (2011) afirmam ainda que de maneira geral o uso do resfriamento a água só é economicamente justificável em grandes sistemas industriais de refrigeração.

Os clorofluorcarbonos (CFCs) geram efeitos negativos para a camada de ozônio e, por isso, não podem mais ser utilizados em vários países do mundo. No entanto, ao longo da história e mesmo nos dias atuais, refrigerantes desta família apresentaram grande relevância, especialmente para usos domésticos. Nos anos de 1920, devido a vários acidentes graves que implicaram até mesmo em mortes causados por vazamentos de refrigerantes tóxicos, houve a necessidade de desenvolvimento de um refrigerante seguro para uso doméstico. Em 1928, foi definido o R-12 como principal fluido para este tipo de finalidade, mas diversos outros fluidos da família CFC foram desenvolvidos e receberam o nome de “freon”.

O freon R-11 foi utilizado em refrigeradores de água gelada instalados em sistemas de condicionamento de ar de edifícios; o R-22 teve sua principal aplicação em aparelhos de ar condicionado de janela, bombas de calor e grandes sistemas de refrigeração industrial; e outros fluidos² dessa mesma família tiveram outras aplicações, como supermercados, refrigeração comercial etc.

Com as ações humanas sobre o meio ambiente e o impacto da família freon sobre a camada de ozônio e o efeito estufa, o uso de CFCs vem sendo reduzido e mesmo eliminado em tratados internacionais. Desta forma, fica evidente que não só o desempenho técnico para a aplicação deve ser levado em conta, mas os efeitos sobre a saúde humana e ambiental também são fatores relevantes para a escolha do refrigerante adequado. Assim, é importante que o fluido não seja tóxico, corrosivo, que apresente uma alta entalpia de vaporização e tenha um custo acessível.

Sob a ótica do desempenho técnico, Boles e Çengel (2011), discorrem que as temperaturas dos meios (espaço a ser refrigerado e o ambiente de rejeição de calor) com que o refrigerante irá trocar calor são parâmetros muito importantes para a escolha do refrigerante adequado. Isso porque nos trocadores de calor de sistemas de refrigeração, o refrigerante deve sair do evaporador de forma completamente vaporizada para não provocar danos no compressor

² O R-502, uma combinação entre R-115 e R22, foi amplamente utilizado no setor comercial porque permite baixas temperaturas nos evaporadores e opera a compressão de um único estágio. O R-12 foi amplamente utilizado em condicionadores de ar automotivos, além de refrigeradores domésticos.

e para promover um funcionamento eficiente o fluido deve sair com título zero ou sub resfriado do condensador.

3.7. CICLOS DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR MODIFICADOS

Em aplicações especiais em que se deseja aumentar o desempenho do ciclo de compressão de vapor, é possível utilizar algumas variações do ciclo básico de compressão de vapor. O enfoque maior da utilização dos ciclos de compressão de vapor modificados é encontrado na indústria, onde o aperfeiçoamento do desempenho se sobrepõe a outras características, como a simplicidade do sistema. Esta seção irá abordar o ciclo de compressão em cascata e o ciclo de compressão de vapor em múltiplos estágios, sendo este o objeto de estudo deste trabalho de conclusão de curso.

3.7.1. CICLOS DE REFRIGERAÇÃO EM CASCATA

Uma combinação de ciclos de refrigeração utilizada na forma de um sistema único é chamada de ciclo de refrigeração em cascata. Çengel e Boles (2011) afirmam que normalmente arranjos com essa configuração são utilizados na indústria, onde existem grandes intervalos de temperatura e portanto, a utilização de dois ou mais ciclos em série é uma solução para possíveis problemas que poderiam ser causados com tais intervalos de temperatura. Isso porque, um grande intervalo de temperatura significa um grande intervalo de pressão no ciclo e um desempenho ruim de um compressor alternativo (Çengel e Boles, 2011).

Um ciclo de refrigeração em cascata pode ter dois ou mais ciclos em série que são unidos através de um trocador de calor intermediário, denominado trocador de calor contracorrente. Neste trocador intermediário, o calor rejeitado pela condensação no ciclo de baixa pressão é utilizado para evaporar o fluido refrigerante no ciclo de alta pressão. Assim, a baixa temperatura desejada é obtida no ciclo de baixa pressão e a rejeição de calor do ciclo de refrigeração em cascata é encontrada no condensador de alta temperatura. No sistema de refrigeração em cascata os refrigerantes dos ciclos conectados pelo trocador de calor podem ou não ser iguais. A Figura 8 apresenta um ciclo em cascata com dois estágios.

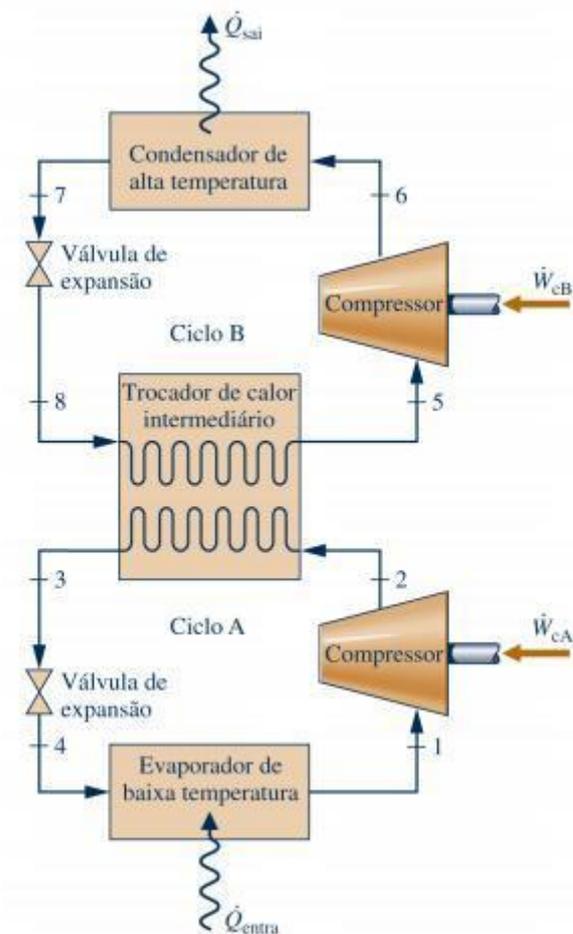


Figura 8 - Ciclo ideal de compressão de vapor em cascata
Fonte: Moran e Shapiro (2012)

A Figura 9 ilustra o diagrama T-s, indicando claramente como o trabalho do compressor é reduzido e a capacidade de refrigeração aumentada ao utilizar este tipo de sistema de refrigeração.

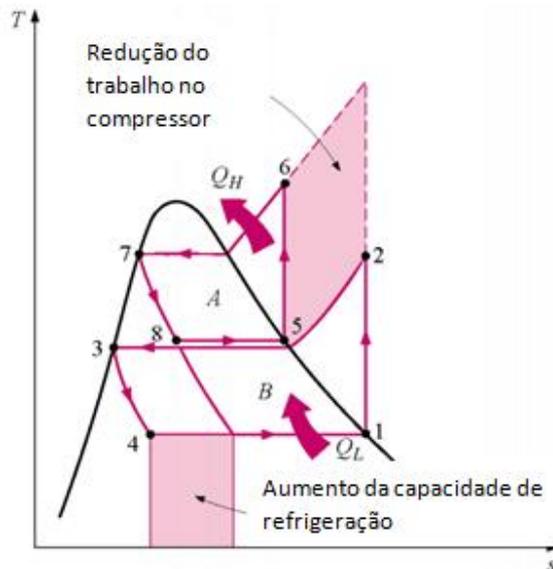


Figura 9 - Diagrama T-s do ciclo de compressão de vapor em cascata
Fonte: Çengel e Boles (2011)

A Figura 10 ilustra um sistema de refrigeração real com ciclo de refrigeração em cascata que utiliza CO_2 como refrigerante no estágio de baixa pressão e amônia como refrigerante no estágio de alta pressão. Este sistema de refrigeração se trata de uma unidade (*skid*) para uma planta de liquefação de CO_2 com capacidade de 100 ton/dia.

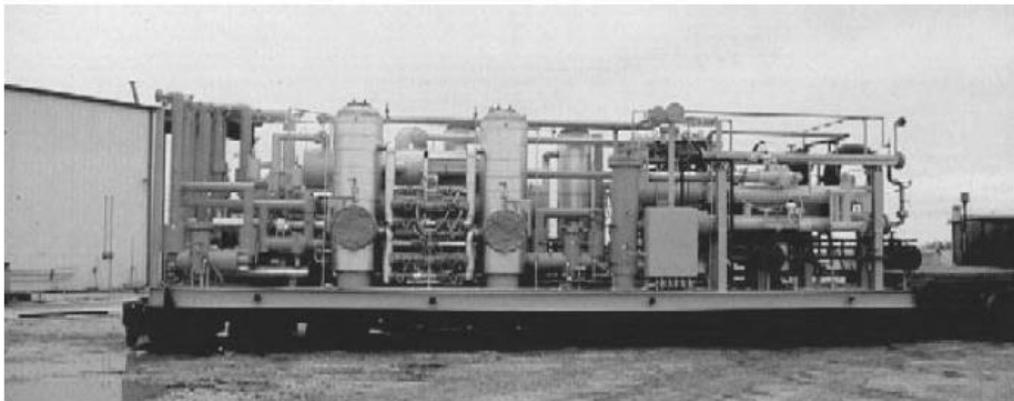


Figura 10 – Sistema de refrigeração real com ciclo de refrigeração em cascata
Fonte: Dinçer e Ibrahim (2010)

3.7.2. CICLOS DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS

Quando o mesmo fluido refrigerante é utilizado em todo ciclo de refrigeração em cascata, o trocador de calor que conecta os estágios pode ser substituído por um separador de líquido. Esta substituição é justificada pela superioridade das características de transferência de calor do separador de líquido em comparação com um trocador de calor que não permite a mistura das correntes que transferem calor. Este ciclo de refrigeração que utiliza apenas um fluido refrigerante e que possui um separador de líquido é chamado de ciclo de refrigeração por compressão em múltiplos estágios.

Threlkeld, Ramsey e Kuehn. (1998) mencionam que os ciclos de refrigeração por compressão de vapor em dois estágios que usam amônia como fluido refrigerante são frequentemente utilizados em aplicações industriais e comerciais. Na indústria de alimentos, por exemplo, este ciclo refrigera câmaras frias (frigoríficas) para armazenamento de sorvetes e túneis de congelamento. Além disso, os ciclos de refrigeração por compressão de vapor em dois estágios que usam Refrigerante 22 são amplamente utilizados em câmaras de testes de baixa temperatura.

Çengel e Boles (2011) explicam que neste ciclo de refrigeração, a mistura líquida - vapor de fluido refrigerante se expande na primeira válvula de expansão até entrar no separador de líquido, onde as fases líquido e vapor são separadas em duas correntes. O líquido saturado expande-se novamente na segunda válvula de expansão para em seguida evaporar-se no evaporador e sua pressão ser elevada no compressor de baixa pressão. O vapor saturado se mistura com o vapor que sai do compressor de baixa pressão e esta mistura segue em direção ao compressor de alta pressão. Em seguida, o calor do fluido refrigerante superaquecido é rejeitado no ambiente pelo condensador, reiniciando o ciclo de refrigeração em múltiplos estágios. A Figura 11 ilustra um ciclo de refrigeração por compressão de vapor em múltiplos estágios.

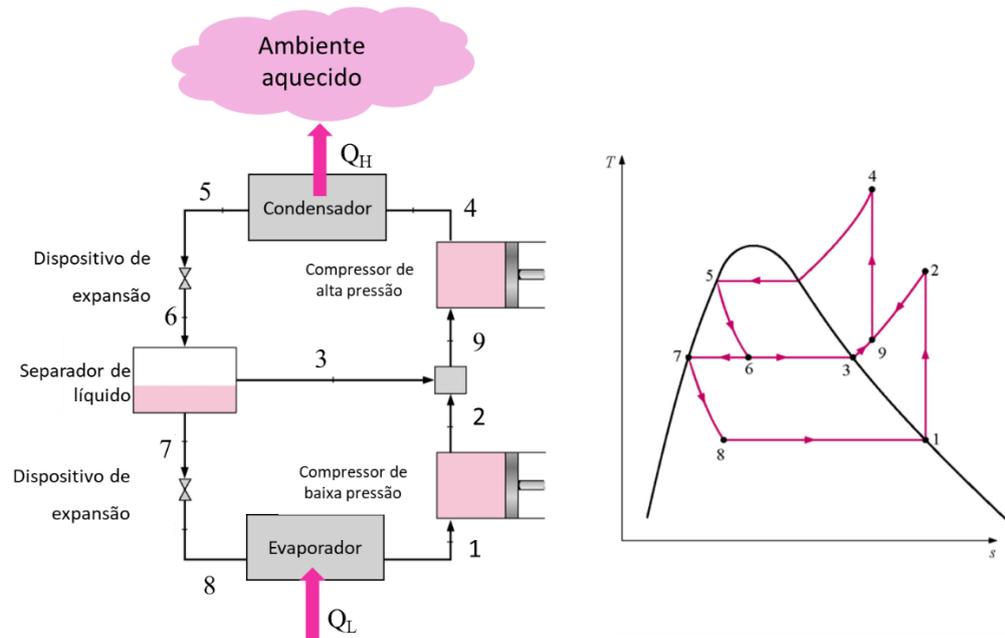


Figura 11 - Ciclo ideal de refrigeração por compressão de vapor em múltiplos estágios
Fonte: adaptada de Çengel e Boles (2011)

3.7.3. MODELAGEM ENERGÉTICA DE UM CICLO DE DOIS ESTÁGIOS COM UM SEPARADOR DE LÍQUIDO

Conforme descrito anteriormente na Equação 1 o coeficiente de performance é obtido através da relação entre o resfriamento por remoção de calor e a entrada de trabalho no sistema. Desta forma, faz-se necessário identificar a taxa de remoção de calor do ambiente refrigerado e a entrada de potência no compressor³.

Para tanto, utilizaremos o diagrama T-s, encontrado na Figura 11 que também ilustra o ciclo de refrigeração por compressão de vapor em duplo estágio. Os ciclos de baixa pressão e alta pressão estão conectados por um separador de líquido.

No sistema de refrigeração com compressão de vapor em múltiplos estágios, após a expansão do refrigerante no primeiro dispositivo de expansão, o refrigerante entra no separador de líquido como uma mistura de duas fases (líquido-vapor), apresentando um título $x_{título}$. O vapor saturado que sai do separador de líquido é misturado ao vapor superaquecido que sai do

³ Essa modelagem é feita pelo IPSEpro. No entanto, podem haver algumas diferenças entre os modelos do IPSEpro e os modelos apresentados adiante nesta subseção. Os modelos do IPSEpro serão apresentados no anexo A do trabalho.

compressor de baixa pressão e o líquido saturado que sai do separador de líquido é direcionado para o segundo dispositivo de expansão.

Com base na Figura 11, Çengel e Boles (2011) explicam que a fração do vapor formado no separador de líquido é igual ao título ($x_{título}$) do refrigerante no estado 6. Assim, a fração de líquido formada é igual a $(1 - x_{título})$. A Equação 3 indica o título $x_{título}$ no estado 6.

$$x_{título,6} = \frac{h_{l6} - h_l}{h_{lv}} \quad (3)$$

Onde h_{l6} é a entalpia do líquido saturado na pressão P_6 , h_l é a o valor de entalpia do líquido saturado para a pressão de operação do separador de líquido e h_{lv} é a entalpia para mudança de fase na pressão em que o separador de líquido opera.

A Equação 4 é adotada para identificar a quantidade de calor removida do espaço refrigerado.

$$Q_L = (1 - x_{título,6}) \times (h_{g1} - h_8) \quad (4)$$

Por se tratar de um processo sem variação de entalpia, a entalpia no ponto 8 (h_8) é igual a h_7 , ou seja, entalpia do líquido saturado na pressão P_7 . Na equação acima, h_{g1} é a entalpia do vapor saturado na pressão P_1 .

A Equação 5 indica o trabalho total realizado pelo sistema e a Equação 6 calcula a entrada de trabalho no compressor por unidade de massa do refrigerante.

$$W_e = W_{compressor\ alta\ pressão} + W_{compressor\ baixa\ pressão} \quad (5)$$

$$W_e = (1 - x_{título,6}) \times (h_2 - h_{g1}) + (h_4 - h_9) \quad (6)$$

Onde h_2 é a entalpia do vapor superaquecido na pressão P_2 . Para que seja possível encontrar a entalpia no estado 9, é realizado um balanço de energia no separador de líquido, de acordo com a Equação 7.

$$h_9 = x_{título,6}h_3 + (1 - x_{título,6})h_2 \quad (7)$$

Na equação acima h_3 é a entalpia do vapor superaquecido na pressão P_3 . Com o valor da entalpia na pressão P_9 é possível encontrar a entropia no estado 9 (s_9), como mostra a Equação 8.

$$s_9 = x_{título,6} s_3 + (1 - x_{título,6}) s_2 \quad (8)$$

Sabe-se ainda que S_9 é igual a S_4 (entropia do estado 9 é igual a entropia do estado 4) e, de posse do valor da pressão de condensação, é possível encontrar na tabela do estado superaquecido do refrigerante em questão o valor da entalpia no estado 4. Com isso, todas as informações para obtenção do trabalho requerido pelo sistema estão disponíveis. Assim, é possível encontrar o COP, como já apresentado Equação 1.

3.7.4. CICLOS DE REFRIGERAÇÃO COM MÚLTIPLOS PROPÓSITOS EM UM ÚNICO COMPRESSOR

De acordo com Çengel e Boles (2011), os ciclos de refrigeração com múltiplos propósitos em um único compressor é uma alternativa apropriada para sistemas que possuem diferentes evaporadores operando a diferentes temperaturas.

Um exemplo disso é encontrado nos supermercados onde um evaporador visa resfriar os alimentos a 7°C e outro visa congelar os alimentos a -30°C . Para solucionar demandas como esta, é possível desenvolver sistemas independentes, cada um atendendo a uma temperatura de evaporação.

No entanto, essa alternativa nem sempre é viável economicamente e de modo a viabilizar essa operação a temperaturas distintas de evaporação, é possível utilizar um sistema de refrigeração utilizando dois evaporadores e um único compressor. As diferentes correntes de saída dos evaporadores são direcionadas para um único compressor que realizará todo o processo de compressão do sistema. A Figura 12 ilustra este ciclo de refrigeração.

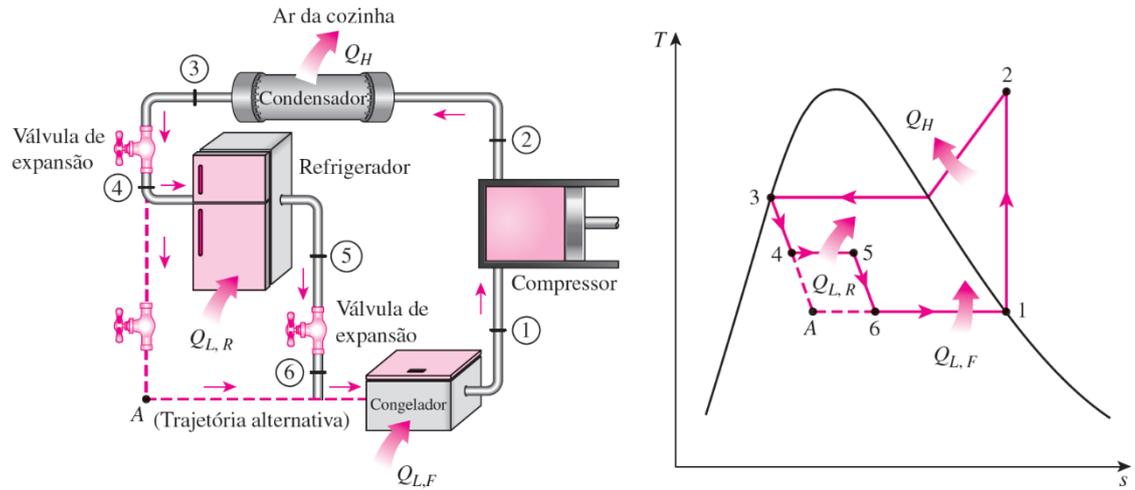


Figura 12 - Ciclo de refrigeração com múltiplos propósitos e um único compressor
Fonte: Çengel e Boles (2011)

4 INTRODUÇÃO À OTIMIZAÇÃO

Esta seção irá apresentar os conceitos básicos de otimização que, de acordo com Jaluria (2007) envolvem a maximização ou a minimização de características ou quantidades de parâmetros de interesse, considerando uma série de restrições, dada uma condição inicial.

Para tanto, iniciará apresentando as ideias básicas de otimização, sua formulação geral, a classificação dos principais métodos de otimização e, por fim, explorará os conceitos básicos dos algoritmos genéticos, método de otimização utilizado para alcançar os objetivos deste estudo. A otimização é uma continuação natural do processo de desenvolvimento de um projeto e pode ser definida como o processo de determinação das melhores condições de um projeto, entre todas as condições aceitáveis.

4.1. IDEIAS BÁSICAS

Jaluria (2007) aponta que, de maneira geral, muitos parâmetros afetam a performance e custos de um sistema térmico e uma solução ótima pode ser obtida em termos quantitativos como custos, eficiência, consumo energético por unidade de produto acabado (em qualquer segmento que demande refrigeração em processo produtivo), etc.

As configurações de operação realísticas são obtidas para todas as faixas possíveis de variáveis do sistema, levando em consideração as restrições e imposições existentes. Normalmente, uma solução única não é obtida e diferentes possibilidades de projeto podem ser elaboradas para cada aplicação avaliada. Por outro lado, a configuração ótima está associada à melhor solução possível que maximize ou minimize um parâmetro específico do problema. Importante apontar que muitas vezes o resultado ótimo de uma variável não é encontrado padronizado no mercado, fazendo com que na prática a escolha final dos parâmetros do sistema esteja a mais próxima possível do resultado ótimo.

Nesse sentido, outro conceito importante é o do domínio de configurações aceitáveis (“*domain of acceptable designs*” ou “*workable designs*”). A Figura 13 (representa esse conceito quando, para um problema de otimização de um sistema térmico qualquer, temos duas variáveis de decisão x_1 e x_2 . Os limites possíveis das duas variáveis definem a região denominada “*workable designs*” ou “*domain of acceptable designs*”. Qualquer projeto dentro

dessa região é um projeto viável, no entanto na prática estamos interessados no melhor projeto (projeto ótimo).

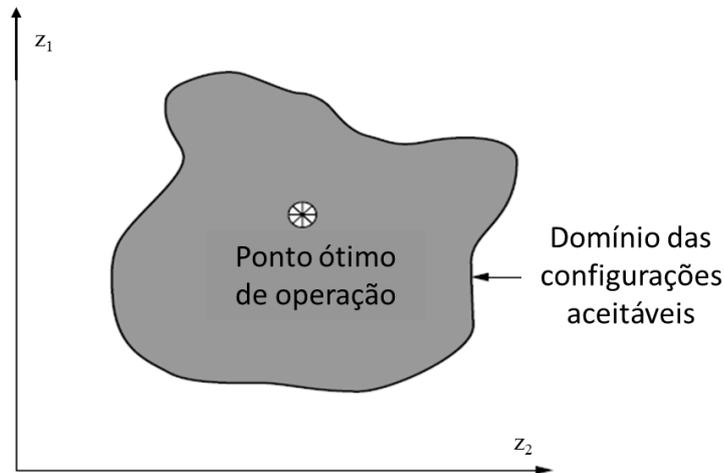


Figura 13 - Ilustração do conceito de configurações aceitáveis ou “workable designs”
Fonte: Jaluria (2007)

No exemplo acima, x_1 e x_2 podem ser, por exemplo, as dimensões de um parafuso (diâmetro e comprimento) para que a resistência ao escoamento (função objetivo) seja maximizada. No caso de um sistema térmico, x_1 e x_2 são propriedades termodinâmicas (pressão e temperatura) e/ou índices de desempenho (eficiência isentrópica) para que a eficiência do ciclo seja maximizada. No caso de refrigeradores, para que o Coeficiente de Performance (COP) seja maximizado. Outro exemplo ainda, é o caso onde x_1 e x_2 são parâmetros de um sistema qualquer de engenharia para o qual o custo total deve ser minimizado.

Uma vez que o modelo ótimo deve respeitar uma série de restrições e devem ser consideradas todas as possibilidades aceitáveis de arranjos entre as variáveis, é fundamental que se tenha conhecimento sobre como identificar as possíveis configurações de modelos aplicáveis ao sistema.

4.2. FORMULAÇÃO GERAL DE UM PROBLEMA DE OTIMIZAÇÃO

Conforme mencionado anteriormente, Jaluria (2007) estabelece que qualquer processo de otimização exige que se tenha uma função que deve ser maximizada ou minimizada e essa função é denominada Função Objetivo.

Para um cenário que visa otimizar o parâmetro U , onde U é função de n variáveis independentes, $x_1, x_2, x_3, \dots, x_n$, a função objetivo pode ser expressa como indica a Equação 9.

$$U = U(x_1, x_2, x_3, \dots, x_n) \Rightarrow U^* \quad (9)$$

Onde U^* é o resultado ótimo encontrado para U . Desta forma, são considerados todos os modelos aplicáveis para que determinado parâmetro seja maximizado ou minimizado sem que as restrições do problema sejam violadas. A Figura 14 ilustra o ponto ótimo (máximo global) da função objetivo U , considerando a faixa de valores possíveis para a variável independente x_1 .

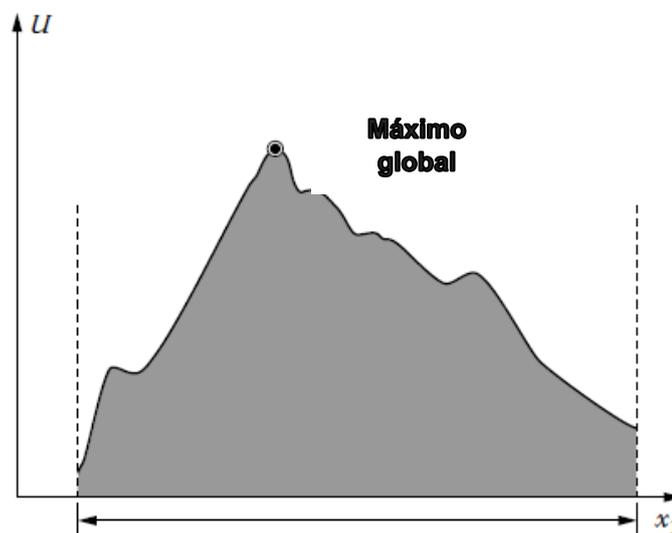


Figura 14 - Ponto ótimo da função objetivo U , considerando os valores possíveis para a variável independente x_1

Fonte: adaptada de Jaluria (2007)

De modo geral, Jaluria (2007) explica que a não linearidade das equações que modelam os sistemas térmicos faz a função objetivo desses sistemas apresentar vários máximos

e mínimos locais no domínio de configurações aceitáveis. Como o interesse maior é o máximo (ou mínimo) global, é necessário estudar cada extremo para assegurar que o ótimo global foi obtido. Isso é representado pela Figura 15.

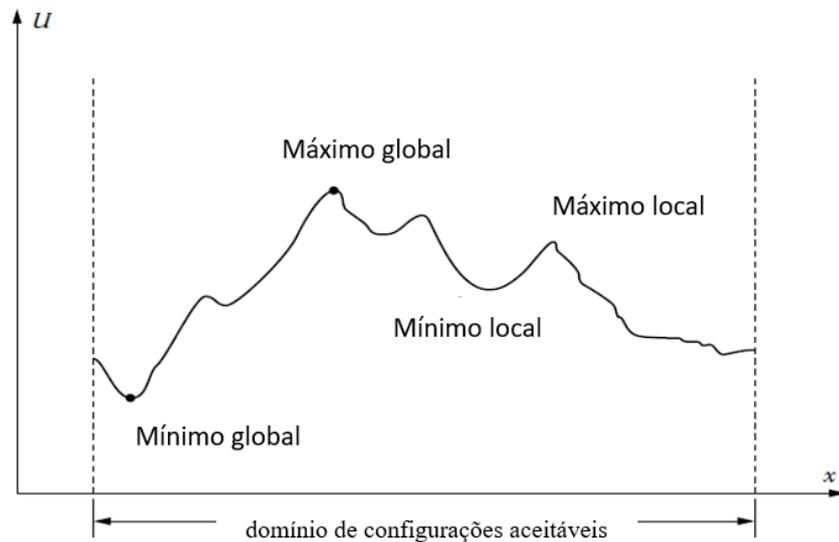


Figura 15 - Máximo e mínimo local e global dentro do domínio de configurações aceitáveis de um problema.
Fonte: Jaluria (2007)

Além da função objetivo, todo problema de otimização envolve as suas respectivas restrições, ou seja, restrições associadas aos limites das variáveis de decisão (variáveis independentes que podem ser modificadas no processo de otimização da função objetivo) e as restrições associadas às limitações impostas por princípios básicos de conservação. No presente trabalho as restrições são os limites inferior e superior das variáveis de decisão e os balanços de massa e energia.

De forma geral, as restrições que envolvem as leis de conservação estão associadas com massa, momento e energia. Assim, a massa que entra em um sistema deve ser igual a massa que sai desse sistema. Esta condição implica em uma formulação de função objetivo que considere essa restrição de valor da variável massa de modo a atingir um resultado ótimo que não viole princípios básicos da física. As restrições também podem envolver limitações de operação como espaço, parâmetros de funcionamento do equipamento, faixa de temperatura aceitável, materiais empregados etc.

Uma das vantagens de se utilizar um simulador de processos é que as restrições que envolvem as leis de conservação são tratadas pelo próprio algoritmo do simulador e não precisam ser escritas de maneira explícita ou como parte da função objetivo.

4.2.1. Formulação Geral De Um Problema De Otimização

Nesta seção apresentamos a formulação matemática de um problema de otimização, como indicam as Equações 10, 11 e 12.

$$\text{máx/ min } U(x_1, x_2, x_3, \dots, x_n) \quad (10)$$

$$G_i = G_i(x_1, x_2, x_3, \dots, x_n) = 0, \text{ para } i = 1, 2, 3 \dots m \quad (11)$$

$$H_i = H_i(x_1, x_2, x_3, \dots, x_n) \leq \text{ ou } \geq C, \text{ para } i = 1, 2, 3, \dots, l \quad (12)$$

Onde C é uma constante ou função conhecida.

As 3 equações representam o objetivo de maximizar⁴ ou minimizar uma função objetivo U , estando sujeito a m restrições de igualdade (G) e restrições de desigualdade (H)⁵.

Os vários passos envolvidos na formulação de um problema de otimização são:

- 1) Determinação das variáveis de decisão;
- 2) Seleção e definição da função objetivo, U ;
- 3) Determinação das restrições de igualdade, G ;
- 4) Determinação das restrições de desigualdade, H ;

⁴ Maximizar a função U equivale a minimizar a função $-U$.

⁵ É mais difícil lidar com restrições de desigualdade, e não existem métodos matemáticos generalizados para a solução simultânea de inequações. Em alguns casos, é possível transformar as restrições de desigualdade em restrições de igualdade, caso seja conveniente.

5) Conversão das restrições de desigualdade para restrições de igualdade, se apropriado.⁶

O sucesso da otimização depende muito da seleção das variáveis de decisão e da função objetivo. A complexidade do problema é diretamente influenciada pelo número de variáveis independentes, principalmente para sistemas com comportamento não linear. Além disso, a otimização é normalmente realizada para um conjunto reduzido de variáveis de decisão consideradas mais importantes. Ainda, a função objetivo deve ser sensível às variáveis de decisão utilizadas no projeto e em geral são realizadas manipulações algébricas para a redução do número de restrições do problema.

4.3. CLASSIFICAÇÃO DOS MÉTODOS DE OTIMIZAÇÃO

Existem muitos métodos que podem ser empregados para solucionar um problema de otimização – composto por uma função objetivo, restrições de igualdade e desigualdade e variáveis de decisão. No entanto, existem técnicas mais apropriadas para cada particularidade e outras que podem ou não ser aplicadas para condições específicas do problema. Os métodos de otimização a seguir foram descritos por Rodrigues (2013) e Maudonnet (2017).

Os métodos de otimização podem ser classificados como métodos determinísticos – *Deterministic Optimization*, métodos de busca direta – *Direct Search Algorithms*, ou como métodos randômicos – *Random Strategies* (Saramago, 2008).

4.3.1 Métodos Determinísticos

Os métodos de otimização baseados nos algoritmos determinísticos geram uma sequência determinística de possíveis soluções requerendo, na maioria das vezes, o uso de pelo menos a primeira derivada da função objetivo em relação às variáveis de projeto.

⁶ Os passos 1 a 5 não foram desenvolvidos no presente trabalho, e sim foram definidos com base em um estudo similar encontrado na literatura (Ahmadi, Dincer e Rosen, 2017). No entanto, o desenvolvimento das etapas 1 a 5 seria inviável para um projeto de graduação, devido ao aumento considerável da complexidade e do escopo do presente trabalho. Ainda, as análises de sensibilidade (estudos paramétricos) realizadas na seção 8 do presente trabalho foram propostas no estudo similar encontrado na literatura e são realizadas para a verificação e validação do algoritmo, bem como para o aprendizado de programação e estudos paramétricos. O objetivo principal do presente trabalho é a aplicação de um método para a otimização de um sistema térmico de interesse.

Nestes métodos, a função objetivo e as restrições são dadas como funções matemáticas e relações funcionais. Além disso, a função objetivo deve ser contínua e diferenciável no espaço de busca (Rodrigues, 2013).

Os métodos determinísticos apresentam teoremas que garantem a convergência para uma solução ótima, mas que não necessariamente é a solução ótima global. Como nesses métodos a solução encontrada é extremamente dependente do ponto de partida, pode haver convergência para um ótimo local. Dessa maneira, não são indicados para otimizar funções multimodais, isto é, funções que possuem vários ótimos locais.

Basicamente, os métodos determinísticos estão divididos de acordo com as características da função objetivo e restrições de projeto. Quando a função objetivo e as restrições são funções lineares das variáveis de projeto, os métodos são considerados de Programação Linear – assim como o algoritmo Simplex, que é o método mais tradicional para solucionar este tipo de problema de otimização. Quando se trata de problemas que envolvem função objetivo, ou pelo menos uma das restrições, de característica não-linear das variáveis de projeto, os métodos são considerados de Programação Não-Linear. Nesta classe, os métodos que mais se destacam são: Método de Programação Linear Sequencial, Método de Programação Quadrática Sequencial, Método das Direções Viáveis e Método do Gradiente Reduzido, entre outros (Rodrigues, 2013).

Em geral, os métodos determinísticos exigem elevado nível de processamento para programação computacional, o que motiva a utilização de outros métodos de otimização, como os Métodos Randômicos, que são relativamente mais simples.

4.3.2 Métodos de Busca Direta

Dentro do amplo conjunto dos métodos conhecidos como *derivative-free* (métodos que não calculam derivadas) estão os métodos de busca direta. Este método refere-se a um método iterativo, no qual um conjunto de pontos é testado a cada iteração e somente as avaliações da função objetivo são utilizadas para definir a aproximação seguinte.

Os algoritmos de busca direta são de crescente interesse por apresentarem, em geral, bons resultados teóricos de convergência, por serem fáceis de implementar e por representarem uma alternativa em ocasiões onde métodos mais elaborados falham (Rodrigues, 2013).

Dentre os métodos de busca direta, destacam-se os métodos de Hooke-Jeeves, Powell e Nelder-Mead – que está entre os mais utilizados em anos de computação científica (Rodrigues, 2013).

4.3.3 Métodos Randômicos

Os métodos randômicos apresentam eventos aleatórios em seus algoritmos matemáticos, e dentre eles, podem ser citados o método do Recozimento Simulado – *Simulated Annealing*, o método do Enxame de Partículas – *Particle Swarm*, e os Algoritmos Genéticos – *Genetic Algorithms*. Estes métodos baseiam-se na observação de processos da natureza e não apresentam fundamentação matemática teórica (Rodrigues, 2013).

Assim como os métodos de busca direta, não é necessária uma função contínua ou diferenciável. Portanto, estes algoritmos apresentam caráter genérico, podendo ser aplicados a problemas com função objetivo discreta, contínua ou não diferenciável, e por isso são considerados métodos de ordem zero. Além disso, os métodos randômicos conseguem evitar a convergência prematura da solução para pontos de mínimos locais.

Os Algoritmos Genéticos baseiam-se na evolução biológica. Eles simulam os mecanismos de desenvolvimento natural de populações de organismos vivos (a seleção natural, a hereditariedade, a mutação e a recombinação) que favorecem a sobrevivência dos indivíduos mais aptos. No entanto, em um Algoritmo Genético, cada indivíduo da população é caracterizado por um conjunto de variáveis de decisão representadas por *strings* de variáveis binárias. Essas *strings* correspondem aos cromossomos. Além disso, a aptidão de cada indivíduo é medida pelo valor da sua função objetivo.

De forma resumida, o método parte de uma população inicial de pontos (indivíduos) que é renovada a cada iteração por uma nova população gerada através da reprodução. Nessas novas populações, os indivíduos mais aptos são mantidos e os menos aptos eliminados.

Os Algoritmos Genéticos⁷ são robustos e apresentam grande probabilidade de alcançar o ótimo global. Por isso, eles são adequados à solução de problemas mistos com variáveis contínuas e discretas e espaços de busca descontínuos.

⁷ Para maior compreensão do método de algoritmo genético, recomendamos o exercício 14.8 do livro “Inteligência Artificial”, do autor Bem Coppin (2004). Este exercício apresenta a otimização de uma função matemática utilizando algoritmo genético.

5 O SIMULADOR DE PROCESSOS IPSEPRO

Esta seção tem como objetivo apresentar ao leitor o programa IPSEPro, um simulador de processos profissional utilizado na modelagem do problema de referência. O capítulo descreverá as suas principais características, bem como o módulo PSE (*Process Simulation Environment*), que é o ambiente de simulação do IPSEpro.

Após isso, este estudo desenvolverá uma introdução ao método de cálculo utilizado pelo PSE, a fim de que o leitor se ambientasse com os métodos matemáticos utilizados pelo simulador. Por fim, a biblioteca de Refrigeração do IPSEpro, denominada Frigo_Lib, será apresentada.

5.1. PRINCIPAIS CARACTERÍSTICAS DO IPSEPRO

O IPSEpro é um simulador de processos profissional para a simulação de processos termodinâmicos por meio de equações de balanço de massa e energia. Trata-se de um simulador bastante flexível e detalhado, cujo desenvolvimento foi realizado pela SimTech, empresa austríaca criada a partir do Institute of Thermal Turbomachinery and Machine Dynamics da Graz University of Technology.

O IPSEpro é composto por uma série de módulos com diferentes funcionalidades, tendo como principal o módulo PSE, o ambiente de simulação propriamente dito e a interface principal com o usuário. Os outros módulos atuam como plug-ins e incrementam as análises realizadas no ambiente de simulação, sendo eles: o PSOptimize, o PSExcel, o PSEconomy, o PSValidate, PSXLink e o MDK. O módulo MDK permite a criação de novos modelos e componentes que podem ser utilizados em conjunto aos preexistentes nas bibliotecas. No presente trabalho, foram utilizados os módulos PSE. A Tabela 1 indica alguns dos módulos de utilização do IPSEpro.

Tabela 1 - Módulos de utilização do IPSE-pro (Simtechnology, 2002)

MÓDULO IPSE-pro	UTILIDADE
PSE – Process Simulation Environment	Ambiente de simulação de processo
MDK – Model Development Kit	Conjunto para desenvolvimento de modelos
PSOptimize	Módulo de otimização
PSExcel	Integração das simulações com as planilhas MS-Excel
PSEconomy	Estimativa de custos durante uma simulação
PSValidate	Módulo para a validação dos dados de uma simulação
PSXLink	<p>Módulo mais recente de integração das simulações do IPSEpro com as planilhas do Excel com as mesmas funcionalidades que o PSExcel.</p> <p>No entanto, o PSXLink é mais adequado para o gerenciamento de grandes quantidades de dados, de modo que muitas limitações existentes no PSExcel não existem no PSXLink.</p>

5.2. O MÓDULO PSE (*Process Simulation Environment*)

O PSE, *Process Simulation Environment*, é o ambiente de simulação do IPSEpro. Nele, é possível realizar as mais diversas modelagens de sistemas térmicos utilizando componentes presentes nas bibliotecas disponíveis. Cada componente possui equações respectivas que são calculadas e convergem através de um modelo iterativo durante a simulação. O PSE fornece uma área de edição dos fluxos de dados e modelagem de plantas térmicas, cuja interface é de fácil acesso e entendimento. A Figura 16 apresenta o ambiente de simulação disponibilizado pelo IPSEpro.

IPSEpro-PSE - [Demo_FL_008.pro(Default)]

File Edit View Project Objects Calculation Tools Window Help

Gerenciador de componentes

Object Manager

Class Object

- A_Mixer A_Mixer001
- A_Connector A_Connector002
- A_Xprescription A_Xprescription0...
- A_Xprescription A_Xprescription0...
- A_Absorber A_Absorber002
- motor motor001
- source source005
- sink sink005
- source source004
- sink sink004
- source source003
- sink sink003
- source source002
- sink sink002
- source source001
- sink sink001
- A_Condenser A_Condenser003
- A_Valve A_Valve002
- A_Evaporator A_Evaporator001
- A_Valve A_Valve001
- A_Pump A_Pump001
- A_RefluxCooler A_RefluxCooler001
- A_Desorber A_Desorber001
- A_Htex A_Htex001
- shaft shaft001
- A_Stream A_Stream015
- A_Stream A_Stream016
- stream stream010
- stream stream009
- A_Stream A_Stream014
- stream stream008

Componente

Paleta de componentes

DEMO_FL_008

Absorption refrigeration process with ammonia-water solution

working fluid	ammonia-water
evaporator	[kW] ! Expression Error !
absorber	[kW] ! Expression Error !
desorber	[kW] ! Expression Error !
reflux cooler	[kW] ! Expression Error !
condenser	[kW] ! Expression Error !
COP	[-] Expression Error !

ammonia-water		pure water	
mass[kg/s]	z[kg/kg]	mass[kg/s]	h[kJ/kg]
p[bar]	t[°C]	p[bar]	t[°C]

Figura 16 – Ambiente de Simulação IPSEpro – módulo PSE (*Process Simulation Environment*) versão 6.0

A modelagem dos processos ocorre quando o usuário seleciona os componentes desejados na biblioteca, insere-os na área de projeto, especifica os dados de entrada de cada um através de uma tabela que se abre ao clicar duas vezes sobre o componente e, em seguida, conecta-os de acordo com a necessidade através de um fluxograma editável. Para a simulação e processo de cálculo do modelo, o PSE utiliza métodos matemáticos numéricos baseados no método de Newton-Raphson.

Basicamente, a simulação no IPSEpro corresponde ao cálculo de dados de saída (incógnitas) a partir dos dados de entrada fornecidos pelo usuário. Uma parte fundamental para o sucesso deste processo é assegurar que o sistema de equações esteja balanceado. Ou seja, é necessário que o número de incógnitas seja igual ao número de equações da simulação. A garantia deste balanceamento é facilitada pela compreensão dos tipos de dados com os quais o PSE trabalha.

O IPSEpro faz uma distinção entre dois tipos principais de dados de entrada: os parâmetros (denominados por *parameters* pelo simulador) e as variáveis definidas (denominadas por *settings* pelo simulador). Matematicamente, os dois seriam classificados como as variáveis independentes do sistema de equações. No entanto, o valor dos parâmetros deve ser obrigatoriamente informado pelo usuário, enquanto que o valor de uma variável pode ser definido pelo usuário (configurando a variável como uma variável definida, ou seja, como um *setting*) ou calculado pelo IPSEpro (configurando a variável como incógnita, ou seja, como um dado de saída). A definição (fixação) do valor de uma variável é feita ao selecionar a opção “*set*” presente na janela do componente. Essa janela surge ao clicar duas vezes sobre o componente. Quando esta opção não é selecionada, a variável é reconhecida como uma incógnita e seu valor é calculado pelo próprio simulador (seu valor passa a ser um dado de saída). Matematicamente, essas variáveis sem o valor definido pelo usuário seriam classificadas como as variáveis dependentes do sistema de equações. Para favorecer a convergência da simulação, o PSE fornece a opção de estimarmos o valor inicial das variáveis dependentes. Para isso, basta clicar na opção “*estimate*” presente na janela do componente e inserir o valor inicial desejado na caixa de texto ao lado.

Caso a quantidade de equações exceda a quantidade de incógnitas, deve-se diminuir a quantidade de variáveis selecionadas como *settings*. Um processo de simulação satisfatório no módulo PSE consiste na solução sem erros (*errors*) ou alertas (*warnings*) do sistema de equações, que ocorre quando o sistema está modelado e balanceado corretamente. A Figura 17

apresenta a janela de um componente com suas respectivas variáveis. Ao lado das variáveis, nota-se a existência das duas opções de configuração: “set” e “estimate”.

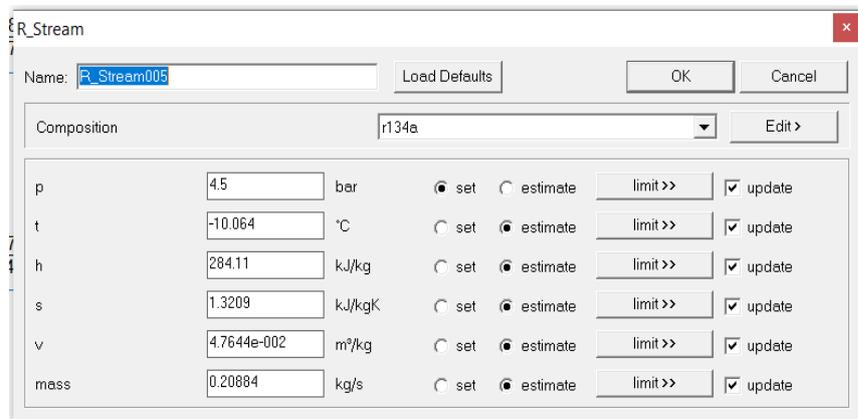


Figura 17 - Variáveis de um componente definidas como “set” e “estimate”

Os resultados, obtidos utilizando o comando “Run Simulation” ou ao apertar a tecla “F5”, são exibidos em formato de fluxograma ou em tabelas que podem ser configuradas manualmente na área de trabalho. Após cada rodada de simulação, o PSE gera automaticamente protocolos de saída detalhando a rotina de cálculo com as seguintes informações: valor e quantidade de incógnitas que convergiram, número de equações do modelo, o tempo computacional de simulação e número de erros (*errors*) e alertas (*warnings*) da simulação.

5.2.1. Métodos De Cálculo Utilizados Pelo Módulo PSE

Todos os métodos de solução de sistemas de equações algébricas não lineares são realizados por um processo iterativo, onde se escolhe um valor arbitrário como valor inicial. A convergência do algoritmo depende fortemente da seleção acertada e viável dos valores iniciais. Percebe-se, com isso, que a seleção desses valores é um ponto chave para a solução bem sucedida dos sistemas de equação.

Segundo Macedo (2017), o método apropriado para a simulação de um sistema de conversão de energia depende essencialmente da natureza das equações que constituem o modelo matemático envolvido. A versão 6.0 do IPSEpro é estruturada em uma abordagem que considera regime permanente (*steady state*) e condições uniformes (*lumped circumstances*) no processo de modelagem. Na hipótese “regime permanente”, não há variação das propriedades com o tempo, ao passo que na hipótese “condições uniformes” um valor médio é atribuído às

propriedades, em outras palavras não há variação das propriedades com o espaço.

O módulo PSE possui um solucionador de equações (*calculation kernel solver*) para o processo de simulação. O módulo PSE evita a solução direta do sistema de equações. O PSE utiliza uma estratégia na qual as variáveis são agrupadas, de modo que os grupos de equações são calculados sequencialmente. As variáveis de cada grupo são escolhidas de modo que o tamanho do grupo é minimizado. Esse direcionamento limita os problemas de convergência em uma única equação não linear ou em um sistema de equações não lineares com um número reduzido de variáveis.

Macedo (2017) exemplifica, ainda, as duas fases que o PSE utiliza para solucionar um problema. A primeira fase consiste na análise do sistema. Nessa fase, o PSE determina o melhor método para a solução do problema. Para esse propósito, o PSE analisa a ordem na qual ocorrerá o tratamento das variáveis do modelo. Para um problema que envolve a solução simultânea de muitas equações, essas últimas são agrupadas pelo PSE. O PSE também escolhe o melhor método para a solução de cada grupo de equações. A segunda fase consiste na solução numérica do problema. O PSE utiliza o método de Newton-Raphson, realizando a linearização das equações a partir do valor inicial escolhido. A linearização resulta em um novo valor inicial a cada iteração. Esse processo é repetido até o limite estabelecido pelo critério de parada.

No PSE, duas variações do método de Newton-Raphson estão disponíveis: método de Newton não amortecido (*undamped Newton method*) e método de Newton amortecido (*damped Newton method*). O PSE utiliza a variação método de Newton amortecido somente no caso da ocorrência de uma falha em uma determinada iteração.

Se a solução do sistema falhar, mesmo que a estrutura do sistema seja válida, a PSE Documentation (2014) apresenta os seguintes motivos:

1. O processo de solução excede o número máximo de etapas de iteração;
2. A precisão necessária não pôde ser alcançada;
3. Valores iniciais ruins não permitem convergência;
4. O processo de solução travou em alguns limites;
5. O modelo de processo é inválido.

5.3. A BIBLIOTECA FRIGO_LIB (*Refrigeration Process Library*)

A biblioteca de Processos de Refrigeração (*Friigo_Lib*) do IPSEPro é a biblioteca que permite ao usuário modelar e simular processos de refrigeração e troca de calor e é a que será utilizada no presente trabalho. A versão da biblioteca utilizada é a 4.0.

A *Friigo_Lib* possibilita a análise de diferentes tecnologias de refrigeração: a refrigeração por absorção e a refrigeração por compressão de vapor. Nela, ainda há a inclusão e disponibilização de modelos de fluidos pré-estabelecidos. No presente trabalho, utilizaremos o modelo de refrigerante R-134a como fluido de trabalho.

A biblioteca *Friigo_Lib* faz distinção dos componentes das tecnologias de refrigeração disponíveis por prefixos. Os componentes com prefixo “R_” (Figura 18) são referentes à refrigeração por compressão de vapor.

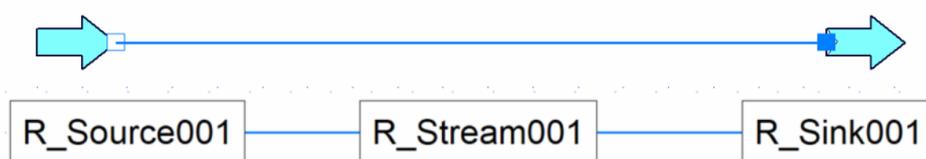


Figura 18 – Componentes do ciclo de refrigeração por compressão de vapor

Já os componentes com o prefixo “A_” (Figura 19) à refrigeração por absorção do sistema NH₃-H₂O (Amônia/Água).

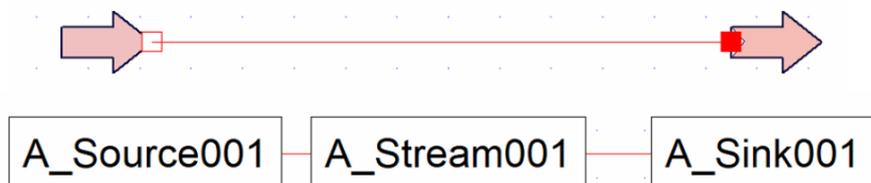


Figura 19 – Componentes do ciclo de refrigeração por absorção do sistema NH₃-H₂O

E os componentes com prefixo “LiBr_” (Figura 20) à refrigeração por absorção do sistema LiBr-H₂O (Brometo de Lítio/Água).



Figura 20– Componentes do ciclo de refrigeração por absorção do sistema LiBr-H₂O

6 MODELAGEM E SIMULAÇÃO DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR EM MÚLTIPLOS ESTÁGIOS

Esta seção apresentará o problema de referência não otimizado, bem como sua modelagem e simulação no IPSEPro. O problema de referência foi retirado de Çengel e Boles (2011) e, de posse dos resultados teóricos, foi possível estabelecer um critério de validação para verificar se os resultados encontrados após a simulação estão dentro ou não de uma faixa aceitável de desvio. Os resultados encontrados foram apresentados e discutidos, como será visto adiante.

6.1. O PROBLEMA DE REFERÊNCIA

Considere um ciclo de refrigeração em cascata de dois estágios com uma câmara de *flash* (separador de líquido), como mostrado na Figura 21 ele utiliza o refrigerante-134a como fluido de trabalho. A temperatura do evaporador é de -10°C , e a pressão do condensador é de 1600 kPa. O refrigerante sai do condensador como líquido saturado e é estrangulado até um separador de líquido que opera a 0,45 MPa. Parte do refrigerante evapora durante esse processo de separação, e o vapor resultante é misturado ao refrigerante que sai do compressor de baixa pressão. A mistura é então comprimida até a pressão do condensador pelo compressor de alta pressão. O líquido do separador de líquido é estrangulado até a pressão do evaporador e resfria o espaço refrigerado à medida que é vaporizado no evaporador. A vazão mássica do refrigerante através do compressor de baixa pressão é de 0,11 kg/s. Considerando que o refrigerante sai do evaporador como vapor saturado e uma eficiência isentrópica de 86% para ambos os compressores, determine (a) a vazão mássica do refrigerante através do compressor de alta pressão, (b) a taxa da refrigeração fornecida pelo sistema e (c) o COP desse refrigerador.⁸

⁸ O problema de referência original apresenta ainda uma quarta questão (“Determine também (d) a taxa da refrigeração e o COP, considerando agora que esse refrigerador é operado em um ciclo por compressão de vapor de estágio único entre a mesma temperatura de evaporação e pressão do condensador com a mesma eficiência do compressor e o mesmo fluxo calculado no item (a).”) que não faz parte do escopo do presente trabalho e, portanto, foi ignorada.

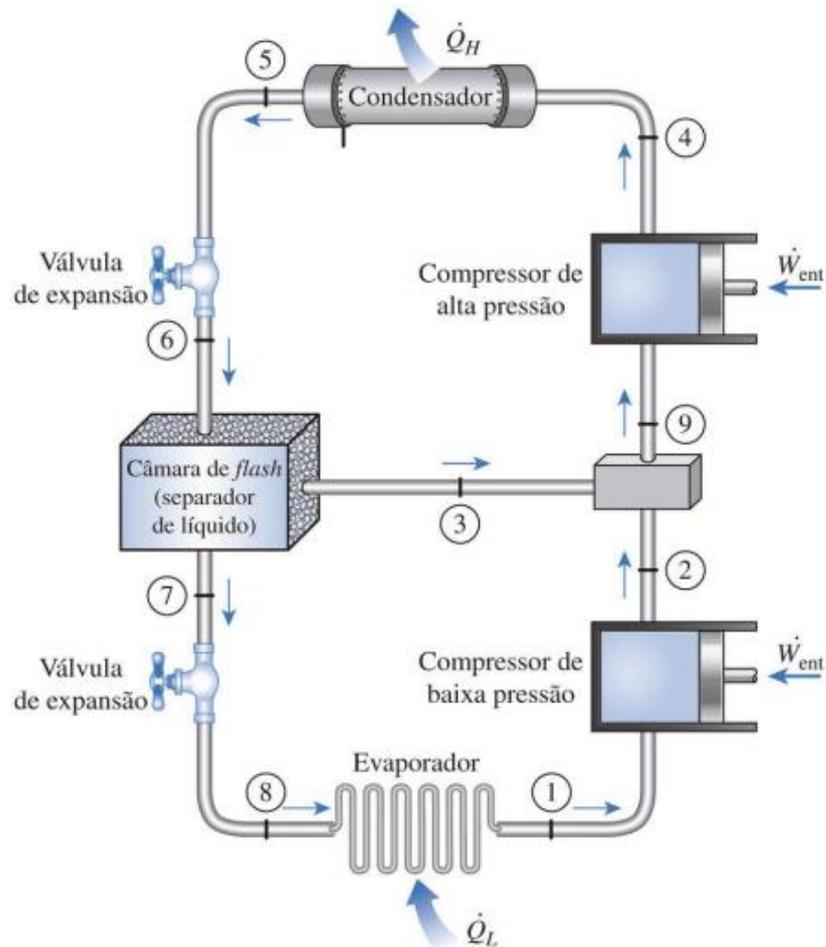


Figura 21 – Ciclo de refrigeração do problema de referência
Fonte: Çengel e Boles (2013)

Os resultados para o problema de referência são apresentados abaixo.

- (a) Vazão mássica do refrigerante através do compressor de alta pressão: 0,1707 kg/s
- (b) A taxa de refrigeração fornecida pelo sistema: 19,33 kW
- (c) O COP desse refrigerador: 2,59

6.2. CRITÉRIO PARA VALIDAÇÃO

A utilização de um problema de referência é uma boa estratégia didática para trabalhos de conclusão de curso de graduação. No entanto, é fundamental que os resultados do problema de referência (valores de referência) estejam disponíveis para que os resultados obtidos (valores

calculados) sejam comparados e validados de acordo com os critérios estabelecidos no trabalho desenvolvido⁹.

De posse dos valores de referência, é possível estabelecer uma faixa aceitável de desvio dos valores calculados, com a utilização do cálculo do erro relativo percentual e comparação com uma tolerância pré-definida¹⁰. A Equação 10 apresenta a formulação para o critério de validação para um ponto utilizado no presente trabalho.

$$\delta (\%) = \left\{ \left| \frac{\text{Valor de referência} - \text{Valor calculado}}{\text{Valor de referência}} \right| \times 100 \right\} < \varepsilon \quad (10)$$

Na Equação 9, δ (%) representa o erro relativo percentual e ε representa a tolerância escolhida. Para o presente trabalho, o valor de referência é um resultado do problema de referência, enquanto o valor calculado é o valor encontrado na simulação com o simulador de processos IPSEpro. Nesse trabalho, a simulação é considerada validada se noventa por cento ou mais dos erros relativos percentuais estiver abaixo da tolerância escolhida, nesse caso 1%¹¹.

6.3. DETALHAMENTO DA MODELAGEM E SIMULAÇÃO DO PROBLEMA DE REFERÊNCIA NO IPSEPRO

Esta subseção abordará os componentes existentes do ciclo de refrigeração por compressão em múltiplos estágios, apresentado no problema de referência e seus modelos. A interligação dos componentes forma o modelo a ser simulado no IPSEpro.

⁹ Os resultados do problema de referência foram disponibilizados pelo orientador para utilização somente no presente trabalho, ou seja, somente para fins didáticos e de orientação de monografia.

¹⁰ O valor escolhido para a tolerância depende essencialmente da natureza do problema. Em linhas gerais, quando se tem em mãos valores de referência provenientes de uma modelagem bem definida (todos os modelos e hipóteses disponíveis), a tolerância escolhida é igual a 1%. No entanto, esse valor poderá ser maior quando a modelagem não estiver bem definida. Em determinados problemas, uma tolerância bem maior poderá ser utilizada e estará associada a valores satisfatórios.

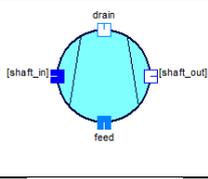
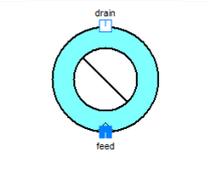
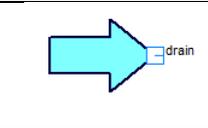
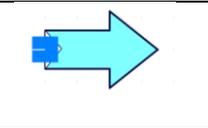
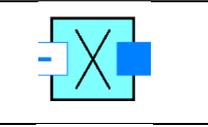
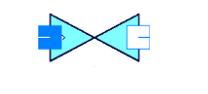
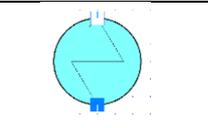
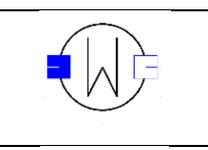
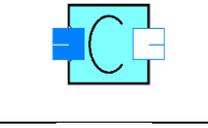
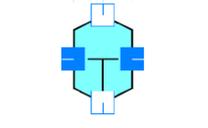
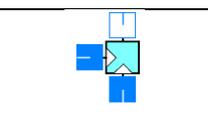
¹¹ É importante observar que o problema de referência é um problema não otimizado. Assim sendo, outro critério de validação será utilizado no processo de otimização do problema de referência. No entanto, a validação da simulação é fundamental para o ponto operacional do problema de referência e para o desenvolvimento do modelo utilizado no processo de otimização.

O processo de resolução das equações envolvidas na modelagem desenvolvida é denominado simulação. Conforme mencionado na seção anterior, o IPSEpro realiza essa resolução internamente utilizando o método de Newton-Raphson.

6.3.1. Objetos Utilizados Na Modelagem E Simulação Do Problema

Inicialmente, faz-se necessária uma breve apresentação dos objetos na modelagem e simulação executada por este trabalho, como mostra a Tabela 2. O Anexo A apresenta os modelos utilizados nos objetos da simulação, desenvolvidos pelo IPSEpro.

Tabela 2 - Objetos do IPSE-pro (SIMTECHNOLOGY, 2002) utilizados na modelagem e simulação

	Compressor
	Condensador
	Fonte
	Sumidouro
	R_Xprescripton
	Dispositivo de expansão
	Evaporador
	Motor
	Conector
	Separador de líquido
	<i>Mixer</i>

6.3.2. Fluido Refrigerante

Para modelar o problema no IPSEpro, o primeiro passo consistiu em especificar o fluido de trabalho R-134a no programa. Para tanto, seguiram-se as etapas: clicar em “*objects*” na barra de opções localizada no topo do programa; marcar a opção “*new global project*”; encontrar o objeto que especifica a composição (*R_Composition*); nomeá-lo - este trabalho optou por fazer a nomeação mais intuitiva possível, chamando a corrente de “R134a”; e finalmente clicar em “ok”.

O próprio IPSEpro, em sua base de dados interna, já possui modelos de muitos fluidos refrigerantes, inclusive o R-134a.¹² Nas próximas subseções serão inseridas as correntes e, ao clicar duas vezes em cada uma delas, é possível associar o fluido R134a à corrente em questão.

A Figura 22 ilustra o processo de criação desse objeto e a Figura 23 a associação do refrigerante a uma corrente.

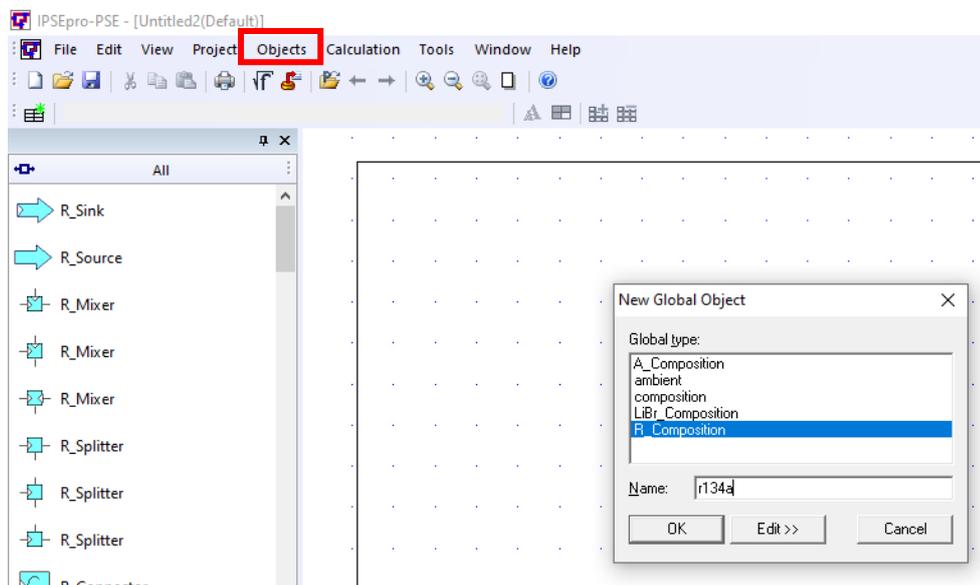


Figura 22 - Criação de um objeto para a especificação do fluido refrigerante

¹² Para informações a respeito das equações de estado e funções matemáticas para o cálculo das propriedades termodinâmicas dos refrigerantes, consultar o manual da biblioteca *Refrigeration Process Library, Library Version 4.0 Manual Version 4.0.000, Copyright: SimTech, 2000-2003*

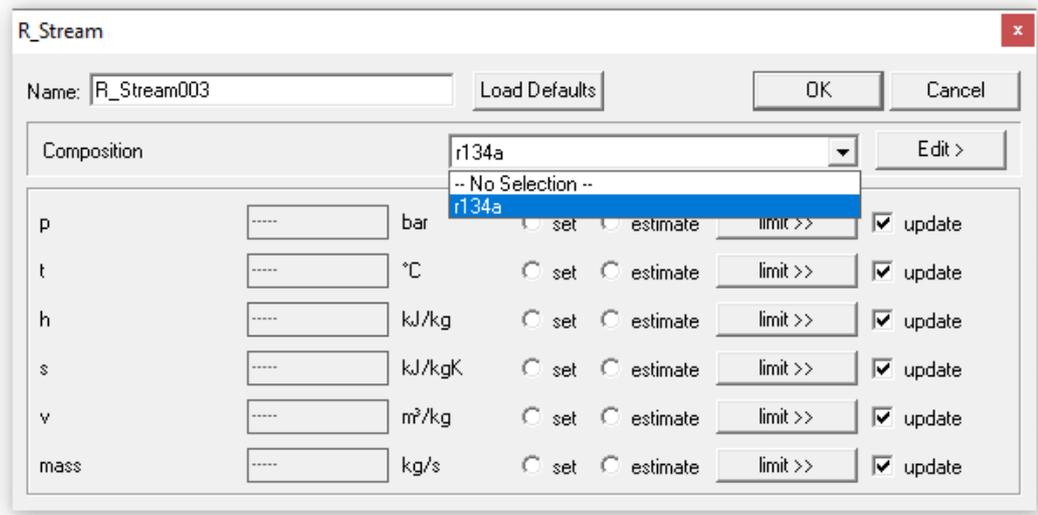


Figura 23– Associação da corrente a um fluido refrigerante

6.3.3. Condensador

Conforme previamente mencionado, o problema de referência forneceu pressão e título do refrigerante na saída do condensador, portanto, foi tomado este componente como primeiro modelo a ser desenvolvido no IPSEpro.

Para isso, o condensador ($R_Heat_sink001$) foi adicionado à área de trabalho do IPSEpro, além de uma fonte (R_Source) e um sumidouro (R_Sink) que serviram apenas como ferramentas provisórias para a verificação da correta modelagem do condensador¹³, como mostra a Figura 24. Para incluir um componente na área de trabalho, o mesmo deve ser escolhido na paleta dos componentes a esquerda, clicando no objeto desejado e arrastando-o até a área de trabalho.



Figura 24– Princípio da modelagem do problema pelo condensador

¹³ É recomendável que a modelagem no IPSEpro seja feita de forma gradativa, ou seja, sem inserir todos os componentes simultaneamente na área de trabalho. Essa estratégia evita um número excessivo de erros e/ou avisos na simulação, e pode acelerar o processo de modelagem do sistema. A simulação satisfatória é aquela sem erros e sem avisos (*warnings*).

De modo a indicar a fração mássica de vapor saturado do fluido que sai do condensador, o simulador de processos IPSEpro possui um objeto denominado *R_Xprescription*, onde é possível definir o título, como mostra a Figura 25. Todos os componentes foram interligados pelas correntes do sistema (*streams*).

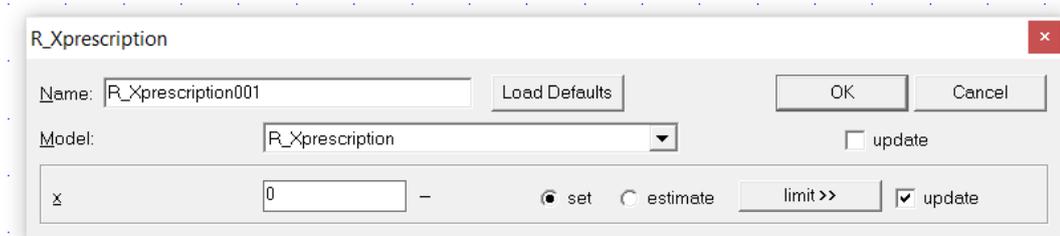


Figura 25– Configuração do título na saída do compressor

O simulador de processos IPSEpro permite que os valores das variáveis sejam definidos e fixados ao longo de toda simulação, configurando essas variáveis como um *setting* ao selecionar a opção “*set*” presente na janela do componente. Além disso, é possível deixar que os valores das variáveis sejam calculados como dados de saída da simulação. Para favorecer a convergência da simulação é possível definir uma estimativa inicial para o dado de saída ao selecionar a opção “*estimate*” presente na janela do componente. Para determinar a pressão de trabalho do condensador, a pressão especificada pelo problema de referência (transformada para bar, conforme configuração padrão do IPSEpro) foi configurada como *set* na corrente de entrada do trocador de calor, como mostra a Figura 26.

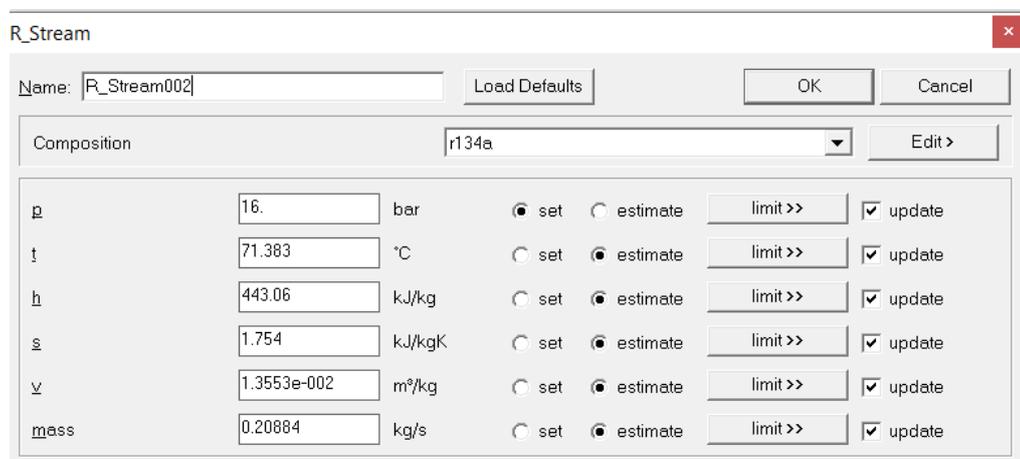


Figura 26– Configuração da pressão na entrada do condensador

A pressão de trabalho não foi o único dado de entrada declarado neste primeiro momento e, para indicar que o processo é isobárico, estabeleceu-se variação de pressão nula (“ Δp ”), como mostra a Figura 27. Os outros dados de cada objeto mencionado foram calculados pelo IPSEpro a partir de seus valores iniciais fornecidos pelo usuário, ao selecionar a opção “*estimate*” do programa.

Figura 27- Configuração da transferência de calor no condensador

6.3.4. Válvula de Expansão na Saída do Condensador

Em seguida, foi incluído o dispositivo de expansão na área de trabalho (*R_Valve*) e conectado ao objeto *R_Xprescription*¹⁴ que indica o título do fluido. Neste objeto nenhum *set* foi definido, ou seja, seus resultados serão obtidos por meio da convergência do processo como um todo. Para tanto, ao clicar duas vezes no dispositivo de expansão na área de trabalho, selecionou-se a opção *estimate* para queda de pressão e delta de pressão (diferença entre pressão a jusante e pressão a montante do objeto). A Figura 28 apresenta as configurações deste dispositivo.

¹⁴ Este objeto não um componente do sistema, trata-se de um recurso para mostrar o título no ponto onde ele é inserido.

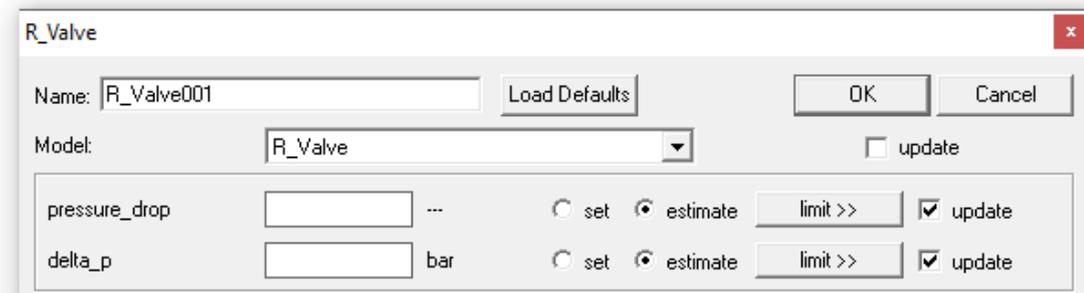


Figura 28 – Janela da válvula de expansão onde os dados de saída serão calculados pelo IPSEpro

6.3.5. Separador de Líquido

Conectado ao dispositivo de expansão está o separador de líquido e, conforme descrito na Seção 3 deste trabalho, tal componente opera separando e direcionando o refrigerante líquido-vapor. Desta forma, a fase líquida deste fluido será direcionada para uma corrente e a fase vapor deste fluido irá ser direcionada para outra.

O separador de líquido no IPSEpro é denominado *R_Separator* e o problema de referência informou que ele opera a 450 kPa. Esta pressão foi definida como set na corrente de entrada do componente, como mostra a Figura 29.

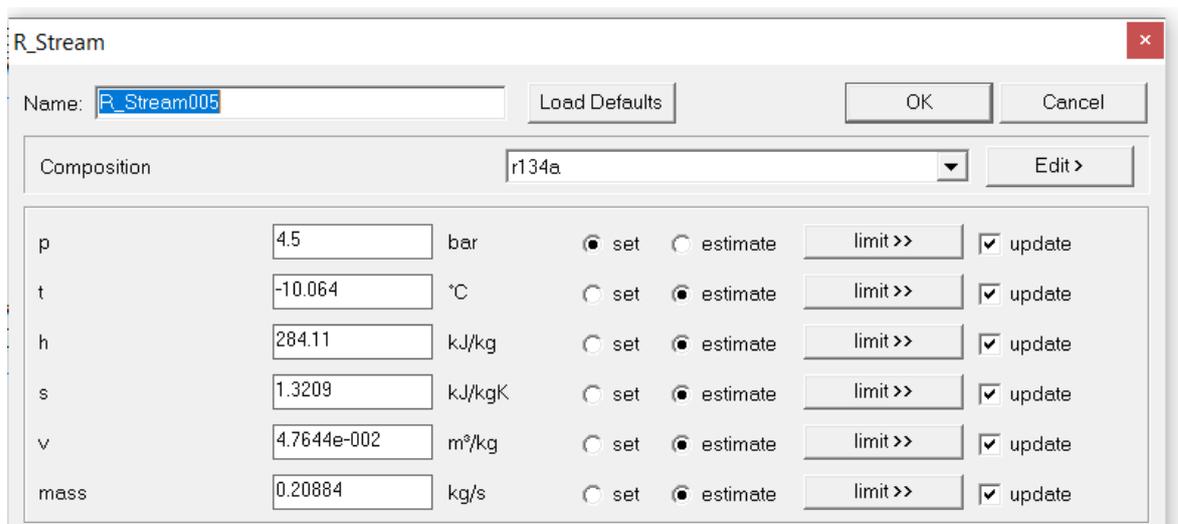


Figura 29– Configuração da pressão na entrada do separador de líquido

O objeto apresenta ainda uma variável ($y_{separation}$) que indica a razão entre a vazão mássica de líquido que sai do separador de líquido e é direcionado para o dispositivo de expansão e a vazão mássica de líquido saturado que entra no separador de líquido. O trabalho considera $y_{separation}$ igual a 1. A Figura 30 mostra como o separador de líquido foi configurado.

R_Separator

Name: Load Defaults OK Cancel

Model: update

$y_{separation}$ — set estimate limit >> update

Figura 30- Configuração do separador de líquido no sistema

6.3.6. Válvula De Expansão Na Saída Do Separador

O líquido do refrigerante que sai do separador de líquido é direcionado para uma segunda válvula de expansão. O problema de referência indica que a vazão mássica no compressor de baixa pressão é de 0,11 kg/s e, como não há perda nem adição de refrigerante entre esse segundo dispositivo de expansão e o compressor de baixa pressão, a vazão mássica da corrente entre o separador de líquido e a válvula de expansão foi configurada com o valor de 0,11 kg/s, como mostra a Figura 31.

R_Stream

Name: Load Defaults OK Cancel

Composition: Edit >

p	<input type="text" value="2.001"/>	bar	<input type="radio"/> set	<input checked="" type="radio"/> estimate	limit >>	<input checked="" type="checkbox"/> update
t	<input type="text" value="-10.064"/>	°C	<input type="radio"/> set	<input checked="" type="radio"/> estimate	limit >>	<input checked="" type="checkbox"/> update
h	<input type="text" value="186.61"/>	kJ/kg	<input type="radio"/> set	<input checked="" type="radio"/> estimate	limit >>	<input checked="" type="checkbox"/> update
s	<input type="text" value="0.95033"/>	kJ/kgK	<input type="radio"/> set	<input checked="" type="radio"/> estimate	limit >>	<input checked="" type="checkbox"/> update
v	<input type="text" value="7.9852e-004"/>	m³/kg	<input type="radio"/> set	<input checked="" type="radio"/> estimate	limit >>	<input checked="" type="checkbox"/> update
mass	<input type="text" value="0.11"/>	kg/s	<input checked="" type="radio"/> set	<input type="radio"/> estimate	limit >>	<input checked="" type="checkbox"/> update

Figura 31– Configuração da corrente de entrada da segunda válvula de expansão

6.3.7. Evaporador

O evaporador, que opera a $-10\text{ }^{\circ}\text{C}$ segundo o problema de referência, é então conectado à válvula de expansão descrita no tópico 6.3.6. Na corrente de saída do evaporador, o refrigerante encontra-se na condição de vapor saturado.

Em função de ser uma variável definida pelo problema de referência, a temperatura operação do refrigerante no evaporador foi configurada com o valor de $-10\text{ }^{\circ}\text{C}$. No entanto, para que a pressão possa ser utilizada como dado de entrada principal na etapa de otimização do sistema, utilizou-se o recurso denominado inversão de simulação (a pressão encontrada foi configurada como valor fixado e a temperatura encontrada foi definida como um valor a ser calculado pela simulação). Desta forma, o número de variáveis declaradas e fixadas (pressão e título) permaneceu o mesmo, tornando possível o cálculo do valor da temperatura do evaporador pelo IPSEpro.

Assim como no condensador, foi necessária a utilização do componente *R_Xprescription* para informar o título na saída do evaporador, como mostra a Figura 32. A variação de pressão no componente foi definida como zero, como ilustra a Figura 33. A Figura 34 apresenta a configuração de corrente antes do trocador de calor em questão.

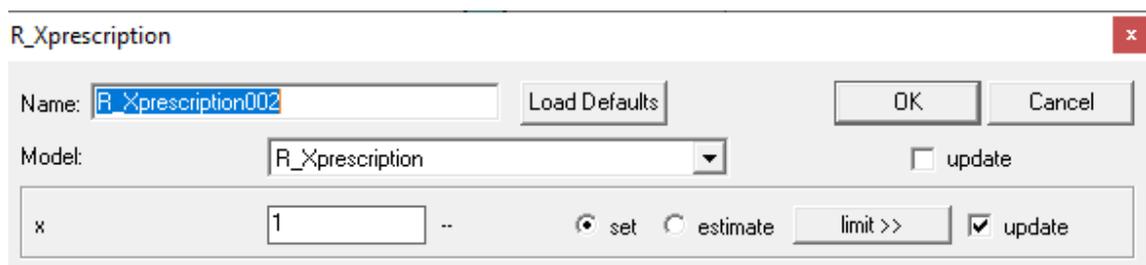


Figura 32– Configuração do título do refrigerante na saída do evaporador

R_Heat_source

Name: Load Defaults OK Cancel

Model: update

delta_p	<input type="text" value="0"/>	bar	<input checked="" type="checkbox"/>	update
delta_t	<input type="text" value="-4.3477e-016"/>	°C	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update
q_trans	<input type="text" value="19.324"/>	kW	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update

Figura 33– Configuração dos dados de entrada do evaporador

R_Stream

Name: Load Defaults OK Cancel

Composition: Edit >

p	<input type="text" value="2.0058"/>	bar	<input checked="" type="radio"/> set <input type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update
t	<input type="text" value="-10.003"/>	°C	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update
h	<input type="text" value="216.99"/>	kJ/kg	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update
s	<input type="text" value="1.0658"/>	kJ/kgK	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update
v	<input type="text" value="1.5289e-002"/>	m³/kg	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update
mass	<input type="text" value="0.11"/>	kg/s	<input type="radio"/> set <input checked="" type="radio"/> estimate	<input type="button" value="limit >>"/> <input checked="" type="checkbox"/> update

Figura 34– Configuração da corrente na entrada do evaporador

6.3.8. Compressor de Baixa Pressão

O compressor de baixa pressão é conectado então ao evaporador e também a um motor que fornecerá energia suficiente para promover o funcionamento do sistema. O problema de referência afirma que os compressores do sistema têm uma eficiência isentrópica de 86%¹⁵, sendo este valor declarado e fixado no IPSEpro, como mostra a Figura 35. Os demais dados de entrada do componente (vazão volumétrica e taxa de compressão) foram estimados pelo programa.

¹⁵ O valor da eficiência isentrópica depende substancialmente do projeto do compressor.

R_Compressor

Name: Load Defaults OK Cancel

Model: update

eta_s	<input type="text" value="0.86"/>	—			<input checked="" type="checkbox"/> update
pressure_ratio	<input type="text" value="1.0005"/>	—	<input type="radio"/> set	<input checked="" type="radio"/> estimate	<input type="text" value="limit >>"/> <input checked="" type="checkbox"/> update
V	<input type="text" value="1.0991e-002"/>	m³/s	<input type="radio"/> set	<input checked="" type="radio"/> estimate	<input type="text" value="limit >>"/> <input checked="" type="checkbox"/> update

Figura 35– Configuração do compressor de baixa pressão

6.3.9. Mixer

O objeto *Mixer* (R_Mixer) direciona ao compressor de alta pressão o vapor proveniente do compressor de baixa pressão e do separador de líquido, logo a sua vazão mássica será a soma da vazão mássica dos dois componentes mencionados. Nele não há nenhuma variação de pressão, logo, sua configuração fica como indicado na Figura 36.

R_Mixer

Name: Load Defaults OK Cancel

Model: update

delta_p_1	<input type="text" value="0"/>	bar	<input checked="" type="radio"/> set	<input type="radio"/> estimate	<input type="text" value="limit >>"/> <input checked="" type="checkbox"/> update
delta_p_2	<input type="text" value="0"/>	bar	<input checked="" type="radio"/> set	<input type="radio"/> estimate	<input type="text" value="limit >>"/> <input checked="" type="checkbox"/> update

Figura 36– Configuração do Mixer

6.3.10. Compressor de Alta Pressão

Assim como o compressor de baixa pressão, o compressor de alta pressão possui uma eficiência isentrópica de 86%, de acordo com os dados do problema de referência. Nenhum outro dado de entrada, além da eficiência isentrópica, foi definido pelo problema de referência, ficando a modelagem do compressor de alta pressão indicada na Figura 37.

R_Compressor

Name: Load Defaults OK Cancel

Model: update

eta_s	<input type="text" value="0.86"/>	—		<input checked="" type="checkbox"/> update
pressure_ratio	<input type="text" value="7.996"/>	—	<input type="radio"/> set <input checked="" type="radio"/> estimate <input type="button" value="limit >>"/>	<input checked="" type="checkbox"/> update
V	<input type="text" value="2.0848e-002"/>	m³/s	<input type="radio"/> set <input checked="" type="radio"/> estimate <input type="button" value="limit >>"/>	<input checked="" type="checkbox"/> update

Figura 37– Configuração do compressor de alta pressão

6.3.11. Motores

O simulador IPSEpro possibilita a inclusão de motores em suas simulações. As Figuras 38 e 39 ilustram as janelas de configuração dos motores elétricos dos compressores de baixa e alta pressão, respectivamente, que indicam que a potência necessária foi estimada. Os demais parâmetros (eficiência elétrica e eficiência mecânica) são os valores padrão do programa.

motor

Name: Load Defaults OK Cancel

Model: update

power	<input type="text" value="8.6008e-004"/>	kW	<input type="radio"/> set <input checked="" type="radio"/> estimate <input type="button" value="limit >>"/>	<input checked="" type="checkbox"/> update
eta_el	<input type="text" value="1."/>	—		<input checked="" type="checkbox"/> update
eta_m	<input type="text" value="1."/>	—		<input checked="" type="checkbox"/> update

Figura 38– Configuração do motor elétrico do compressor de baixa pressão

motor

Name: Load Defaults OK Cancel

Model: update

power	<input type="text" value="10.533"/>	kW	<input type="radio"/> set <input checked="" type="radio"/> estimate <input type="button" value="limit >>"/>	<input checked="" type="checkbox"/> update
eta_el	<input type="text" value="1."/>	—		<input checked="" type="checkbox"/> update
eta_m	<input type="text" value="1."/>	—		<input checked="" type="checkbox"/> update

Figura 39– Configuração do motor elétrico do compressor de alta pressão

6.3.12. Conector

O conector é um objeto do simulador de processos IPSEpro que é inserido em um ciclo termodinâmico fechado para eliminação de redundâncias no balanço de massa. Não se trata, portanto, de um equipamento do ciclo, mas sim de um recurso do IPSEpro. Não há configurações adicionais neste objeto, como ilustra a Figura 40.

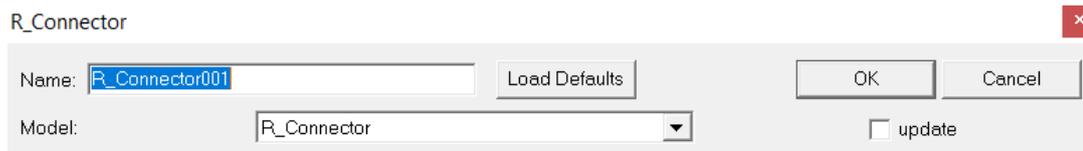


Figura 40– Configuração do conector

6.4. RESULTADOS E DISCUSSÃO DA SIMULAÇÃO

Após interligados todos os componentes, é preciso fazer o sistema rodar, o que é realizado com o seguinte caminho: clique na opção “*calculation*” na parte superior da tela e em seguida “*run*”.

A Figura 41 mostra como fica a modelagem completa do sistema, bem como os resultados encontrados após a simulação. No total foram modelados 27 objetos, sendo 1 separador de líquido, 1 mixer, 2 *R_Xprescriptions*, 1 condensador, 1 evaporador, 2 compressores, 2 válvulas de expansão, 2 motores, 12 correntes, eixos e 1 conector.

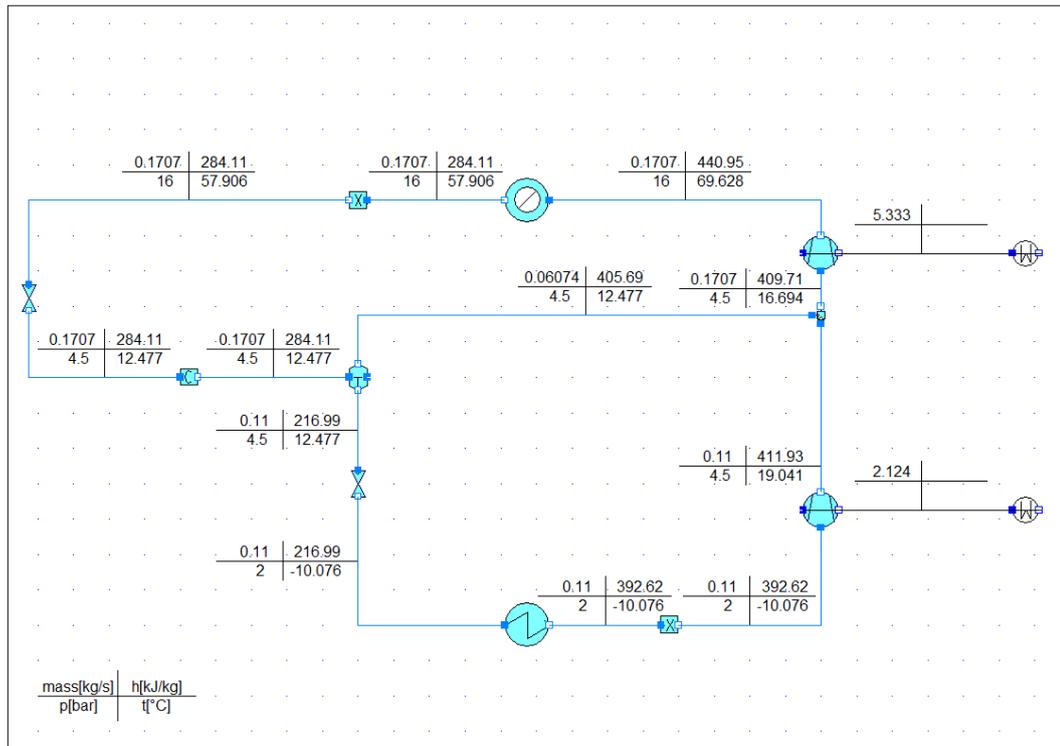


Figura 41– Modelagem do sistema e resultados após a simulação

Conforme é possível verificar na Figura 41, o programa IPSEpro exibe os resultados – vazão mássica, entalpia específica, pressão e temperatura - de forma muito fácil de identificar.

Se por um lado, o IPSEpro retorna ao usuário informações de forma bem clara e direta, por outro, os cálculos envolvidos e as iterações realizadas pelo sistema são bastante complexas. A simulação realizada no presente trabalho compreendeu a resolução de um conjunto de 85 equações, como mostra a Figura 42. A Figura 43 mostra parte do relatório gerado pelo IPSEpro que indica que a simulação realizada não contém erros ou avisos (*warnings*). O relatório completo extraído do IPSEpro da simulação do problema de referência pode ser encontrado no Apêndice A.

```

=====
Loading system ...

System has been loaded successfully.
System analysis ...
=====
System model contains 94 variables
9 variables set to fixed values
System to be solved contains 85 variables and 85 equations
85 groups found. Largest group contains 1 variables.
=====
System analysis finished.

```

Figura 42– Parte do relatório gerado pelo IPSEpro que apresenta a dimensão do modelo (85 variáveis e 85 equações)

```

=====
Solution
Start:      Oct 26 2020, 20:52:36,695
End:       Oct 26 2020, 20:52:36.894
Actual Time: 0.199000 seconds
CPU Time:   0.203000 seconds
CPU Percentage: 12%
0 errors (0 recovered errors)
0 warnings (0 recovered warnings)
=====

```

Figura 43– Relatório gerado pelo IPSEpro que indica a quantidade de os erros e avisos na simulação

6.5. COMPARAÇÃO COM O PONTO DE PROJETO

De posse dos valores de referência e da metodologia para cálculo do erro percentual, foi possível identificar que os resultados globais encontrados estão abaixo do erro previsto de 1%, sendo, portanto, validados pelo critério de validação aqui estabelecido. A exceção é encontrada na faixa dos valores de entalpia, que será justificada na Seção 6.6.

A Tabela 3 mostra os valores de vazão mássica, pressão, temperatura e entalpia específica dos pontos apresentados no trabalho de referência (PR) e obtidos com a simulação no IPSEpro (SM), bem como os desvios percentuais do SM em relação ao PR.

Tabela 3 - Comparação entre os resultados de vazão mássica, pressão, temperatura e entalpia específica do IPSEpro e do trabalho de referência.

Ponto	Vazão mássica (Kg/s)			Pressão (Mpa)			Temperatura (°C)			Entalpia específica (KJ/Kg)		
	PR	SM	δ (%)	PR	SM	δ (%)	PR	SM	δ (%)	PR	SM	δ (%)
1	0,11	0,11	0	0,2	0,2	0	-10	-10,076	0,76	244,51	392,62	60,57
2	0,11	0,11	0	0,45	0,45	0	19,025	19,041	0,08	263,76	411,93	56,18
3	0,0607	0,0607	0	0,45	0,45	0	12,46	12,477	0,14	257,53	405,69	57,53
4	0,1707	0,1707	0	1,6	1,6	0	70	69,628	0,53	292,78	440,95	50,61
5	0,1707	0,1707	0	1,6	1,6	0	57,88	57,906	0,04	135,93	284,11	109,01
6	0,1707	0,1707	0	0,45	0,45	0	12,46	12,477	0,14	135,93	284,11	109,01
7	0,11	0,11	0	0,45	0,45	0	12,46	12,477	0,14	68,81	216,99	215,35
8	0,11	0,11	0	0,2	0,2	0	-10	-10,076	0,76	68,81	216,99	215,35
9	0,1707	0,1707	0	0,45	0,45	0	16,683	16,694	0,07	261,54	409,71	56,65

Legenda: PR = Problema de referência; SM = simulação no IPSEpro; δ (%) = erro relativo percentual

6.5.1. Considerações Sobre os Erros Relativos

Percebemos grandes erros relativos em todas as comparações entre as entalpias específicas do trabalho de referência com a simulação realizada no IPSEPro. Ao analisar a origem desses resultados, percebemos um padrão que se mostrou constante em todos os pontos: a diferença absoluta entre a entalpia específica da simulação e a do problema de referência. A Tabela 4 mostra a diferença encontrada entre os pontos.

Tabela 4 - Diferença entre os valores relativos da entalpia específica ponto a ponto

Ponto	Entalpia específica (KJ/Kg)		
	PR	SM	Diferença
1	244,51	392,62	148,11
2	263,76	411,95	148,19
3	257,53	405,75	148,22
4	292,78	441	148,22
5	135,93	284,11	148,18
6	135,93	284,11	148,18
7	68,81	216,99	148,18
8	68,81	216,99	148,18
9	261,54	409,75	148,21

Sabe-se que entalpia específica é uma propriedade termodinâmica e não se caracteriza como um valor absoluto, mas sim um valor relativo a partir um ponto de referência. Desta forma, é possível concluir que o ponto de referência utilizado no trabalho de referência difere do utilizado pela simulação em um valor aproximado de 148,18 kJ/kg.

Assim, pode-se desconsiderar os valores das entalpias e utilizar as demais validações ponto a ponto. Neste cenário, 100% dos erros relativos ficaram dentro da faixa determinada de 1%, o que indica que a simulação foi validada.

6.6. VALIDAÇÃO DOS RESULTADOS DA SIMULAÇÃO

De posse dos resultados teóricos e da metodologia para cálculo do erro percentual, foi possível identificar que os resultados globais encontrados estão inferiores a 1%, sendo, portanto, aceitos no critério de validação aqui estabelecido. A Tabela 5 mostra os valores principais interações energéticas (Calor e Potência) do problema de referência e da simulação, e os respectivos erros percentuais associados.

Na Tabela 5 “W_comp_baixa” representa a potência do compressor de baixa pressão do ciclo; “W_comp_alta” representa a potência do compressor de alta pressão do ciclo; “Q_evap” é a taxa de calor transferida no evaporador; e “Q_cond” é a taxa de calor transferida no condensador.

Tabela 5 - Erros percentuais das principais interações energéticas

Interações	PR	SM	δ (%)
W_comp_baixa (kW)	2,1175	2,12396	0,30508
W_comp_alta (kW)	5,33267	5,3308	0,035
Q_evap (kW)	19,33	19,3195	0,05432
Q_cond (kW)	26,7743	26,7766	0,00861

Legenda: PR = Problema de referência; SM = simulação no IPSEpro; δ (%) = erro percentual

7 ANÁLISE ENERGÉTICA DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS

Esta seção apresenta as ferramentas necessárias para a realização da análise energética, com o cálculo do coeficiente de performance (COP). Ela descreve a integração entre o MATLAB e o IPSEPro, ilustra o desenvolvimento de códigos que extraem os dados termodinâmicos desejados do simulador de processos e que são processados pelo MATLAB, possibilitando assim a elaboração da análise do desempenho do sistema em termos da primeira lei da termodinâmica.

7.1. INTEGRAÇÃO ENTRE O SIMULADOR DE PROCESSOS IPSEPRO E O MATLAB

Com o objetivo de avaliar como foram determinados os dados de interesse do ciclo analisado exercem influência sobre o coeficiente de performance – COP, a integração entre o Matlab e o IPSEPro mostra-se uma solução confiável e relativamente rápida para a obtenção de dos resultados desejados.

Uma grande vantagem de utilizar o simulador de processos IPSEpro é encontrada pelo fato deste simulador conter códigos e modelos consagrados para a realização dos balanços de massa e energia, proporcionando resultados confiáveis e precisos e excluindo a necessidade de programar os princípios de conservação mencionados.

Além disso, a integração entre dois programas consagrados é uma boa estratégia pois dessa forma utilizamos todo o potencial de cada um deles para realizar uma tarefa específica, ou seja, o IPSEpro realiza os balanços de massa e energia e o MatLab realiza os demais cálculos, incluindo a geração de gráficos e o processo de otimização. Portanto, extraímos o melhor de cada programa e combinamos os mesmos para realizar uma tarefa maior.

7.1.1. Principais Comandos Utilizados na Interface COM

De modo a estabelecer a comunicação entre o MATLAB e o IPSEpro, foi utilizada a interface COM – *Component Object Model* desenvolvida pela Microsoft® (Microsoft, 2020). A princípio, para a integração entre dois programas quaisquer utilizando a interface COM, ambos devem ter essa opção de comunicação em seus respectivos códigos. A Tabela 6 apresenta

os principais comandos utilizados para a interface entre o IPSEpro e o MatLab utilizando a interface COM.

Tabela 6 – Principais comandos utilizados na interface COM

Fonte: Macedo, 2017

Sintaxe	Descrição
<code>app = actxserver('PSE.Application')</code>	O comando <code>actxserver</code> cria um servidor automático COM, em outras palavras cria a interface COM entre o IPSEpro (módulo PSE) e o MatLab.
<code>proj = invoke(proj,'openProject',ProjPath)</code>	Essa sintaxe invoca para a interface COM criada o método <code>openProject</code> (classe <code>Application</code>), no diretório <code>ProjPath</code> . Em outras palavras, abre o arquivo <code>.pro</code> criado no PSE.
<code>run = invoke(proj,'runSimulation',datasetn)</code>	Essa sintaxe invoca o método <code>runSimulation</code> (classe <code>Project</code>), para um determinado <i>dataset</i> . Em outras palavras executa a simulação do projeto aberto em <code>proj</code> .
<code>fluxo01 = invoke(proj,'findObject',1)</code>	Essa sintaxe invoca o método <code>findObject</code> (classe <code>Project</code>), onde o objeto em questão é uma <i>stream</i> e 1 é o seu nome no PSE. A <i>stream</i> é parte do projeto aberto em <code>proj</code> . O objeto também pode ser uma <i>unit</i> ou uma <i>global</i> .
<code>temp = invoke(flujo01,'findItem',datasetn,'mass')</code>	Essa sintaxe invoca o método <code>findItem</code> (classe <code>Object</code>), para um determinado <i>dataset</i> . O item em questão é a vazão mássica (<i>mass</i>). Em outras palavras, essa sintaxe seleciona o item referenciado dentro do objeto.
<code>m01 = invoke(temp,'resultValue')</code>	Essa sintaxe invoca a propriedade <code>resultValue</code> (classe <code>Item</code>). Em outras palavras, retorna o valor numérico da vazão mássica do fluxo 01 e o atribui à variável <code>m01</code> do MatLab.

7.2. DIAGRAMA ESTRUTURAL DO PROGRAMA COMPUTACIONAL DESENVOLVIDO NO MATLAB

Com o objetivo de descrever graficamente como as diferentes partes do programa estão conectadas e auxiliar os estágios iniciais do desenvolvimento do programa¹⁶, foi criado o diagrama estrutural do programa computacional desenvolvido no MatLab. Um diagrama estrutural exibe a organização de um programa sem mostrar os detalhes dos cálculos e dos processos de decisão (Palm, 2013). Em linhas gerais, um programa é composto de um *script* principal e módulos que executam tarefas específicas. Um diagrama estrutural mostra a conexão entre o *script* principal e os módulos. A Figura 44 apresenta o diagrama estrutural para o programa desenvolvido neste trabalho, onde as setas representam a troca de dados entre o MatLab e o IPSEpro.

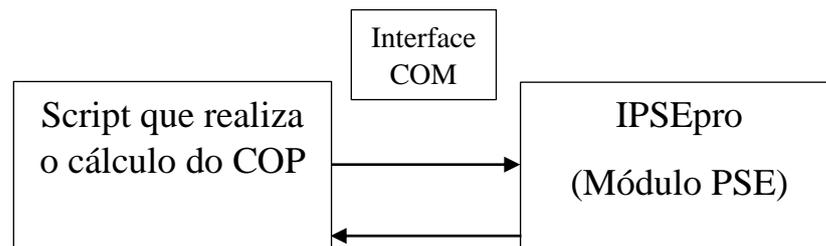


Figura 44– Diagrama estrutural do programa computacional desenvolvido no MATLAB

7.3. FLUXOGRAMA DO CÓDIGO COMPUTACIONAL DESENVOLVIDO

A Figura 45 apresenta o fluxograma do código computacional desenvolvido que realiza o cálculo do COP. Esta função é utilizada internamente pelo algoritmo genético como função objetivo a ser maximizada (ou minimizada, ao utilizar o inverso do COP).

Assim como o diagrama estrutural, a elaboração de um fluxograma computacional é de grande importância, uma vez que serve para o desenvolvimento, a visualização e a documentação de um programa.

¹⁶ Embora o projeto do programa não seja uma etapa obrigatória para o desenvolvimento de uma solução computacional, o programa do presente trabalho foi projetado de acordo com o conceito de programação estruturada, com a utilização do método de projeto *top-down*. Para mais detalhes a respeito de programação estruturada e projeto *top-down*, consultar a referência bibliográfica (Palm, 2013).

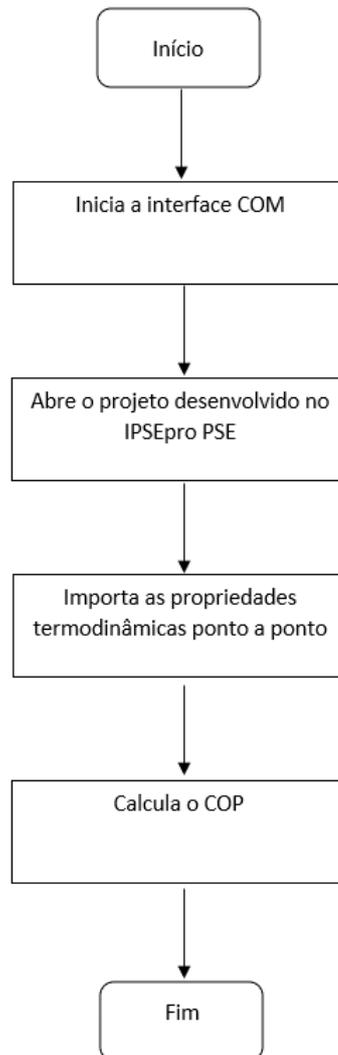


Figura 45– Fluxograma da função que realiza o cálculo do COP

7.4. CRITÉRIO PARA VALIDAÇÃO

Assim como a validação da simulação do sistema foi baseada na comparação entre o resultado de referência e o resultado calculado na simulação, o critério de validação para a análise energética seguirá a mesma métrica.¹⁷ Os dados de saída validados são as interações energéticas (taxa de calor e potência) e o COP do sistema.

¹⁷ A exposição dos valores das taxas de transferência de energia calculados pelo IPSEpro e a comparação destes valores com os valores apresentados pelo problema de referência foram abordadas previamente na sexta seção, como parte da validação ponto a ponto da modelagem e simulação termodinâmicas feitas no IPSEpro. Na seção anterior, os valores das entalpias específicas ponto a ponto apresentaram erros relativos acima de 1%, porque o cálculo da entalpia específica do IPSE foi realizado a partir de um estado de referência diferente do estado de referência utilizado pelo problema de referência. Para mostrar que a presença desses erros relativos acima da

A Equação 10, apresentada na Seção 6, indica a formulação para o critério de validação utilizado no presente trabalho. Nesse trabalho, a simulação é considerada validada se noventa por cento ou mais dos erros relativos percentuais estiver abaixo da tolerância escolhida, nesse caso 1%.

7.5. RESULTADOS E DISCUSSÃO DA ANÁLISE ENERGÉTICA

Os resultados obtidos para o ponto operacional do problema de referência com a utilização do programa computacional desenvolvido, ou seja, os resultados das taxas de transferência de calor (condensador, evaporador), das potências (compressor de baixa pressão e compressor de alta pressão) e COP são apresentados na Tabela 7.

Tabela 7 – Resultados obtidos para o ponto operacional do problema de referência

Dado de saída analisado	Valor calculado
COP	2,5908
Taxa de calor transferido no evaporador (kW)	19,3195
Taxa de calor no condensador (kW)	26,7766
Potência no compressor de alta pressão (kW)	5,3331
Potência no compressor de baixa pressão (kW)	2,124

7.6. VALIDAÇÃO DOS RESULTADOS DA ANÁLISE ENERGÉTICA

A Tabela 8 apresenta os valores de referência e os valores calculados para o Coeficiente de Performance (COP), taxa de calor e potência.

tolerância de 1% deve-se exclusivamente a essa diferença no estado de referência e que isso não interfere nos valores das variações de entalpia, os erros relativos das taxas de transferência de energia foram apresentados de antemão na sexta seção. No entanto, na sétima seção, a apresentação dos erros relativos das taxas de transferência de energia faz parte da validação da análise energética.

Tabela 8 – Valores de referência do COP e calculados através do MatLab

Dado de saída analisado	Valor calculado	Valor de referência	Erro percentual
COP	2,5908	2,59	0,03%
Taxa de calor transferido no evaporador (kW)	19,3195	19,33	0,05%
Taxa de calor no condensador (kW)	26,7766	26,7743	0,01%
Potência no compressor de alta pressão (kW)	5,3331	5,33267	0,01%
Potência no compressor de baixa pressão (kW)	2,124	2,1175	0,31%

Conforme é possível verificar, os valores do erro percentual para todos os dados de saída analisados – taxa de calor, potência e COP - são inferiores a 1%. Assim, análise energética do sistema no ponto operacional do problema de referência está validada.

8 OTIMIZAÇÃO ENERGÉTICA DO CICLO DE REFRIGERAÇÃO POR COMPRESSÃO EM MÚLTIPLOS ESTÁGIOS

Esta seção apresenta o problema de otimização do ciclo de refrigeração por compressão em múltiplos estágios, de modo que o objetivo é a maximização do coeficiente de performance (COP) do ciclo. Assim sendo, foram levantados alguns estudos semelhantes na literatura que serviram tanto para demonstrar a importância da otimização para os projetos de engenharia como para servir de referência na formulação, resolução e validação do problema de otimização do presente trabalho.

8.1. CRITÉRIO PARA A VALIDAÇÃO DA OTIMIZAÇÃO

Para a validação desta parte do trabalho, será utilizado um estudo de otimização encontrado em Dincer, Rosen e Ahmadi (2017) do mesmo tipo de sistema de refrigeração do problema de referência (sistema de refrigeração por compressão de vapor em múltiplos estágios), já apresentado, simulado e analisado nas seções 6 e 7.

Como o sistema a ser otimizado é aquele do problema de referência, o estudo similar de otimização será utilizado somente para validar por comparação a parte de otimização do problema de referência. Portanto, o ciclo do estudo similar de otimização tem um ponto operacional diferente do problema de referência e não é utilizado.

Ou seja, o problema similar da literatura não é um segundo problema de referência, mas somente serve para validar a parte de otimização do único problema de referência do presente trabalho.

8.1.1. Estudos Similares Encontrados Na Literatura

O estudo encontrado em Dincer, Rosen e Ahmadi (2017) é desenvolvido com mesmo tipo de ciclo de refrigeração do problema de referência (ciclo de refrigeração por compressão de vapor em múltiplos estágios) e, por isso, o estudo do Dincer e todas as respectivas análises realizadas serviram como diretrizes para a aplicação da otimização ao problema de referência.

Assim sendo, o presente trabalho utilizou a mesma função objetivo, as mesmas restrições, as mesmas variáveis de decisão e os mesmos limites para as variáveis de decisão do estudo de otimização similar encontrado na literatura. A escolha das variáveis de decisão requer um estudo a parte, além de experiência com otimização e com o ciclo estudado. Para este trabalho, as variáveis de decisão foram escolhidas de acordo com a análise de sensibilidade do COP realizada para alguns parâmetros de interesse (ou seja, por meio da realização dos estudos paramétricos) selecionados por Dincer, Rosen e Ahmadi (2017)¹⁸.

As condições operacionais não otimizadas do estudo similar da literatura são encontradas no Anexo B e os gráficos obtidos pelo referido estudo para verificar os efeitos das variáveis de decisão (eficiência isentrópica do compressor, pressão do evaporador, pressão do condensador e pressão do separador de líquido) sobre o COP em uma otimização de sistema de refrigeração por compressão de vapor em múltiplos estágios, são encontrados no Anexo C.

8.2. FUNÇÃO OBJETIVO DO PROBLEMA DE OTIMIZAÇÃO

A função objetivo para a otimização do sistema estudado na sétima seção e otimização (seção 8) é o coeficiente de performance - COP, apresentado na Equação 9.

$$\text{COP}_R = \frac{\dot{Q}_{evap}}{\dot{W}_{total}} \quad (9)$$

¹⁸ Por se tratar de um projeto de graduação e primeiro contato com otimização, todas essas etapas preliminares necessárias para a própria otimização (escolha da função objetivo, escolha das variáveis de decisão e seus limites, realização dos estudos paramétricos, etc.) foram seguidas de acordo com o estudo do Dincer, Rosen e Pouria (2017) pelo fato deste estudo analisar exatamente o mesmo tipo de ciclo de refrigeração do problema de referência.

Na Equação 9, \dot{Q}_{evap} representa a taxa de calor retirado do ambiente refrigerado e \dot{W}_{total} representa a potência total consumida pelo sistema, ou seja, a soma das potências consumidas em ambos os compressores.

8.3. RESTRIÇÕES DO PROBLEMA DE OTIMIZAÇÃO

Conforme descrito na seção 4, as restrições de problemas de otimização são as limitações impostas pelas variáveis do sistema e em sistemas térmicos são representadas essencialmente por leis de conservação, nesse caso envolvendo massa e energia.

Assim sendo, as restrições do problema de otimização do presente trabalho são: atendimento aos princípios de conservação de massa e energia e aos limites das variáveis de decisão. Para os balanços de massa e energia é utilizado o simulador de processos IPSEpro, o que reduz o tamanho do problema de otimização, dispensando-se a incorporação de uma série de equações de restrição impostas pelos balanços termodinâmicos. Além disso, torna-se possível a utilização de modelos termodinâmicos mais sofisticados para os componentes do sistema, os quais nem sempre podem ser facilmente explicitados para serem incorporados à rotina de otimização (Vieira, 2003). Para o atendimento das restrições associadas aos limites (inferior e superior) das variáveis de decisão é utilizada uma solução¹⁹ que penaliza²⁰ a função utilizada para o cálculo do COP (função objetivo) caso as variáveis estejam fora de seus limites inferior e superior.

Nenhuma restrição adicional foi implementada na solução computacional para a otimização do problema de referência.

8.4. VARIÁVEIS DE DECISÃO

O presente trabalho utilizou as variáveis de decisão com base no estudo de otimização similar encontrado na literatura. No referido estudo, houve a seleção de algumas variáveis de decisão e, para concretização das mesmas, foram realizados estudos paramétricos para avaliar

¹⁹ Essa solução pode ser implementada, a princípio, em qualquer *software* de computação científica que possa ser integrado com um simulador de processos. No caso do presente trabalho, essa solução foi implementada no MatLab.

²⁰ Penalizar um indivíduo não viável significa acrescentar pesos a cada restrição que for violada pela solução que ele representa. Ou seja, a penalização manipula o valor da função objetivo do indivíduo de forma a torná-lo menos apto e, conseqüentemente, diminuir sua probabilidade de permanecer como membro das próximas gerações.

como cada variável impacta o COP do sistema. Isso porque, por exemplo, o COP pode aumentar com o aumento de algumas variáveis e diminuir com o aumento de outras variáveis. Ou seja, pode haver competição entre variáveis.

8.4.1. Estudos Paramétricos

Para verificar o efeito sobre o COP quando as variáveis de decisão trabalham em uma determinada faixa de valores, uma boa estratégia é integrar o simulador de processos, IPSEpro com o software de computação científica, MatLab. Os códigos computacionais desenvolvidos para os estudos paramétricos são encontrados no Apêndice B do presente relatório.

Assim, por meio de programação, o MatLab define o valor de um parâmetro de interesse dentro de uma faixa pré-determinada, executa o IPSEpro, obtém as propriedades ponto a ponto no ciclo, e calcula por fim o resultado do COP. Por se tratar de uma faixa de valores avaliada, o código é executado um determinado número de vezes, utilizando um laço de repetição²¹.

Os estudos paramétricos realizados no presente trabalho utilizaram faixas que se encontram dentro dos mesmos limites das variáveis de decisão, utilizados por Dincer, Rosen e Ahmadi (2017)²². Em linhas gerais, os limites das variáveis de decisão estão associados a restrições existentes na prática, como por exemplo: limites metalúrgicos que restringem temperaturas; limites operacionais em equipamentos que restringem variáveis como pressão, temperatura, vazão mássica; disponibilidade de tecnologia no mercado como por exemplo faixas de desempenho de equipamentos (por exemplo, eficiência isentrópica, rendimento térmico, COP, entre outros).

²¹ Neste trabalho foi utilizado um laço de repetição do tipo *for-end*, ou seja, com um número de repetições pré-definido. Há ainda outro tipo de laço de repetição que é o do tipo *while-end*, onde não é pré-definido um número de repetições. No entanto, o laço de repetição do tipo *while-end* não foi utilizado nos estudos paramétricos.

²² Levando em conta que os estudos paramétricos são utilizados para selecionar as variáveis de decisão (selecionadas por este trabalho de acordo com estudo similar de otimização), e não o contrário, a princípio, não seria necessária a realização destes estudos paramétricos. No entanto, esta etapa do projeto (análises paramétricas) foi de grande valor para aprendizagem em programação em MatLab, teoria da otimização, verificação da competição entre as variáveis de decisão e verificação do estudo paramétrico aqui desenvolvido com o estudo paramétrico do problema de otimização similar encontrado na literatura (Dincer, Rosen e Ahmadi, 2017)

8.4.1.1. Efeito da eficiência isentrópica do compressor no COP

O primeiro estudo paramétrico avalia o efeito da eficiência isentrópica do compressor sobre o COP. No problema de referência o valor da eficiência isentrópica é igual a 86% para ambos os compressores. A faixa de eficiência isentrópica estudada está compreendida entre 75% e 90%.

As figuras 46 e 47 apresentam o efeito da eficiência isentrópica sobre o COP, respectivamente, para o compressor de baixa pressão e para o compressor de alta pressão.

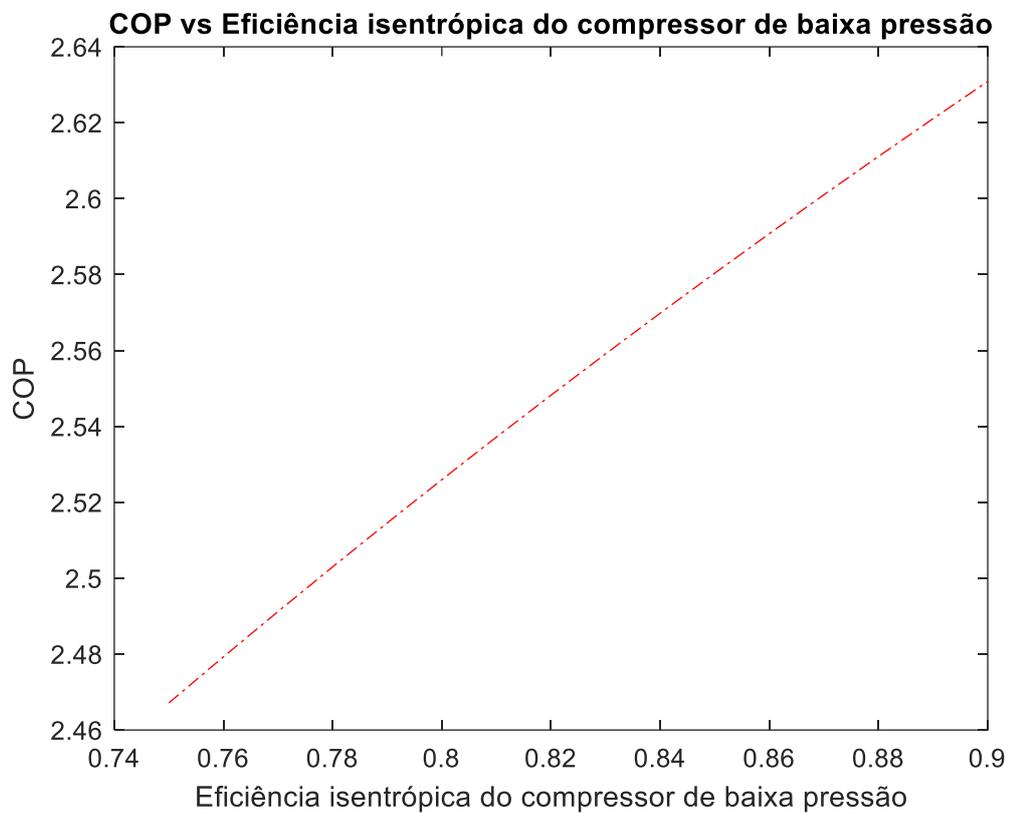


Figura 46 - Efeito da variação da eficiência isentrópica do compressor de baixa pressão sobre o COP

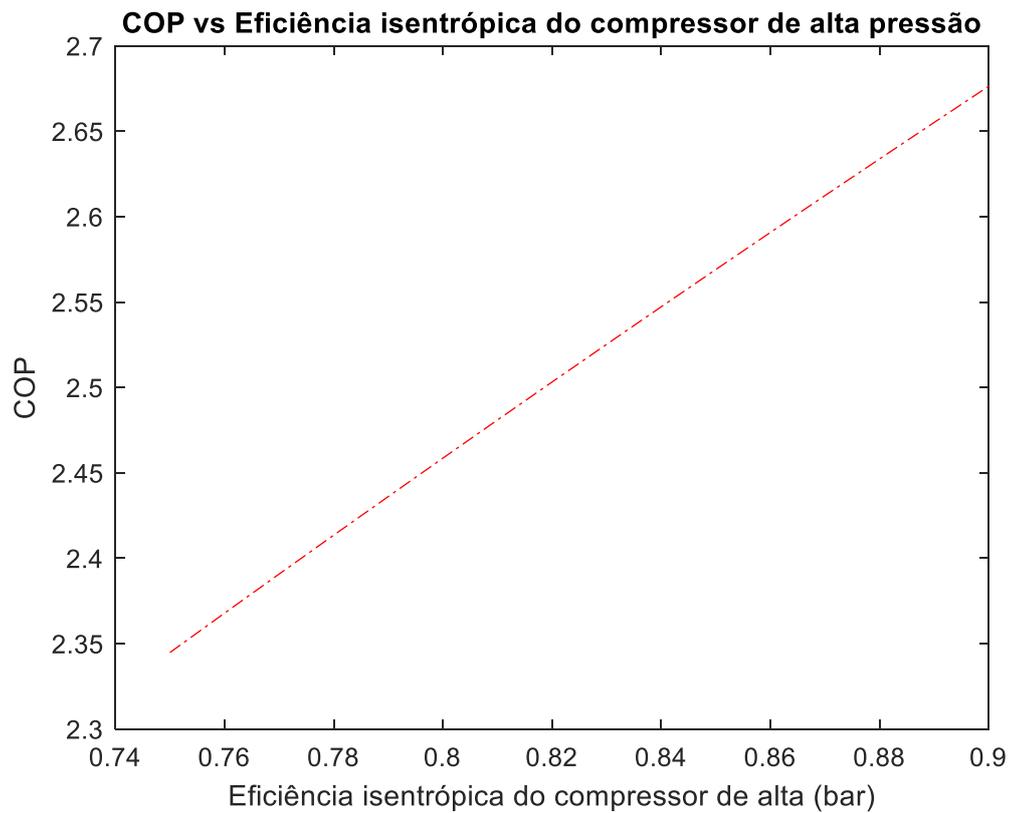


Figura 47– Efeito da variação da eficiência isentrópica do compressor de alta pressão sobre o COP

8.4.1.2. Efeito da pressão do evaporador no COP

Em seguida, foi analisado o efeito da pressão do evaporador no COP. A faixa de variação da pressão estudada está compreendida entre 100 kPa e 150 kPa. A Figura 48 ilustra o gráfico obtido com a verificação do COP quando há variação na pressão do condensador.

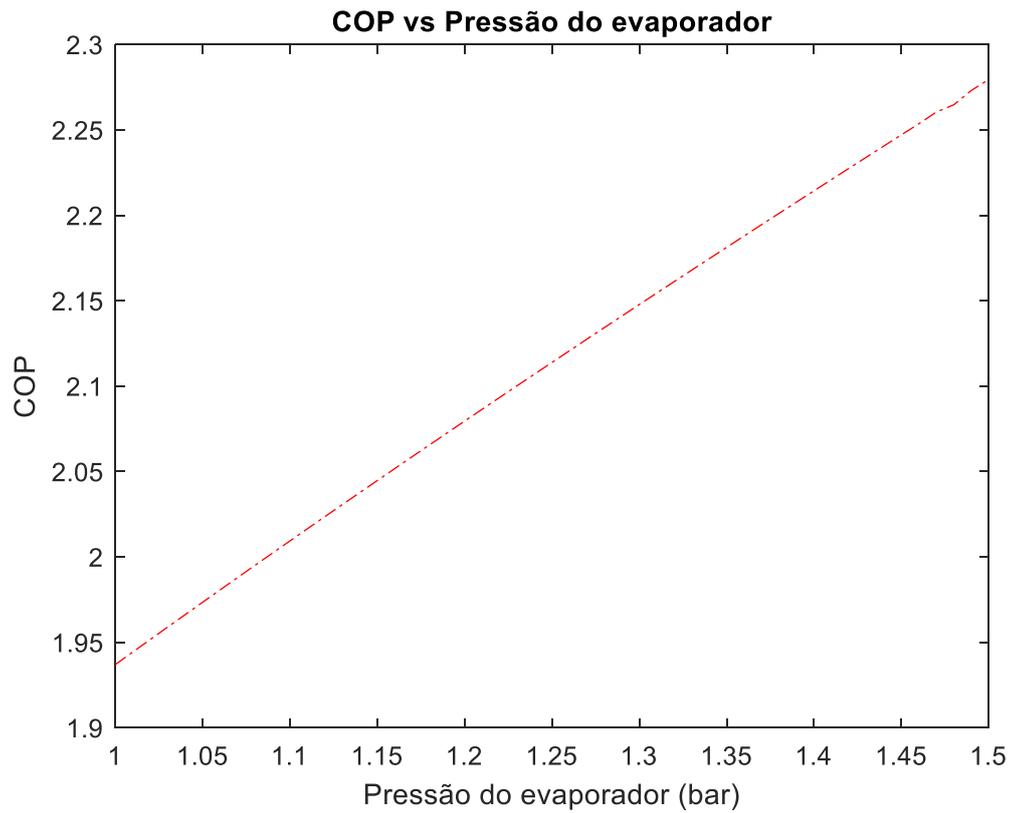


Figura 48– Efeito da variação da pressão do evaporador sobre o COP

8.4.1.3. Efeito da pressão do condensador no COP

A próxima análise identificou o efeito da pressão do condensador no COP. A faixa de pressão do condensador estudada está compreendida entre 1.000 kPa e 1.600 kPa. Os resultados são apresentados na Figura 49.

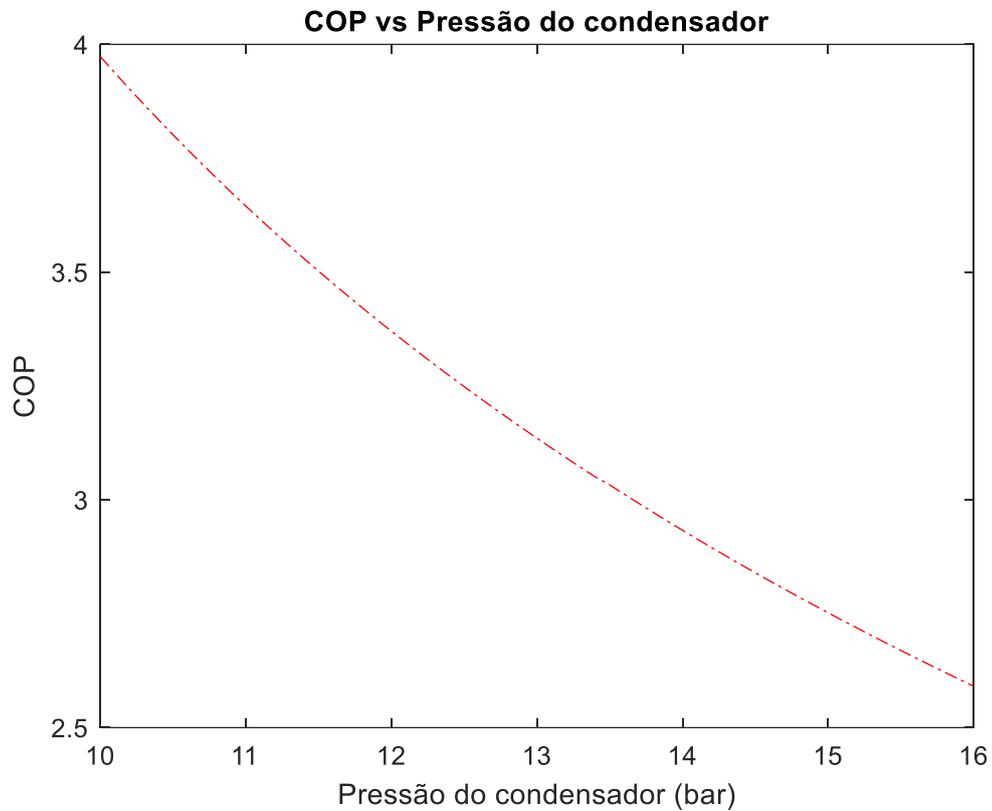


Figura 49– Efeito da variação da pressão do condensador sobre o COP

8.4.1.4. Efeito da pressão do separador de líquido no COP

Por fim, este estudo verificou o efeito da pressão do separador de líquido no COP. Para esta última análise paramétrica a faixa de valores de pressão do separador de líquido está compreendida entre 100 kPa e 700 kPa e de modo a verificar a tendência de ponto de máximo de COP próxima ao limite superior da pressão (700 kPa), este trabalho destacou a faixa de 600 kPa a 700 KPa. Os resultados encontrados estão ilustrados na Figura 50.

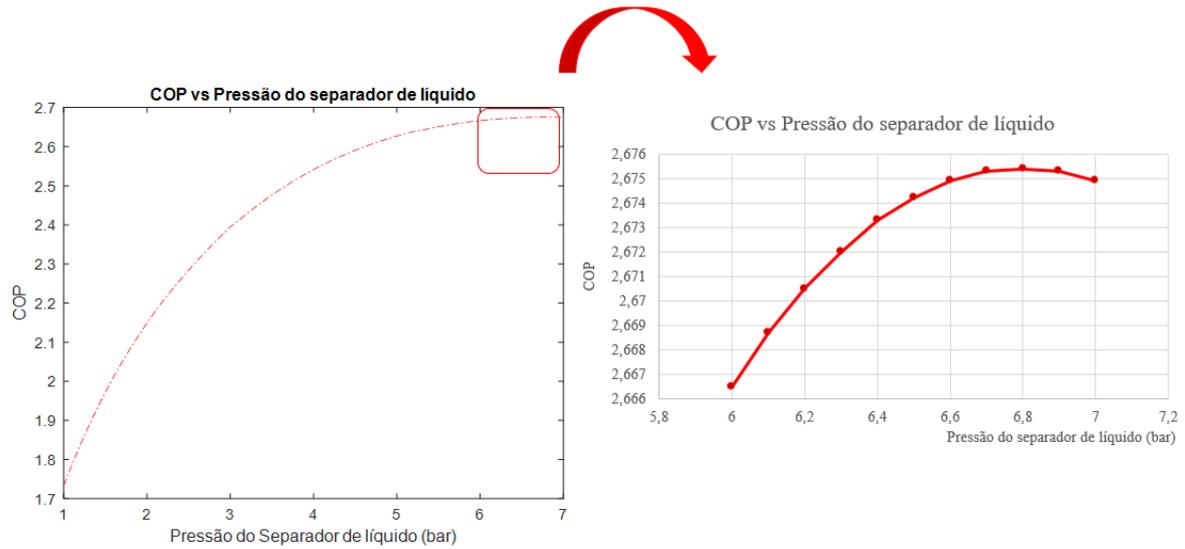


Figura 50– Efeito da variação da pressão do separador de líquido sobre o COP

8.4.1.5. Resultados e discussão das análises paramétricas

É possível observar um comportamento linear (ou muito próximo de linear) para toda a faixa estudada (de 75% a 90%) da eficiência isentrópica dos compressores de alta e baixa pressões e para toda a faixa estudada (100 kPa a 150 kPa) de pressão do evaporador. Além disso, foi observado que o COP aumenta com o aumento das eficiências isentrópicas dos compressores e com o aumento da pressão do evaporador. Por outro lado, para toda faixa de pressão do condensador (1.000 kPa e 1.600 kPa), a diminuição do COP foi observada, conforme a pressão do condensador aumentava.

Por fim, é possível observar um comportamento não linear do COP com tendência de ponto de máximo. Nesse caso, a faixa à esquerda do ponto de máximo (próximo ao limite superior de 7 bar da pressão do separador de líquido) apresenta um aumento do COP com o aumento da pressão do separador de líquido, ao passo que a estreita faixa à direita do ponto de máximo apresenta uma pequena diminuição do COP com o aumento da pressão do separador de líquido. Ainda, é possível observar um aumento rápido do COP na faixa de 100 kPa a 400 kPa de pressão do separador de líquido. A partir de 400 kPa, até 700 kPa, o gráfico começa a apresentar um aumento lento do COP.

8.4.1.6. Validação dos resultados das análises paramétricas

Para a validação dos resultados foi realizada a comparação dos estudos paramétricos desenvolvidos no presente trabalho com estudos similares encontrados em Dincer, Rosen e Ahmadi (2017).

Com a comparação dos gráficos desenvolvidos no presente trabalho com aqueles desenvolvidos no problema de otimização da literatura, foi possível observar similaridade no comportamento das variáveis de decisão em todos os estudos paramétricos. Desta forma, os estudos paramétricos estão validados, o que era esperado para sistemas térmicos de mesma configuração estrutural.

8.5. METODOLOGIA

O MatLab é um software interativo robusto voltado para o cálculo numérico. O MatLab integra análise numérica, cálculo matricial, processamento de sinais e construção de gráficos. A escolha desse software se deve à sua possibilidade de integração com o ambiente de simulação do IPSEpro. Além disso, a integração é feita de maneira muito simples e eficiente devido à existência da interface COM em ambos os *softwares*.

8.5.1 A Ferramenta de Otimização do MatLab

A *Global Optimization Toolbox* fornece ferramentas para calcular parâmetros que minimizam ou maximizam funções objetivo enquanto satisfazem as restrições impostas pelo problema de otimização.

Uma dessas ferramentas é o comando *ga*, utilizado para a aplicação de Algoritmo Genético, que é um método para resolver problemas de otimização baseado na seleção natural. O algoritmo genético modifica e cruza repetidas vezes uma população de soluções individuais. Em cada geração, ele seleciona aleatoriamente indivíduos da população atual para serem pais e os usa para produzir filhos para a próxima geração. Depois de sucessivas gerações, a população original "evolui" para uma população ótima.

Neste trabalho, foi definida a utilização de algoritmo genético para a otimização do problema de referência de acordo com Rodrigues (2013), Cunha (2016), Maudonet (2017) e

Dincer, Rosen e Ahmadi (2017). Além disso, a escolha se deve principalmente à possibilidade de validar o algoritmo genético a partir da otimização de um sistema de refrigeração similar ao problema de referência encontrado em Dincer, Rosen e Ahmadi (2017) com mesma configuração estrutural e com o mesmo fluido refrigerante R-134a.

De acordo com Dincer, Rosen e Ahmadi (2017), a utilização de algoritmo genético se deve ao fato de as variáveis de decisão afetarem de maneiras distintas o COP do sistema. Além disso, a escolha do algoritmo genético foi realizada também por conta da robustez desse tipo de algoritmo, o que é extremamente interessante para a otimização de sistemas térmicos²³.

8.5.1.1 O programa desenvolvido utilizando algoritmo genética

Nas primeiras linhas de comando do programa desenvolvido, foram definidos a quantidade de variáveis de decisão²⁴ (*nvar*), o limite inferior (LI) e o limite superior (LS) das variáveis e um vetor inicial de variáveis de decisão (*x0*). Além disso, as variáveis LI e LS foram declaradas como variáveis globais²⁵. Abaixo segue a representação escrita desta parte do código.

```
nvar = 5;
```

```
LI = [0.75, 0.75, 1, 10, 1];26
```

```
LS = [0.9, 0.9, 1.5, 16, 7];
```

```
x0 = [0.86, 0.86, 1.25, 14, 4.5];
```

```
global LI LS
```

²³ Os algoritmos genéticos são métodos robustos apropriados para a solução de problemas mistos com variáveis contínuas e discretas e espaços de busca descontínuos e apresentam grande probabilidade de alcançar o ótimo global.

²⁴ No algoritmo genético, um indivíduo representa uma variável de decisão ou um conjunto de variáveis de decisão. A população representa um conjunto de indivíduos de uma determinada geração.

²⁵ De acordo com Gilat (2011), toda função possui um conjunto particular de variáveis locais que não são compartilhadas com outras funções ou com a área de trabalho da janela Command Window e/ou dos programas. Entretanto, é possível tornar uma variável comum (pública) às várias funções e, possivelmente, a toda área de trabalho. Nesse caso a variável deve ser declarada como uma variável global (*global variable*).

²⁶ Os limites superior e inferior foram retirados do problema similar encontrado em Dincer, Rosen e Ahmadi (2017).

A Tabela 9 mostra os limites superior e inferior das variáveis de decisão.

Tabela 9 – Limites superior e inferior das variáveis de decisão

Variável de decisão	Limite inferior	Limite superior
Eficiência isentrópica do compressor de baixa pressão	0,75	0,9
Pressão do evaporador (bar)	1	1,5
Pressão do compressor (bar)	10	16
Pressão de separador de líquido (bar)	1	7
Eficiência isentrópica do compressor de alta pressão	0,75	0,9

A seguir, foi implementada a função “*str2func*” para criar um identificador de função, neste caso para a função que realiza o cálculo do oposto do COP²⁷.

8.5.1.1 Seleção dos parâmetros do algoritmo genético

As próximas linhas do código são utilizadas para a definição dos parâmetros do algoritmo genético. A primeira linha define os limites inferior e superior da população inicial (vetor inicial de variáveis de decisão) - *Initial Range Population*, variável IR do MatLab. Após isso, é definido o tamanho da população por geração, *Population Size*, variável Population Size do MatLab.

Em seguida, foi definido o número de gerações, *Generations*, variável Generations do MatLab. Logo após, foi definido o número máximo de gerações sem melhoria. E, por fim, foi definido o vetor inicial de variáveis de decisão, variável IP do MatLab. Abaixo segue a representação desta parte do código.

```
fun = str2func('funcop');
```

²⁷ De acordo com Balaji (2019), a maximização de uma função y_1 equivale à minimização de $-y_1$, o que é denominado princípio da dualidade.

$IR = [LI;LS];$

$PopulationSize = 5*numvar;$ ²⁸

$Generations = 5*numvar;$

$StallGenLimit = 2*numvar;$

$IP = x0;$

No entanto, a implementação desses parâmetros não ocorre de forma automática no algoritmo genético, exigindo a utilização de uma estrutura específica denominada “*gaoptimset*”.

Com o comando *gaoptimset* é possível configurar o valor de vários parâmetros do algoritmo genético. Abaixo segue a sintaxe da estrutura *gaoptimset* implementada no código do programa desenvolvido no presente trabalho.

```
options = gaoptimset ('Display', 'iter', 'PlotFcns', @gaplotbestf, 'PopulationSize',
PopulationSize, 'PopInitRange', IR, 'Generations', Generations, "StallGenLimit",
StallGenLimit, 'InitialPopulation', IP);
```

Nessa estrutura do *gaoptimset*, a definição dos valores é realizada por meio da identificação do parâmetro por aspas simples, seguida da variável definida anteriormente. Os comandos “*Display*”, “*iter*” e “*PlotFcns*”, escritos nesta ordem, definem a apresentação gráfica do melhor resultado da função objetivo à cada interação.

O comando “*gaplotbestf*” plota o melhor resultado e o resultado médio do valor da função objetivo por geração.

Por fim, representada abaixo, a linha de código que contém a sintaxe do algoritmo genético para a minimização da função objetivo.

```
[x, f, reason, output, finalpopulation, finalscores] = ga (fun, numvar, options);
```

²⁸ Os valores dos parâmetros do algoritmo genético foram definidos com base em Costermani (2012).

Nesta sintaxe, “x” e “f” retornam o vetor de variáveis e o valor final da função, “output” retorna um número entre -5 e 4, referente à razão do algoritmo genético ter sido finalizado e “finalpopulation” e “finalscores”, parâmetros que serão retornados à cada interação.

No presente trabalho, não foram realizados os estudos de sensibilidade no algoritmo genético. Os parâmetros (população inicial, tamanho da população, número máximo de gerações e gerações sem melhoria) foram escolhidos segundo Costermani (2012) e a sua utilização se mostrou ser satisfatória com base nos resultados encontrados.

A Figura 51 mostra o gráfico gerado durante o processo de otimização com o melhor valor da função objetivo e o seu valor médio a cada geração do algoritmo genético.

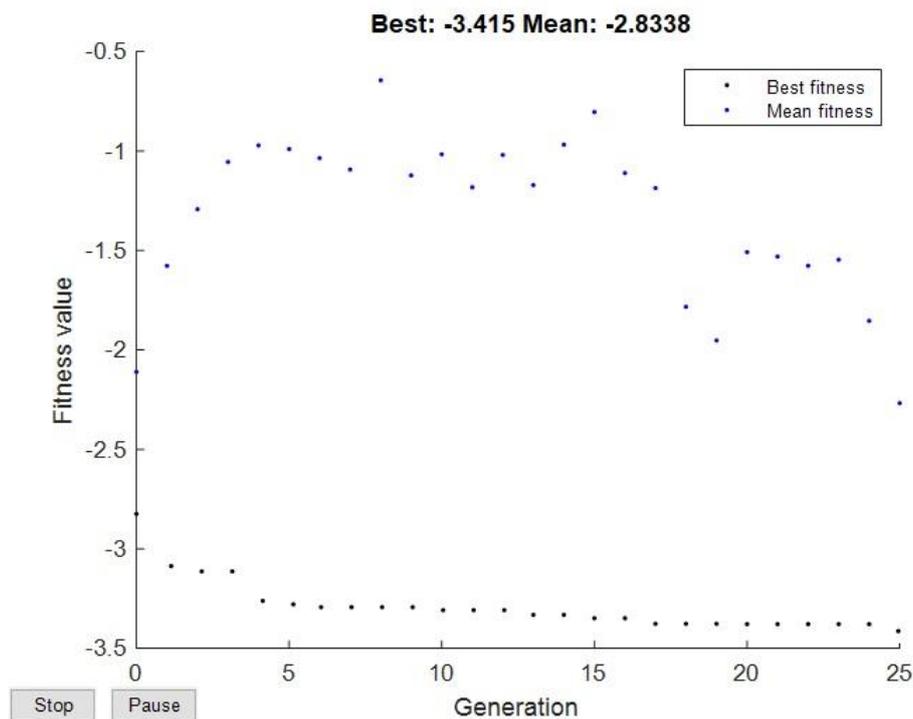


Figura 51– Gráfico gerado durante o processo de otimização

A Figura 52 mostra o gráfico do valor da função objetivo (COP) em função do número de chamadas da função objetivo (valor cumulativo).

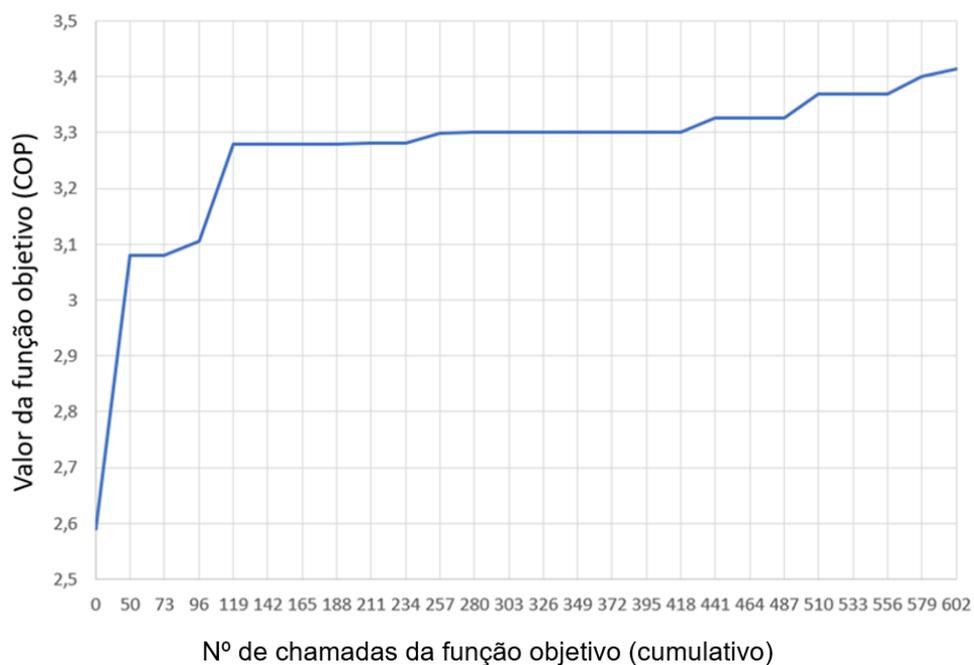


Figura 52– Gráfico do valor da função objetivo (COP) em função do número de chamadas da função objetivo

Os valores relativos ao número da geração, número de chamadas da função objetivo, melhor resultado na geração, valor médio na geração e número de gerações sem melhoria estão presentes na Tabela 10.

Tabela 10 – Valores gerados pelo algoritmo genético a cada geração.

Generation	Func-count	Best $f(x)$	Mean $f(x)$	Stall Gen
-	-	2,59	-	-
1	50	3,08	1,777	0
2	73	3,08	1,464	1
3	96	3,105	1,405	0
4	119	3,279	1,136	0
5	142	3,279	1,325	1
6	165	3,279	1,408	2
7	188	3,279	1,712	3
8	211	3,281	1,317	0
9	234	3,281	1,173	1
10	257	3,298	1,04	0
11	280	3,3	1,221	0
12	303	3,3	1,23	1
13	326	3,3	1,22	2
14	349	3,3	1,062	3
15	372	3,3	0,7105	4
16	395	3,3	1,043	5
17	418	3,3	1,223	6
18	441	3,326	1,371	0
19	464	3,326	1,728	1
20	487	3,326	1,747	2
21	510	3,369	1,404	0
22	533	3,369	1,074	1
23	556	3,369	1,604	2
24	579	3,401	2,123	0
25	602	3,415	2,834	0

Legenda: *Generation* = Gerações; *Func-count* = Número de chamadas da função objetivo (valor cumulativo); *Best $f(x)$* = Melhor resultado na geração; *Mean $f(x)$* = Valor médio na geração; *Stall Gen* = Número de gerações sem melhoria

8.6. DIAGRAMA ESTRUTURAL DO PROGRAMA COMPUTACIONAL DESENVOLVIDO NO MATLAB

Com o objetivo de descrever graficamente como as diferentes partes do programa estão conectadas, foi criado o diagrama estrutural do programa computacional relativo ao algoritmo genético desenvolvido no MatLab, como ilustra a Figura 53.

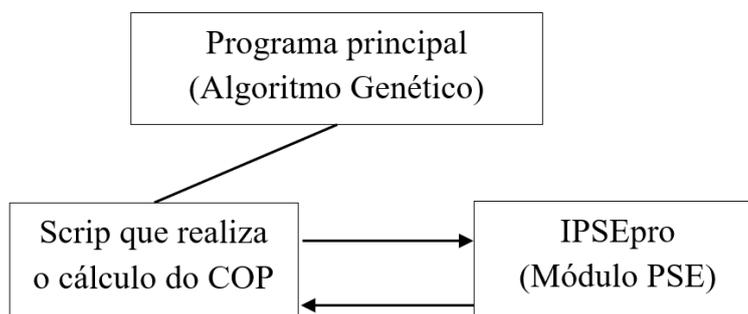


Figura 53- Diagrama estrutural do programa computacional desenvolvido no MATLAB referente ao algoritmo genético

8.7. FLUXOGRAMA DOS CÓDIGO COMPUTACIONAIS DESENVOLVIDOS

A Figura 54 ilustra o fluxograma com o passo a passo do processo, bem como as tomadas de decisão do algoritmo genético para o objetivo proposto: minimizar a função inversa do COP respeitando os limites superior e inferior das variáveis de decisão, sem erros ou alertas na simulação do IPSEPro.

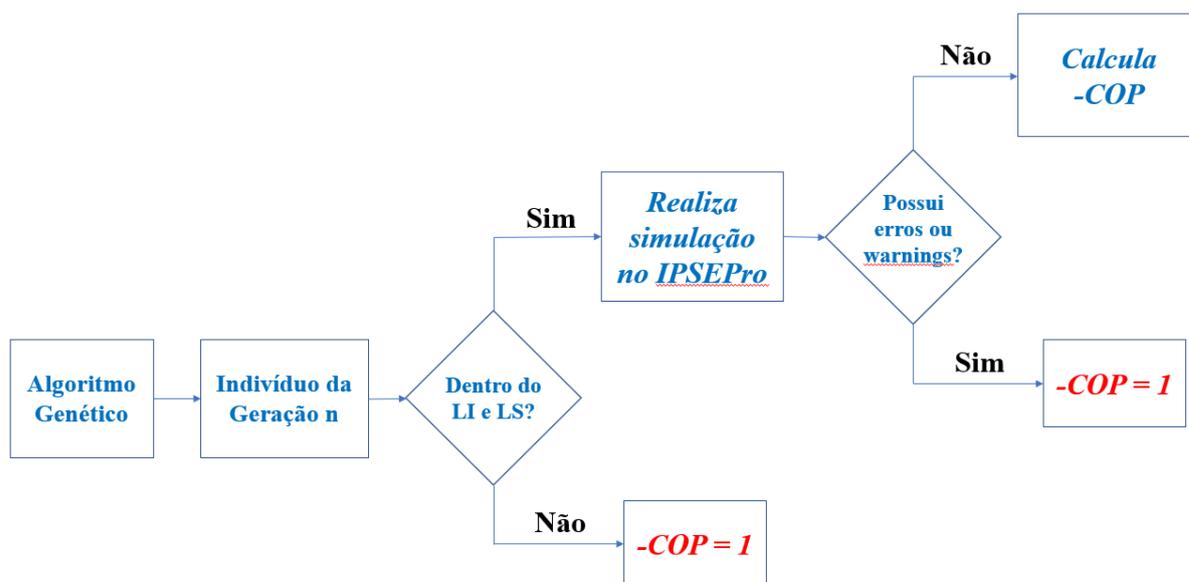


Figura 54– Fluxograma do Algoritmo Genético, bem como suas tomadas de decisão

8.8. RESULTADOS E DISCUSSÃO DA OTIMIZAÇÃO

O método de otimização por algoritmo genético foi aplicado no problema de referência a fim de encontrar o ponto ótimo, ou seja, o melhor vetor de variáveis de decisão (vetor ótimo) que maximiza o valor do COP (valor ótimo).

A figura 55 apresenta os resultados obtidos para a simulação do problema de referência no ponto ótimo.

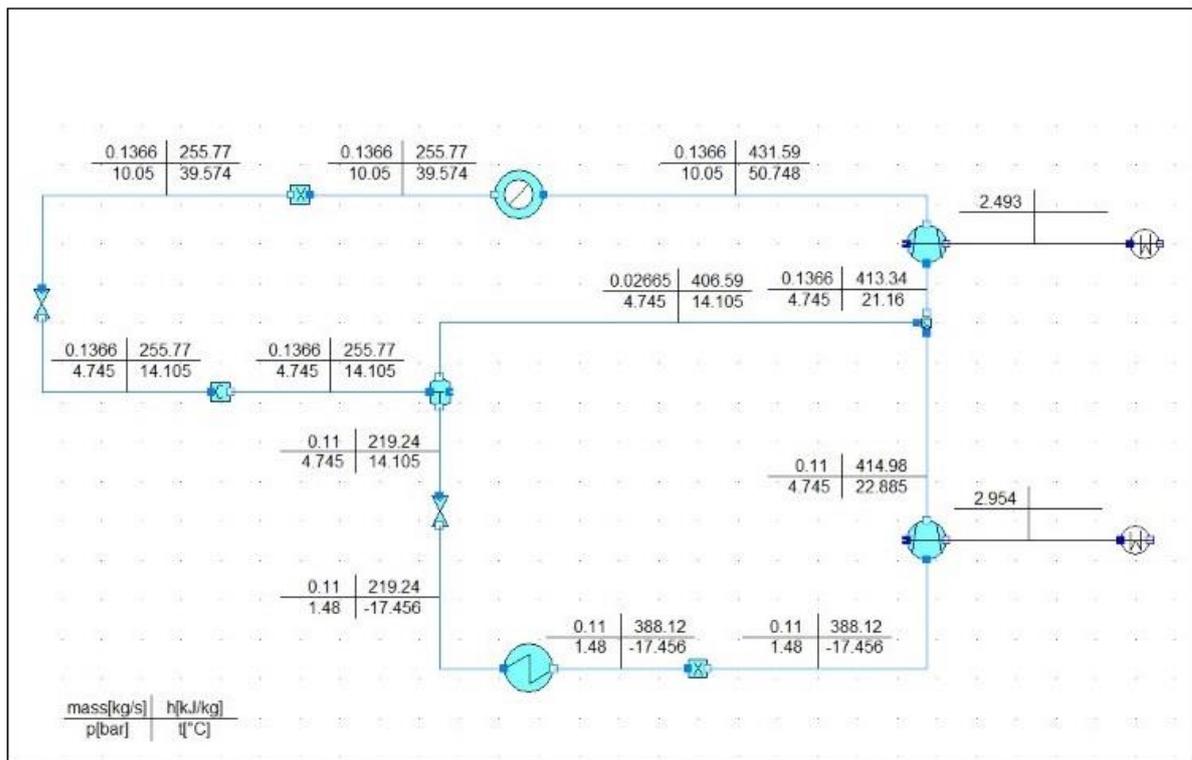


Figura 55- Resultados obtidos para a simulação do problema de referência no ponto ótimo

A Tabela 11 apresenta os resultados do problema de referência otimizado (PRO) e a comparação com os resultados do problema de referência não otimizado (PRNO).

Tabela 11 – Valores das variáveis de decisão, das taxas de transferência de energia e do COP do Problema de Referência otimizado em comparação com os resultados do problema de referência não otimizado.

Parâmetros	PRNO	PRO
Taxa de calor absorvido pelo evaporador (kW)	19,31	18,57
Potência consumida pelos compressores, (kW)	7,45	5,44
COP	2,59	3,41
Taxa de calor rejeitado pelo condensador (kW)	26,77	24,01
Pressão no evaporador (bar)	2	1,48
Pressão no Condensador (bar)	16	10,05
Pressão no Separador de Líquido (bar)	4,5	4,74
Eficiência do compressor de baixa pressão	0,89	0,89
Eficiência do compressor de alta pressão	0,89	0,88

Ao realizar a análise dos resultados, é possível perceber que o COP otimizado é 31% maior que o COP no problema de referência não otimizado (PRNO). No entanto, a taxa de calor absorvido pelo refrigerador (efeito da refrigeração) é quase 1kW menor no problema de referência otimizado (PRO). Essa queda é compensada por uma potência consumida pelos compressores significativamente menor no ponto ótimo. No exemplo do sistema de refrigeração, esses resultados mostram que a otimização do COP nem sempre se traduz em maiores efeitos de refrigeração. O seu objetivo, no entanto, está em produzir o maior efeito de refrigeração possível por unidade de trabalho realizado.

Ainda na análise dos resultados, tanto a pressão do condensador quanto a pressão do evaporador seguiram os estudos paramétricos e tiveram valores no PRO próximos dos limites inferior e superior das variáveis, respectivamente. É possível perceber que a pressão ótima do separador de líquido está no meio da faixa, embora o COP aumente com o aumento da variável (estudo paramétrico). Isso evidencia que o ponto ótimo não necessariamente se encontra próximo aos valores das variáveis de decisão que fizeram o COP ser máximo nos estudos paramétricos.

Como as variáveis de decisão competem (o aumento de uma variável de decisão faz o COP aumentar enquanto o aumento de outra variável de decisão faz o COP diminuir), o algoritmo genético avalia o efeito global das alterações simultâneas de todas as variáveis de decisão, retornando o ótimo no final do processo.

Por fim, em relação aos compressores, o ponto ótimo se manteve associado ao valor do limite superior da variável de eficiência isentrópica, seguindo o estudo paramétrico.

8.9. VALIDAÇÃO DOS RESULTADOS DA OTIMIZAÇÃO

Para a validação dos resultados da otimização do problema de referência, foi utilizada uma estratégia na qual o programa computacional desenvolvido foi aplicado a um problema similar otimizado por Dincer, Rosen e Ahmadi (2017). Essa estratégia foi utilizada porque o problema de referência do presente trabalho não foi otimizado por Çengel e Boles (2011), o que impossibilita a utilização do critério de validação de resultados escolhido (90% ou mais dos pontos possuir erro relativo menor ou igual à 1% em comparação aos resultados da literatura).

Os resultados obtidos na otimização do problema similar foram então comparados com os resultados obtidos por Dincer, Rosen e Ahmadi (2017), tornando a validação do algoritmo genético possível e, por consequência, da otimização do problema de referência.

A Tabela 12 apresenta a comparação dos resultados da simulação do problema similar, ainda não otimizado, em comparação aos resultados fornecidos pela literatura.

Tabela 12 - Comparação entre os resultados da simulação com o IPSEpro do problema similar e os encontrados na literatura.

Grandezas de interesse	Resultados da simulação com o IPSEpro	Resultados encontrados na literatura	Erro relativo (%)
Taxa de calor absorvido pelo evaporador (kW)	28,54	28,5	0,14
Potência consumida pelos compressores (kW)	11,41	11,4	0,1
COP	2,50	2,5	0,003

De acordo com a Tabela 12, todos os resultados encontrados estão abaixo do erro previsto de 1% e, portanto, a simulação é considerada validada. Em seguida, foi realizada a otimização do problema similar por algoritmo genético. A Tabela 13 apresenta a comparação entre os resultados da otimização com os resultados do estudo similar encontrados na literatura.

Tabela 13 - Comparação entre os resultados da otimização com os resultados do problema similar encontrado na literatura

Grandezas de interesse	Resultados da otimização com o programa desenvolvido	Resultados do problema similar otimizado por Dincer, Rosen e Ahmadi (2017)	Erro relativo (%)
Taxa de calor absorvido pelo evaporador (kW)	33,78	31,58	6,97
Potência consumida pelos compressores (kW)	10,13	10,5	3,52
COP	3,33	3	11,00
Taxa de calor rejeitada pelo condensador (kW)	43,92	42,09	4,3
Pressão do evaporador (bar)	1,48	1,50	1,33
Pressão do condensador (bar)	1019	1214	16,06
Pressão do separador de líquido (bar)	371	470	21,06
Eficiência isentrópica do compressor de baixa pressão	0,88	0,89	1,12
Eficiência isentrópica do compressor de alta pressão	0,89	-	-

Para validar o algoritmo genético desenvolvido, é preciso avaliar a disponibilidade de informações oferecidas pelo problema similar encontrado na literatura. São elas:

- 1) A função objetivo utilizada;
- 2) Quais e quantas foram as variáveis de decisão escolhidas;
- 3) Os limites das variáveis de decisão;
- 4) O método de simulação (exemplo, simulador de processos);
- 5) O ponto inicial definido;

6) Os parâmetros do algoritmo genético.

O estudo similar não forneceu informações sobre os itens 4²⁹, 5 e 6 acima. Ainda, o item 2 possui relevância, pois o estudo não informou a forma que utilizou a eficiência dos compressores. Ou seja, o estudo não informou se foi atribuída uma variável de decisão para cada eficiência isentrópica dos compressores de alta e baixa pressão (totalizando cinco variáveis de decisão) ou se foi atribuída uma única variável de decisão para as duas eficiências isentrópicas juntas (totalizando quatro variáveis de decisão).

Desta maneira, é natural que os erros relativos sejam consideráveis. A solução pelo algoritmo genético depende do ponto inicial escolhido, dos parâmetros do algoritmo genético, como tamanho da população, do número máximo de gerações, do número de gerações sem melhoria, do número de variáveis de decisão utilizadas e dos modelos utilizados para a simulação. De modo que todos esses pontos podem exercer grande influência na solução.

Apesar do apresentado, o algoritmo genético desenvolvido maximizou o COP do problema similar encontrado na literatura, e as variáveis do vetor ótimo ficaram dentro dos limites estabelecidos. Desta maneira, este trabalho considera o algoritmo desenvolvido consistente e, por consequência, a otimização do problema de referência validada.

²⁹ Dincer, Rosen e Ahmadi (2017) informaram que utilizaram o software *Engineering Equation Solver* - EES para realizar tanto os estudos paramétricos quanto a simulação. No entanto, como Dincer, Rosen e Ahmadi (2017) não apresentaram os modelos contidos no EES utilizados para os componentes, o item 4 é considerado como um dado não informado.

9 CONCLUSÃO

A partir dos fundamentos teóricos oferecidos pela revisão bibliográfica, foi possível identificar projetos semelhantes que reforçam a relevância dos sistemas de refrigeração na indústria, na economia e no dia a dia da sociedade e a importância de torná-los eficientes. Quando otimizados, há diversos ganhos associados, como redução de custos, uso racional dos recursos energéticos e redução de impactos negativos no meio ambiente.

O estudo utilizou a integração entre dois softwares, IPSEpro e MatLab, e aplicou um algoritmo genético comercial (*toolbox* do MatLab). A integração das ferramentas computacionais para a simulação e a otimização do sistema de refrigeração de múltiplos estágios se mostrou bastante eficiente, pois o simulador já tem incorporado as equações dos balanços termodinâmicos (conservação de massa e energia) como restrições do problema de otimização. Além disso, a utilização do IPSEpro facilita a consideração das restrições, pois o atendimento às mesmas é garantido pela simulação sem erros e sem warnings no simulador.

Associado ao bom desempenho do IPSEpro, o código escrito no MatLab para otimização do problema de referência, se mostrou uma solução adequada e bem sucedida. A rotina de otimização desenvolvida por meio da *toolbox* do MatLab (*Optimization toolbox*) que oferece ferramentas para implementação do método de algoritmo genético trouxe velocidade e precisão aos cálculos associados à otimização.

Entre as metodologias, ferramentas computacionais e conhecimentos técnicos adquiridos no trabalho, destacam-se: relevância do segmento da refrigeração no contexto da indústria e consumo energético no país; o aprofundamento dos ciclos de refrigeração e dos conceitos termodinâmicos; as abordagens teóricas de métodos de otimização, especialmente o algoritmo genético; e teoria e a aplicação prática de otimização energética; programação intermediária em MatLab; contato e simulação em um pacote profissional de refrigeração (IPSEpro); e integração entre dois *softwares* computacionais consagrados (MatLab e IPSEpro).

Rodrigues (2013) explica que a grande vantagem dos algoritmos genéticos está no fato de não ser necessário conhecer o funcionamento da função objetivo, apenas tê-la disponível para ser aplicada aos indivíduos e comparar os resultados. Algumas das vantagens do método de algoritmos genéticos são a potencialidade deste método de evitar a convergência prematura para os pontos de mínimos locais; a relativa facilidade de se realizar a programação

computacional; e ausência de restrições na formulação do ponto de partida dentro do espaço de busca (Rodrigues, 2013).

Por outro lado, como os indivíduos são gerados aleatoriamente, cada vez que o algoritmo é executado, ele pode retornar um valor ótimo diferente. Entretanto, com o aumento do número de indivíduos e gerações essa diferença pode ser minimizada.

O presente trabalho considera seu objetivo alcançado, uma vez que cada um dos objetivos foi atingido (realização da modelagem e simulação do problema de referência; realização da análise energética do problema de referência; e realização da otimização energética do problema de referência) e todos os resultados foram validados.

No entanto, como esperado, este trabalho encontrou inúmeros desafios e se limitou à otimização energética do ponto de vista essencialmente teórico, tendo valor para o meio acadêmico, nas áreas de pesquisa e educação em engenharia, e para o meio industrial, na aplicação a sistemas novos ou existentes. Após a otimização do sistema algumas análises adicionais poderiam ser realizadas, como a comparação entre o investimento financeiro em energia elétrica com e sem a otimização do sistema.

É possível constatar ainda que para atingir a otimização do problema de referência por meio da integração do simulador de processos profissional IPSEpro ao MatLab e utilização de um algoritmo genético comercial (toolbox do MatLab), o trabalho teve que cumprir etapas intermediárias.

A primeira delas foi desenvolver a modelagem e a simulação do problema de referência no simulador de processos IPSEpro, como mostrado na Seção 6. A validação dos resultados, foi realizada com cálculos de erros relativos das propriedades termodinâmicas ponto a ponto no sistema para o ponto operacional do problema de referência. Como critério de validação, foi estabelecido que se 90% dos dados estivessem com erro relativo inferior abaixo de 1%, o problema estaria validado.

Em seguida, na sétima seção, foi realizada a análise energética do problema de referência. Os resultados obtidos na simulação das taxas de transferências de calor (condensador, evaporador), das potências consumidas (compressor de baixa pressão e compressor de alta pressão) e do COP foram devidamente validados para o ponto de operação do problema de referência. O critério de validação utilizado foi o mesmo da seção anterior já apresentado.

Por fim, a validação da metodologia de otimização foi realizada na Seção 8. No entanto, para esclarecer e aprender a necessidade de uma otimização, foi feito um estudo paramétrico no qual constatou-se a competição entre algumas variáveis de decisão, deixando claro que alcançar um máximo global não envolvia apenas igualar o valor das variáveis de decisão aos valores que fazem a função objetivo (no nosso caso o COP) ser máximo no estudo paramétrico. Era necessário um método de otimização robusto para alcançar o máximo global.

Como o problema de referência não possuía dados de um estudo paramétrico, foi utilizado na Seção 8 um estudo da literatura sobre um ciclo de refrigeração por compressão de vapor em duplo estágio com o mesmo fluido refrigerante utilizado no problema de referência. Os dados calculados do estudo paramétrico foram devidamente validados, por meio da comparação entre o comportamento das curvas calculadas das alterações nas variáveis de decisão sobre o COP com o comportamento das curvas do estudo da literatura. Vale ressaltar que no presente trabalho foram utilizados os mesmos limites para as variáveis de decisão e mesmas diretrizes do problema similar de otimização encontrado na literatura.

Ainda na oitava seção, este trabalho utilizou o mesmo estudo da literatura para validar a metodologia de otimização do problema de referência e avaliar a consistência do programa. Para tanto, o programa computacional desenvolvido pelo presente trabalho foi aplicado ao problema do estudo similar ao problema de referência. Para sua validação foram utilizados a taxa de calor transferido, a potência consumida pelos compressores e o COP como parâmetros de interesse. Em seguida, determinou-se o projeto ótimo dentro da delimitação imposta pela formulação do problema de otimização. Assim, foi possível otimizar o problema da literatura e, conseqüentemente, o algoritmo desenvolvido foi considerado validado.

Desta forma, a otimização do problema de referência também foi considerada validada. Fundamentalmente, foi possível maximizar o COP respeitando as restrições para ambos os problemas, o da literatura e o problema de referência (Çengel E Boles, 2011). O COP não otimizado do problema de referência apresentou um valor de 2,59, enquanto o COP otimizado resultou em 3,41, aumentando a eficiência do sistema em 31,66%.

Abaixo estão listadas algumas sugestões de trabalhos futuros. Essas foram as principais oportunidades de pesquisa cogitadas ao longo do presente trabalho, que não foram implementadas por restrições de prazo e escopo:

- Estudos de pós-otimização com a avaliação do efeito da mudança dos parâmetros do algoritmo genético sobre a solução do problema de otimização;

- Estudos de pós-otimização com a avaliação da mudança do ponto inicial sobre a solução do problema de otimização;
- Avaliação de outras funções objetivo, por exemplo funções objetivo para custo e impacto ambiental, inclusive propostas para uma otimização multi-objetivo (quando existe mais de uma função objetivo);
- Outros estudos de otimização, como por exemplo, otimização exergética, exergoeconômica e exergoambiental.
- Simulação e otimização utilizando outros fluidos refrigerantes;
- Otimização do mesmo problema de referência utilizando outros métodos de otimização.
- Utilização do módulo PSOptimize do IPSEpro para a otimização do ciclo de refrigeração por compressão de múltiplos estágios.

REFERÊNCIAS

AHMADU, T. O.; FOLAYAN, C. O.; ANAFI, F. O. **Modeling, Simulation and Optimization of a Solar Absorption Air Conditioning System for an Office Block in Zaria, Nigeria.** International Journal of Air-Conditioning and Refrigeration, 2016.

ANDRETTA, M., **Determinação de raízes de funções: Método de Newton**, ICMC, USP. São Paulo, 2012.

BALAJI, C. **Thermal System Design and Optimization**. 2. ed. Ane Books Pvt, 2011.

BARALDI, Aline. **Análise, modelagem e otimização do ciclo de refrigeração de uma unidade de processamento de gás natural**. 2015. Dissertação (Mestrado Em Engenharia Química) - Escola de Engenharia, Universidade Federal Do Rio Grande Do Sul, Porto Alegre, 2015.

BELL, Ian;, WRONSKI, Jorrit; LEMORT, Vincent. **Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp**. Industrial & Engineering Chemistry Research, 2014. Disponível em: <www.coolprop.org>. Acesso em: 14 dez. 2020.

BELLOS, Evangelos; TZIVANIDIS, Christos; ANTONOPOULOS, Kimon A. **Exergetic, energetic and financial evaluation of a solar driven absorption cooling system with various collector types**. Applied Thermal Engineering, 2016.

BRANDÃO, Rafael Cardoso. **Análise do resfriamento distrital**. 2018. Dissertação (Mestrado em Ciências em Engenharia Mecânica) - COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2018.

CENTRAIS ELÉTRICAS BRASILEIRAS, FUPAI/EFFICIENTIA. **Eficiência Energética em Sistemas de Refrigeração Industrial e Comercial**. Rio de Janeiro: Eletrobrás, 2005. Disponível em: < <https://bucket-gw-cni-static-cms->

[si.s3.amazonaws.com/media/uploads/arquivos/LivroRefrigeracao.pdf](https://s3.amazonaws.com/media/uploads/arquivos/LivroRefrigeracao.pdf) >. Acesso em 25 de outubro de 2020.

CONFEDERAÇÃO NACIONAL DA INDÚSTRIA. **Eficiência energética na indústria. Entre nessa corrente.** Brasília: Portal da Indústria. Disponível em: < <http://www.portaldaindustria.com.br/cni/canais/eficiencia-energetica/>> Acesso em 25 de outubro de 2020.

COSTERMANI, Hugo Rodolfo. **Termoeconomia aplicada à análise e otimização de sistemas de cogeração a biogás.** 2012. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Mecânica) - Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2012.

COPPIN, Ben. **Artificial Intelligence Illuminated.** 1. ed. Mississauga: Jones and Bartlett Publishers, 2004.

CUNHA, Igor. **Otimização de ciclo de turbina a vapor com três extrações utilizando o programa ipsepro.** 2016. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Mecânica) - Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2016.

DINÇER, Ibrahim; ROSEN, Marc A. Rosen.; AHMADI Pouria. **Optimization of Energy Systems.** Nova Jersey: John Wiley & Sons, 2017

DINÇER, Ibrahim; KANOGLU, Mehmet. **Refrigeration systems and applications.** 2. ed. Singapura: Markono Print Media Pte Ltd, 2010.

DOSSAT, R. J. **Principles of Refrigeration.** 4. Hemus, 2007.

Empresa de Pesquisa Energética (Brasil). **Balço Energético Nacional (BEN) 2020: Ano base 2019.** Rio de Janeiro: EPE, 2020. Disponível em < <https://ben.epe.gov.br> >. Acesso em 22 de outubro de 2020.

GILAT, Amos. **Métodos Numéricos para Engenheiros e Cientistas.** 9 ed. Bookman, 2008.

GHADDAR, N.K.; SHIHAB, M.; BDEIR, F. **Modeling and simulation of solar absorption system performance in Beirut**. Renewable Energy, 11 jun. 1996. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0960148196000390>>. Acesso em: 14 dez. 2020.

GUSMÃO, G. **Afinal quem inventou o ar-condicionado?** São Paulo: 2018. Disponível em: <<https://exame.com/tecnologia/afinal-quem-inventou-o-ar-condicionado/>>. Acesso em 12 de novembro de 2020.

JALURIA, Y. **Design and Optmization of Thermal Systems**, 2. ed. CRC Press, 2007.

LEE, Sangick. **Multi-parameter optimization of cold energy recovery in cascade Rankine cycle for LNG regasification using genetic algorithm**. Energy, Busan, Republic of Korea, 2016.

MACAGNAN, Mario H. **Programa de Pós-Graduação em Engenharia Mecânica: Disciplina: 100644 – Refrigeração**. Universidade do Vale do Rio dos Sinos (Unisinos). São Leopoldo, 2018. Disponível em: < <http://www.professor.unisinos.br/mhmac/Refrigeracao/> >. Acesso em 25 de outubro de 2020.

MACEDO, Thiago de Oliveira. **Metodologia de melhoramento de sistemas energéticos baseada em uma abordagem exergo-econômica-ambiental avançada**. Tese de qualificação de doutorado - Programa de Pós-graduação em Engenharia Mecânica, COPPE, da Universidade Federal do Rio de Janeiro, 2017.

MAUDONET, Victor. **Otimização termodinâmica e análise exérgica de sistemas térmicos operando em ciclos simples e com cogeração utilizando o simulador de processos ipsepro integrado ao MatLab**. 2017. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Mecânica) - Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2017.

MICROSOFT – **Component Object Model (COM)**. 2020. Disponível em: <<https://docs.microsoft.com/en-us/windows/win32/com/component-object-model--com--portal>>. Acesso em 06 de dezembro 2020.

MORAN, Michael J.; SHAPIRO, Howard N. **Princípios da Termodinâmica**, 6 ed, Rio de Janeiro: LTC, 2012.

MOSAFFA, A.H.; L. GAROUSI , Farshi; FERREIRA, C.A. Infante; ROSEN, M.A. **Exergoeconomic and environmental analyses of CO₂/NH₃ cascade refrigeration systems equipped with different types of flash tank intercoolers**. *Energy Conversion and Management*, 2016.

PALM, William J. **Introdução ao MatLab para engenheiros**. 3 ed., Porto Alegre: McGraw-Hill, 2013.

RODRIGUES, Rannier C. **Otimização termodinâmica de uma central de cogeração de energia utilizando o simulador de processos IPSEpro**. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Mecânica) - Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2013.

SADEGHI, Mohsen; MAHMOUDI, S.M.S.; SARAY, R. Khoshbakhti. *Energy Conversion and Management*. **Exergoeconomic analysis and multi-objective optimization of an ejector refrigeration cycle powered by an internal combustion (HCCI) engine**, Tabriz, Iran, ed. 9, 2015.

SARAMAGO, S. P., STEFFEN, V., **Introdução às técnicas de otimização em engenharia**, UFU, Uberlândia – MG.

SIM TECH SIMULATION TECHNOLOGY. **“IPSEpro Process Simulator - Refrigeration Process Library”**, *Library Version 4.0, Manual Version 4.0.00*. Graz, Áustria 2002.

SIM TECH SIMULATION TECHNOLOGY. **PSE: Process Simulation Environment**. Disponível em: <<http://www.simtechnology.com/CMS/index.php/ipsepro/program-modules/pse>>. Acesso em: 06 de dezembro de 2020.

SOLA, A. V. H; MOTA, C. M. M. **Melhoria da eficiência energética e m sistemas motrizes industriais**. *Production*.V.25, n. 2, pp. 498-509, 2015.

STOECKER, W.F.; JABARDO, J. M. S. **Refrigeração Industrial: 2ª edição**. São Paulo: Editora Edgard Blucher Ltda, 2002.

TAHA, Hamdy A., “**Pesquisa Operacional**”, 8ed, São Paulo: Pearson Prentice Hall, 2008.

TAVARES, L. V.; CORREIA, F. N. **Optimização Linear e Não Linear – Conceitos, Métodos e Algoritmos**. Fundação Calouste Gulbenkian, 2º edição. Lisboa, 1999.

THRELKELD, James L.; RAMSEY, James W.; KUEHN, Thomas H. **Thermal Environmental Engineering**, 3 ed., Nova Jersey: Prentice-Hall, Inc., 1998.

VARGAS, P.A., “**Sistemas Classificadores para Redução de Perdas em Redes de Distribuição de Energia Elétrica**”. Dissertação de Mestrado, Unicamp. Campinas, SP, 2000.

VIEIRA, Leonardo., **Modelo exergoeconômico iterativo para melhoria de sistemas térmicos complexos integrado a um simulador de processos profissional**. Tese de doutorado - Programa de Pós-graduação em Engenharia Mecânica, COPPE, da Universidade Federal do Rio de Janeiro, 2017.

WANG, Jiangfeng; DAI, Yiping; GAO, Lin. Applied Energy. **Parametric analysis and optimization for a combined power and refrigeration cycle**, PR China, ed. 85, p. 1071-1085, 18 abr. 2008.

WU, Zhixian; FENG, Huijun; CHEN, Ligen; GE, Yanlin., **Performance Optimization of a Condenser in Ocean Thermal Energy Conversion (OTEC) System Based on Constructal Theory and a Multi-Objective Genetic Algorithm**. Entropy, Wuhan, China, 9 jun. 2020.

ÇENGEL Yunus A; BOLES, Michael A. **Termodinâmica**, 7 ed., São Paulo: AMGH, 2011.

APÊNDICE A - Memória de cálculo da simulação (gerada pelo IPSEpro)

```

=====
||
||      PSStat - Static Solver
||      Version 6.0 U, Release 11, Build 1375, Jul 25 2014 11:54:54
||
=====

Loading system ...

System has been loaded successfully.
System analysis ...
=====
System model contains 94 variables
9 variables set to fixed values
System to be solved contains 85 variables and 85 equations
85 groups found. Largest group contains 1 variables.
=====
System analysis finished.

Starting system solution ....
=====
-----
Solving group 1, a group with 1 elements

Group contains the following variables:
r134a.FluidID

Group contains the following equations:
r134a.equR134a

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 2, a group with 1 elements

Group contains the following variables:
R_Stream009.p

Group contains the following equations:
R_Connector001.f_p

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 3, a group with 1 elements

Group contains the following variables:
R_Stream001.p

Group contains the following equations:
R_Heat_sink001.f_p

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 4, a group with 1 elements

Group contains the following variables:
R_Stream001.h

Group contains the following equations:
R_Xprescription001.f4

```

```

Initializing ...
errf = 0.00436368 (>0.000103142)

1. iteration step
Group converged with errx= 1.53589e-005 (<=0.000103142)
-----
Solving group 5, a group with 1 elements

Group contains the following variables:
R_Stream003.h

Group contains the following equations:
R_Xprescription001.f3

Initializing ...
errf = 0.00436368 (>0.000103142)

1. iteration step
Group converged with errx= 1.53589e-005 (<=0.000103142)
-----
Solving group 6, a group with 1 elements

Group contains the following variables:
R_Stream005.h

Group contains the following equations:
R_Valve001.f_h

Initializing ...
errf = 0.00436368 (>0.000103142)

1. iteration step
Group converged with errx= 1.53589e-005 (<=0.000103142)
-----
Solving group 7, a group with 1 elements

Group contains the following variables:
R_Stream009.h

Group contains the following equations:
R_Connector001.f_h

Initializing ...
errf = 0.00436368 (>0.000103142)

1. iteration step
Group converged with errx= 1.53589e-005 (<=0.000103142)
-----
Solving group 8, a group with 1 elements

Group contains the following variables:
R_Stream009.t

Group contains the following equations:
R_Stream009.ft

Initializing ...
errf = 22.5408 (>0.000103142)

```

```

1. iteration step
errx = 1.80662 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 9, a group with 1 elements

Group contains the following variables:
R_Stream009.s

Group contains the following equations:
R_Stream009.fs

Initializing ...
errf = 0.0255193 (>0.000103142)

1. iteration step
errx = 0.0197002 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 10, a group with 1 elements

Group contains the following variables:
R_Stream009.v

Group contains the following equations:
R_Stream009.fv

Initializing ...
errf = 0.0308996 (>0.000103142)

1. iteration step
errx = 1.84537 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 11, a group with 1 elements

Group contains the following variables:
R_Stream009.mass

Group contains the following equations:
R_Separator001.f4

Initializing ...
errf = 0.0245499 (>0.000103142)

1. iteration step
errx = 0.223181 (>0.000103142)
Group converged with errf= 1.38778e-017 (<=0.000103142)
-----
Solving group 12, a group with 1 elements

Group contains the following variables:
R_Stream007.p

Group contains the following equations:
R_Separator001.f2

Initializing ...
errf = 2.499 (>0.000103142)

1. iteration step

```

```

errx = 0.555333 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 13, a group with 1 elements

Group contains the following variables:
R_Stream012.p

Group contains the following equations:
R_Mixer001.f4

Initializing ...
errf = 2.499 (>0.000103142)

1. iteration step
errx = 0.555333 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 14, a group with 1 elements

Group contains the following variables:
R_Stream007.mass

Group contains the following equations:
R_Separator001.f1

Initializing ...
errf = 0.0381029 (>0.000103142)

1. iteration step
errx = 0.627362 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 15, a group with 1 elements

Group contains the following variables:
R_Stream008.mass

Group contains the following equations:
R_Valve002.f_mass

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 16, a group with 1 elements

Group contains the following variables:
R_Stream010.mass

Group contains the following equations:
R_Heat_source001.f_mass

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 17, a group with 1 elements

Group contains the following variables:
R_Stream011.mass

Group contains the following equations:
R_Xprescription002.f1

```

```

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 18, a group with 1 elements

Group contains the following variables:
R_Stream004.mass

Group contains the following equations:
R_Compressor002.f1

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 19, a group with 1 elements

Group contains the following variables:
R_Stream012.mass

Group contains the following equations:
R_Mixer001.f1

Initializing ...
errf = 0.0381049 (>0.000103142)

1. iteration step
errx = 0.223181 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 20, a group with 1 elements

Group contains the following variables:
R_Stream006.p

Group contains the following equations:
R_Separator001.f3

Initializing ...
errf = 2.499 (>0.000103142)

1. iteration step
errx = 0.555333 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 21, a group with 1 elements

Group contains the following variables:
R_Stream006.h

Group contains the following equations:
R_Separator001.f5

Initializing ...
errf = 30.3767 (>0.000103142)

1. iteration step
errx = 0.139993 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 22, a group with 1 elements

Group contains the following variables:

```

R_Stream007.h

Group contains the following equations:
R_Separator001.f6

Initializing ...
errf = 0.793334 (>0.000103142)

1. iteration step
errx = 0.0321973 (>0.000103142)
Group converged with errf= 7.10543e-015 (<=0.000103142)

Solving group 23, a group with 1 elements

Group contains the following variables:
R_Stream010.p

Group contains the following equations:
R_Heat_source001.f_p

Initializing ...
Group converged with errf= 0 (<=0.000103142)

Solving group 24, a group with 1 elements

Group contains the following variables:
R_Stream010.h

Group contains the following equations:
R_Xprescription002.f4

Initializing ...
errf = 0.00109456 (>0.000103142)

1. iteration step
Group converged with errx= 2.78785e-006 (<=0.000103142)

Solving group 25, a group with 1 elements

Group contains the following variables:
R_Stream011.h

Group contains the following equations:
R_Xprescription002.f3

Initializing ...
errf = 0.00109456 (>0.000103142)

1. iteration step
Group converged with errx= 2.78785e-006 (<=0.000103142)

Solving group 26, a group with 1 elements

Group contains the following variables:
R_Stream011.p

Group contains the following equations:
R_Xprescription002.f2

Initializing ...
Group converged with errf= 0 (<=0.000103142)

```

Solving group 27, a group with 1 elements

Group contains the following variables:
R_Stream011.s

Group contains the following equations:
R_Stream011.fs

Initializing ...
Group converged with errf= 4.54047e-006 (<=0.000103142)
-----
Solving group 28, a group with 1 elements

Group contains the following variables:
R_Stream004.p

Group contains the following equations:
R_Mixer001.f3

Initializing ...
errf = 2.499 (>0.000103142)

1. iteration step
errx = 0.555333 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 29, a group with 1 elements

Group contains the following variables:
R_Stream004.h

Group contains the following equations:
R_Compressor002.f2

Initializing ...
errf = 0.0581037 (>0.000103142)

1. iteration step
errx = 0.0468471 (>0.000103142)
Group converged with errf= 7.9401e-005 (<=0.000103142)
-----
Solving group 30, a group with 1 elements

Group contains the following variables:
R_Stream012.h

Group contains the following equations:
R_Mixer001.f2

Initializing ...
errf = 2.91607 (>0.000103142)

1. iteration step
errx = 0.0416869 (>0.000103142)
Group converged with errf= 1.42109e-014 (<=0.000103142)
-----
Solving group 31, a group with 1 elements

Group contains the following variables:
R_Stream012.t

Group contains the following equations:

```

R_Stream012.ft

Initializing ...
 errf = 26.7584 (>0.000103142)

1. iteration step
 errx = 1.60284 (>0.000103142)
 Group converged with errf= 0 (<=0.000103142)

 Solving group 32, a group with 1 elements

Group contains the following variables:
 R_Stream012.s

Group contains the following equations:
 R_Stream012.fs

Initializing ...
 errf = 0.00169576 (>0.000103142)

1. iteration step
 errx = 0.000977386 (>0.000103142)
 Group converged with errf= 0 (<=0.000103142)

 Solving group 33, a group with 1 elements

Group contains the following variables:
 R_Stream012.v

Group contains the following equations:
 R_Stream012.fv

Initializing ...
 errf = 0.0531392 (>0.000103142)

1. iteration step
 errx = 1.13813 (>0.000103142)
 Group converged with errf= 0 (<=0.000103142)

 Solving group 34, a group with 1 elements

Group contains the following variables:
 R_Stream011.t

Group contains the following equations:
 R_Stream011.ft

Initializing ...
 errf = 0.000274019 (>0.000103142)

1. iteration step
 Group converged with errx= 2.71944e-005 (<=0.000103142)

 Solving group 35, a group with 1 elements

Group contains the following variables:
 R_Stream011.v

Group contains the following equations:
 R_Stream011.fv

```

Initializing ...
Group converged with errf= 4.30402e-007 (<=0.000103142)
-----
Solving group 36, a group with 1 elements

Group contains the following variables:
R_Stream010.t

Group contains the following equations:
R_Stream010.ft

Initializing ...
errf = 0.000274019 (>0.000103142)

1. iteration step
Group converged with errx= 2.71944e-005 (<=0.000103142)
-----
Solving group 37, a group with 1 elements

Group contains the following variables:
R_Stream010.s

Group contains the following equations:
R_Stream010.fs

Initializing ...
Group converged with errf= 4.54047e-006 (<=0.000103142)
-----
Solving group 38, a group with 1 elements

Group contains the following variables:
R_Stream010.v

Group contains the following equations:
R_Stream010.fv

Initializing ...
Group converged with errf= 4.30402e-007 (<=0.000103142)
-----
Solving group 39, a group with 1 elements

Group contains the following variables:
R_Stream003.p

Group contains the following equations:
R_Xprescription001.f2

Initializing ...
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 40, a group with 1 elements

Group contains the following variables:
R_Stream003.t

Group contains the following equations:
R_Stream003.ft

Initializing ...
Group converged with errf= 5.82638e-005 (<=0.000103142)
-----
Solving group 41, a group with 1 elements

Group contains the following variables:

```

R_Stream003.s

Group contains the following equations:
R_Stream003.fs

Initializing ...

Group converged with errf= 4.3696e-005 (<=0.000103142)

Solving group 42, a group with 1 elements

Group contains the following variables:
R_Stream003.v

Group contains the following equations:
R_Stream003.fv

Initializing ...

Group converged with errf= 1.81734e-009 (<=0.000103142)

Solving group 43, a group with 1 elements

Group contains the following variables:
R_Stream002.mass

Group contains the following equations:
R_Compressor001.fl

Initializing ...

errf = 0.0381049 (>0.000103142)

1. iteration step

errx = 0.223181 (>0.000103142)

Group converged with errf= 0 (<=0.000103142)

Solving group 44, a group with 1 elements

Group contains the following variables:
R_Stream001.mass

Group contains the following equations:
R_Heat_sink001.f_mass

Initializing ...

errf = 0.0381049 (>0.000103142)

1. iteration step

errx = 0.223181 (>0.000103142)

Group converged with errf= 0 (<=0.000103142)

Solving group 45, a group with 1 elements

Group contains the following variables:
R_Stream003.mass

Group contains the following equations:
R_Xprescription001.fl

Initializing ...

errf = 0.0381049 (>0.000103142)

1. iteration step

errx = 0.223181 (>0.000103142)

```

Group converged with errf= 0 (<=0.000103142)
-----
Solving group 46, a group with 1 elements

Group contains the following variables:
R_Stream001.t

Group contains the following equations:
R_Stream001.ft

Initializing ...
Group converged with errf= 5.82638e-005 (<=0.000103142)
-----
Solving group 47, a group with 1 elements

Group contains the following variables:
R_Stream001.s

Group contains the following equations:
R_Stream001.fs

Initializing ...
Group converged with errf= 4.3696e-005 (<=0.000103142)
-----
Solving group 48, a group with 1 elements

Group contains the following variables:
R_Stream001.v

Group contains the following equations:
R_Stream001.fv

Initializing ...
Group converged with errf= 1.81734e-009 (<=0.000103142)
-----
Solving group 49, a group with 1 elements

Group contains the following variables:
shaft002.power

Group contains the following equations:
R_Compressor002.f3b

Initializing ...
errf = 2.1231 (>0.000103142)

1. iteration step
errx = 0.999595 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 50, a group with 1 elements

Group contains the following variables:
R_Stream008.h

Group contains the following equations:
R_Valve002.f_h

Initializing ...
errf = 30.3767 (>0.000103142)

1. iteration step
errx = 0.139993 (>0.000103142)

```

```

Group converged with errf= 0 (<=0.000103142)
-----
Solving group 51, a group with 1 elements

Group contains the following variables:
R_Stream008.t

Group contains the following equations:
R_Stream008.ft

Initializing ...
errf = 0.000274003 (>0.000103142)

1. iteration step
Group converged with errx= 2.71929e-005 (<=0.000103142)
-----
Solving group 52, a group with 1 elements

Group contains the following variables:
R_Stream008.s

Group contains the following equations:
R_Stream008.fs

Initializing ...
errf = 0.115459 (>0.000103142)

1. iteration step
errx = 0.108332 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 53, a group with 1 elements

Group contains the following variables:
R_Stream008.v

Group contains the following equations:
R_Stream008.fv

Initializing ...
errf = 0.0146137 (>0.000103142)

1. iteration step
errx = 0.950479 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 54, a group with 1 elements

Group contains the following variables:
R_Stream007.t

Group contains the following equations:
R_Stream007.ft

Initializing ...
errf = 22.5408 (>0.000103142)

1. iteration step
errx = 1.80662 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----

```

Solving group 55, a group with 1 elements

Group contains the following variables:
R_Stream007.s

Group contains the following equations:
R_Stream007.fs

Initializing ...
errf = 0.0123665 (>0.000103142)

1. iteration step
errx = 0.00718553 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 56, a group with 1 elements

Group contains the following variables:
R_Stream007.v

Group contains the following equations:
R_Stream007.fv

Initializing ...
errf = 0.0542043 (>0.000103142)

1. iteration step
errx = 1.18805 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 57, a group with 1 elements

Group contains the following variables:
R_Stream006.t

Group contains the following equations:
R_Stream006.ft

Initializing ...
errf = 22.5408 (>0.000103142)

1. iteration step
errx = 1.80662 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 58, a group with 1 elements

Group contains the following variables:
R_Stream006.s

Group contains the following equations:
R_Stream006.fs

Initializing ...
errf = 0.110032 (>0.000103142)

1. iteration step
errx = 0.103768 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 59, a group with 1 elements

Group contains the following variables:
R_Stream006.v

Group contains the following equations:
R_Stream006.fv

Initializing ...
Group converged with errf= 4.37172e-010 (<=0.000103142)

Solving group 60, a group with 1 elements

Group contains the following variables:
R_Stream005.t

Group contains the following equations:
R_Stream005.ft

Initializing ...
errf = 22.5408 (>0.000103142)

1. iteration step
errx = 1.80662 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 61, a group with 1 elements

Group contains the following variables:
R_Stream005.s

Group contains the following equations:
R_Stream005.fs

Initializing ...
errf = 0.0255193 (>0.000103142)

1. iteration step
errx = 0.0197002 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 62, a group with 1 elements

Group contains the following variables:
R_Stream005.v

Group contains the following equations:
R_Stream005.fv

Initializing ...
errf = 0.0308996 (>0.000103142)

1. iteration step
errx = 1.84537 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 63, a group with 1 elements

Group contains the following variables:
R_Stream005.mass

Group contains the following equations:
R_Valve001.f_mass

```

Initializing ...
errf = 0.0381049 (>0.000103142)

1. iteration step
errx = 0.223181 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 64, a group with 1 elements

Group contains the following variables:
R_Stream002.h

Group contains the following equations:
R_Compressor001.f2

Initializing ...
errf = 0.00533773 (>0.000103142)

1. iteration step
errx = 0.00479542 (>0.000103142)
Group converged with errf= 1.15842e-005 (<=0.000103142)
-----
Solving group 65, a group with 1 elements

Group contains the following variables:
shaft001.power

Group contains the following equations:
R_Compressor001.f3b

Initializing ...
errf = 5.19992 (>0.000103142)

1. iteration step
errx = 0.975033 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 66, a group with 1 elements

Group contains the following variables:
R_Stream004.t

Group contains the following equations:
R_Stream004.ft

Initializing ...
errf = 29.1039 (>0.000103142)

1. iteration step
errx = 1.52849 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 67, a group with 1 elements

Group contains the following variables:
R_Stream004.s

Group contains the following equations:
R_Stream004.fs

```

```

Initializing ...
errf = 0.00921775 (>0.000103142)

1. iteration step
errx = 0.0052896 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 68, a group with 1 elements

Group contains the following variables:
R_Stream004.v

Group contains the following equations:
R_Stream004.fv

Initializing ...
errf = 0.0525596 (>0.000103142)

1. iteration step
errx = 1.11191 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 69, a group with 1 elements

Group contains the following variables:
R_Stream002.t

Group contains the following equations:
R_Stream002.ft

Initializing ...
errf = 1.75483 (>0.000103142)

1. iteration step
errx = 0.0252029 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 70, a group with 1 elements

Group contains the following variables:
R_Stream002.s

Group contains the following equations:
R_Stream002.fs

Initializing ...
errf = 0.00619113 (>0.000103142)

1. iteration step
errx = 0.00354223 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 71, a group with 1 elements

Group contains the following variables:
R_Stream002.v

Group contains the following equations:
R_Stream002.fv

Initializing ...

```

errf = 0.00015931 (>0.000103142)

1. iteration step
 errx = 0.0118944 (>0.000103142)
 Group converged with errf= 0 (<=0.000103142)

 Solving group 72, a group with 1 elements

Group contains the following variables:
 R_Heat_source001.delta_t

Group contains the following equations:
 R_Heat_source001.f_t

Initializing ...
 Group converged with errf= 1.52172e-008 (<=0.000103142)

 Solving group 73, a group with 1 elements

Group contains the following variables:
 R_Heat_source001.q_trans

Group contains the following equations:
 R_Heat_source001.f_energy

Initializing ...
 errf = 3.34146 (>0.000103142)

1. iteration step
 errx = 0.172957 (>0.000103142)
 Group converged with errf= 7.10543e-015 (<=0.000103142)

 Solving group 74, a group with 1 elements

Group contains the following variables:
 motor002.power

Group contains the following equations:
 motor002.flb

Initializing ...
 errf = 2.1231 (>0.000103142)

1. iteration step
 errx = 0.999595 (>0.000103142)
 Group converged with errf= 0 (<=0.000103142)

 Solving group 75, a group with 1 elements

Group contains the following variables:
 R_Compressor002.pressure_ratio

Group contains the following equations:
 R_Compressor002.fPressureRatio

Initializing ...
 errf = 2.499 (>0.000103142)

1. iteration step
 errx = 0.555333 (>0.000103142)
 Group converged with errf= 0 (<=0.000103142)

```

-----
Solving group 76, a group with 1 elements

Group contains the following variables:
R_Compressor002.V

Group contains the following equations:
R_Compressor002.f_V

Initializing ...
Group converged with errf= 4.53e-006 (<=0.000103142)
-----
Solving group 77, a group with 1 elements

Group contains the following variables:
R_Valve002.pressure_drop

Group contains the following equations:
R_Valve002.f_p_rel

Initializing ...
errf = 2.49775 (>0.000103142)

1. iteration step
errx = 0.9991 (>0.000103142)
Group converged with errf= 4.44089e-016 (<=0.000103142)
-----
Solving group 78, a group with 1 elements

Group contains the following variables:
R_Valve002.delta_p

Group contains the following equations:
R_Valve002.f_p_abs

Initializing ...
errf = 2.499 (>0.000103142)

1. iteration step
errx = 0.9996 (>0.000103142)
Group converged with errf= 4.44089e-016 (<=0.000103142)
-----
Solving group 79, a group with 1 elements

Group contains the following variables:
R_Valve001.pressure_drop

Group contains the following equations:
R_Valve001.f_p_rel

Initializing ...
errf = 2.49904 (>0.000103142)

1. iteration step
errx = 0.217308 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)
-----
Solving group 80, a group with 1 elements

Group contains the following variables:
R_Valve001.delta_p

```

Group contains the following equations:
R_Valve001.f_p_abs

Initializing ...
errf = 2.499 (>0.000103142)

1. iteration step
errx = 0.217304 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 81, a group with 1 elements

Group contains the following variables:
motor001.power

Group contains the following equations:
motor001.flb

Initializing ...
errf = 5.19992 (>0.000103142)

1. iteration step
errx = 0.975033 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 82, a group with 1 elements

Group contains the following variables:
R_Compressor001.pressure_ratio

Group contains the following equations:
R_Compressor001.fPressureRatio

Initializing ...
errf = 19.982 (>0.000103142)

1. iteration step
errx = 1.24887 (>0.000103142)
Group converged with errf= 3.55271e-015 (<=0.000103142)

Solving group 83, a group with 1 elements

Group contains the following variables:
R_Compressor001.V

Group contains the following equations:
R_Compressor001.f_V

Initializing ...
errf = 0.0128764 (>0.000103142)

1. iteration step
errx = 1.61529 (>0.000103142)
Group converged with errf= 0 (<=0.000103142)

Solving group 84, a group with 1 elements

Group contains the following variables:
R_Heat_sink001.delta_t

Group contains the following equations:

R_Heat_sink001.f_t

Initializing ...

errf = 1.75483 (>0.000103142)

1. iteration step

errx = 0.149702 (>0.000103142)

Group converged with errf= 7.10543e-015 (<=0.000103142)

Solving group 85, a group with 1 elements

Group contains the following variables:

R_Heat_sink001.q_trans

Group contains the following equations:

R_Heat_sink001.f_energy

Initializing ...

errf = 6.41742 (>0.000103142)

1. iteration step

errx = 0.239665 (>0.000103142)

Group converged with errf= 0 (<=0.000103142)

=====
System solution finished.

=====
Results:

x[1]	<r134a.FluidID>	=	23	converged
x[2]	<R_Stream009.p>	=	4.5	converged
x[3]	<R_Stream009.t>	=	12.477	converged
x[4]	<R_Stream009.h>	=	284.11	converged
x[5]	<R_Stream009.s>	=	1.2954	converged
x[6]	<R_Stream009.v>	=	0.016744	converged
x[7]	<R_Stream009.mass>	=	0.17074	converged
x[8]	<R_Stream012.p>	=	4.5	converged
x[9]	<R_Stream012.t>	=	16.694	converged
x[10]	<R_Stream012.h>	=	409.71	converged
x[11]	<R_Stream012.s>	=	1.735	converged
x[12]	<R_Stream012.v>	=	0.04669	converged
x[13]	<R_Stream012.mass>	=	0.17074	converged
x[14]	<R_Stream011.p>	=	2	converged
x[15]	<R_Stream011.t>	=	-10.076	converged
x[16]	<R_Stream011.h>	=	392.62	converged
x[17]	<R_Stream011.s>	=	1.7334	converged
x[18]	<R_Stream011.v>	=	0.099877	converged
x[19]	<R_Stream011.mass>	=	0.11	converged
x[20]	<R_Stream010.p>	=	2	converged
x[21]	<R_Stream010.t>	=	-10.076	converged
x[22]	<R_Stream010.h>	=	392.62	converged
x[23]	<R_Stream010.s>	=	1.7334	converged
x[24]	<R_Stream010.v>	=	0.099877	converged
x[25]	<R_Stream010.mass>	=	0.11	converged
x[26]	<R_Stream003.p>	=	16	converged
x[27]	<R_Stream003.t>	=	57.906	converged
x[28]	<R_Stream003.h>	=	284.11	converged
x[29]	<R_Stream003.s>	=	1.2748	converged
x[30]	<R_Stream003.v>	=	0.0009401	converged
x[31]	<R_Stream003.mass>	=	0.17074	converged
x[32]	<R_Stream001.p>	=	16	converged

```

x[33] <R_Stream001.t> = 57.906 converged
x[34] <R_Stream001.h> = 284.11 converged
x[35] <R_Stream001.s> = 1.2748 converged
x[36] <R_Stream001.v> = 0.0009401 converged
x[37] <R_Stream001.mass> = 0.17074 converged
x[38] <shaft002.power> = 2.124 converged
x[39] <R_Stream008.p> = 2 set
x[40] <R_Stream008.t> = -10.076 converged
x[41] <R_Stream008.h> = 216.99 converged
x[42] <R_Stream008.s> = 1.0658 converged
x[43] <R_Stream008.v> = 0.015375 converged
x[44] <R_Stream008.mass> = 0.11 converged
x[45] <R_Stream007.p> = 4.5 converged
x[46] <R_Stream007.t> = 12.477 converged
x[47] <R_Stream007.h> = 405.69 converged
x[48] <R_Stream007.s> = 1.721 converged
x[49] <R_Stream007.v> = 0.045625 converged
x[50] <R_Stream007.mass> = 0.060735 converged
x[51] <R_Stream006.p> = 4.5 converged
x[52] <R_Stream006.t> = 12.477 converged
x[53] <R_Stream006.h> = 216.99 converged
x[54] <R_Stream006.s> = 1.0604 converged
x[55] <R_Stream006.v> = 0.00079852 converged
x[56] <R_Stream006.mass> = 0.11 set
x[57] <R_Stream005.p> = 4.5 set
x[58] <R_Stream005.t> = 12.477 converged
x[59] <R_Stream005.h> = 284.11 converged
x[60] <R_Stream005.s> = 1.2954 converged
x[61] <R_Stream005.v> = 0.016744 converged
x[62] <R_Stream005.mass> = 0.17074 converged
x[63] <shaft001.power> = 5.3331 converged
x[64] <R_Stream004.p> = 4.5 converged
x[65] <R_Stream004.t> = 19.041 converged
x[66] <R_Stream004.h> = 411.93 converged
x[67] <R_Stream004.s> = 1.7426 converged
x[68] <R_Stream004.v> = 0.047269 converged
x[69] <R_Stream004.mass> = 0.11 converged
x[70] <R_Stream002.p> = 16 set
x[71] <R_Stream002.t> = 69.628 converged
x[72] <R_Stream002.h> = 440.95 converged
x[73] <R_Stream002.s> = 1.7478 converged
x[74] <R_Stream002.v> = 0.013394 converged
x[75] <R_Stream002.mass> = 0.17074 converged
x[76] <R_Separator001.y_separation> = 1 set
x[77] <R_Mixer001.delta_p_1> = 0 set
x[78] <R_Mixer001.delta_p_2> = 0 set
x[79] <R_Xprescription002.x> = 1 set
x[80] <R_Heat_source001.delta_t> = -4.3477e-016 converged
x[81] <R_Heat_source001.q_trans> = 19.32 converged
x[82] <R_Xprescription001.x> = 0 set
x[83] <motor002.power> = 2.124 converged
x[84] <R_Compressor002.pressure_ratio> = 2.25 converged
x[85] <R_Compressor002.V> = 0.010991 converged
x[86] <R_Valve002.pressure_drop> = 0.55556 converged
x[87] <R_Valve002.delta_p> = 2.5 converged
x[88] <R_Valve001.pressure_drop> = 0.71875 converged
x[89] <R_Valve001.delta_p> = 11.5 converged
x[90] <motor001.power> = 5.3331 converged
x[91] <R_Compressor001.pressure_ratio> = 3.5556 converged
x[92] <R_Compressor001.V> = 0.0079716 converged
x[93] <R_Heat_sink001.delta_t> = 11.722 converged
x[94] <R_Heat_sink001.q_trans> = 26.777 converged

```

=====
Solution

Start: Oct 26 2020, 21:12:17,689
End: Oct 26 2020, 21:12:17.965
Actual Time: 0.276000 seconds
CPU Time: 0.171000 seconds
CPU Percentage: 8%
0 errors (0 recovered errors)
0 warnings (0 recovered warnings)

APÊNDICE B – Código computacional do MatLab para cálculo do COP através de
integração com o IPSEpro

```

% o comando actxserver cria um servidor automático COM
% em outras palavras cria a COM interface entre o MATLAB e o
módulo PSE
app = actxserver('PSE.Application')

% Diretório armazenado em ProjPath
ProjPath = 'C:\Users\Mariana Machado\Documents\MARIANA\Eng.
Mecânica\PROJETO FINAL\11.11.2019 - Felipe e Mariana - Set do
Evaporador.pro'

% invoca para a COM interface criada o método "openProject" da
classe "Application", no diretório ProjPath
% em outras palavras abre o arquivo .pro criado no PSE
proj = invoke(app, 'openProject', ProjPath)

% abaixo seguem exemplos para a definição das propriedades do
IPSEpro

% set vazão mássica do evaporador = 0.11
R_Stream006 = invoke(proj, 'findObject', 'R_Stream006')

R_Stream006_mass = invoke(R_Stream006, 'findItem', 0, 'mass')

invoke(R_Stream006_mass, 'inputValue', 0.11)

% set p antes do evaporador = 2.0
R_Stream008 = invoke(proj, 'findObject', 'R_Stream008')

R_Stream008_p = invoke(R_Stream008, 'findItem', 0, 'p')

invoke(R_Stream008_p, 'inputValue', 2.0)

% set delta_p do evaporador = 0
R_Heat_source001 =
invoke(proj, 'findObject', 'R_Heat_source001')

R_Heat_source001_delta_p =
invoke(R_Heat_source001, 'findItem', 0, 'delta_p')

invoke(R_Heat_source001_delta_p, 'inputValue', 0)

% set título após evaporador = 1
R_Xprescription002 =
invoke(proj, 'findObject', 'R_Xprescription002')

R_Xprescription002_x =
invoke(R_Xprescription002, 'findItem', 0, 'x')

```

```

invoke(R_Xprescription002_x, 'inputValue', 1)

% set eficiência isentrópica do comp_alta pressao = 0.86
R_Compressor001 = invoke(proj, 'findObject', 'R_Compressor001')

R_Compressor001_eta_s =
invoke(R_Compressor001, 'findItem', 0, 'eta_s')

invoke(R_Compressor001_eta_s, 'inputValue', 0.86)

% set eficiência isentrópica do comp_baixa pressao = 0.86
R_Compressor002 = invoke(proj, 'findObject', 'R_Compressor002')

R_Compressor002_eta_s =
invoke(R_Compressor002, 'findItem', 0, 'eta_s')

invoke(R_Compressor002_eta_s, 'inputValue', 0.86)

% set eta_m motor baixa pressao = 1
motor002 = invoke(proj, 'findObject', 'motor002')

motor002_eta_m = invoke(motor002, 'findItem', 0, 'eta_m')

invoke(motor002_eta_m, 'inputValue', 1)

% set eta_el motor alta pressao = 1
motor001 = invoke(proj, 'findObject', 'motor001')

motor001_eta_el = invoke(motor001, 'findItem', 0, 'eta_el')

invoke(motor001_eta_el, 'inputValue', 1)

% set eta_m motor alta pressao = 1
motor001 = invoke(proj, 'findObject', 'motor001')

motor001_eta_m = invoke(motor001, 'findItem', 0, 'eta_m')

invoke(motor001_eta_m, 'inputValue', 1)

% set mixer delta_p_1
R_Mixer001 = invoke(proj, 'findObject', 'R_Mixer001')

R_Mixer001_delta_p_1 =
invoke(R_Mixer001, 'findItem', 0, 'delta_p_1')

invoke(R_Mixer001_delta_p_1, 'inputValue', 0)

% set mixer delta_p_2
R_Mixer001 = invoke(proj, 'findObject', 'R_Mixer001')

```

```

R_Mixer001_delta_p_2 =
invoke(R_Mixer001, 'findItem', 0, 'delta_p_2')

invoke(R_Mixer001_delta_p_2, 'inputValue', 0)

% set p antes do condensador = 16
R_Stream002 = invoke(proj, 'findObject', 'R_Stream002')

R_Stream002_p = invoke(R_Stream002, 'findItem', 0, 'p')

invoke(R_Stream002_p, 'inputValue', 16)

% set delta_p do condensador = 0
R_Heat_sink001 = invoke(proj, 'findObject', 'R_Heat_sink001')

R_Heat_sink001_delta_p =
invoke(R_Heat_sink001, 'findItem', 0, 'delta_p')

invoke(R_Heat_sink001_delta_p, 'inputValue', 0)

% set título após condensador = 0
R_Xprescription001 =
invoke(proj, 'findObject', 'R_Xprescription001')

R_Xprescription001_x =
invoke(R_Xprescription001, 'findItem', 0, 'x')

invoke(R_Xprescription001_x, 'inputValue', 0)

% set p antes do separador de vapor = 4.5
R_Stream005 = invoke(proj, 'findObject', 'R_Stream005')

R_Stream005_p = invoke(R_Stream005, 'findItem', 0, 'p')

invoke(R_Stream005_p, 'inputValue', 4.5)

    % obtenção do calor absorvido pelo evaporador
    R_Heat_source001 =
invoke(proj, 'findObject', 'R_Heat_source001')

    R_Heat_source001_q_trans =
invoke(R_Heat_source001, 'findItem', 0, 'q_trans')

    q_trans_evaporador =
invoke(R_Heat_source001_q_trans, 'resultValue')

    % obtenção do trabalho do motor que ativa o compressor de
    alta pressão
    motor001 = invoke(proj, 'findObject', 'motor001')

```

```
motor001_power = invoke(motor001,'findItem',0,'power')

power_compressor_alta =
invoke(motor001_power,'resultValue')

% obtenção do trabalho do motor que ativa o compressor de
baixa pressão
motor002 = invoke(proj,'findObject','motor002')

motor002_power = invoke(motor002,'findItem',0,'power')

power_compressor_baixa =
invoke(motor002_power,'resultValue')

% cálculo do COP
COP = q_trans_evaporador / ( power_compressor_alta +
power_compressor_baixa )
```

ANEXO A - Modelos utilizados nos objetos da simulação, desenvolvidos pelo IPSEpro

COMPRESSOR

Model equations

```

# mass balance
f1:  feed.mass = drain.mass;

f2:  feed.s = drain.Composition.fs(drain.p, feed.h+(drain.h - feed.h)*eta_s);

# both sides connected
ifl ref(shaft_in) && ref(shaft_out) then
    f3a:  (feed.h - drain.h)* feed.mass /eta_m + shaft_in.power - shaft_out.power = 0.0;
endifl

# left side shaft only
ifl ref(shaft_in) && !ref(shaft_out) then
    f3b:  (feed.h - drain.h)* feed.mass / eta_m + shaft_in.power = 0.0;
endifl

# right side shaft only
ifl !ref(shaft_in) && ref(shaft_out) then
    f3c:  (feed.h - drain.h) * feed.mass / eta_m - shaft_out.power = 0.0;
endifl

# pressure ratio
fPressureRatio:  feed.p * pressure_ratio = drain.p;

# test conditions
t1:  test((drain.p - feed.p) >= 0.0)    warning "outlet pressure lower than inlet pressure";
t2:  test ( eta_s >= 0.0)              error "isentropic efficiency < 0.0";
t3:  test ( eta_s <= 1.0)              error "isentropic efficiency >1.0";
t4:  test ( eta_m >= 0.0)              error "mechanical efficiency < 0.0";
t5:  test ( eta_m <= 1.0)              error "mechanical efficiency > 1.0";

```

Parameters

eta_m [----] mechanical efficiency
eta_s [----] isentropic efficiency

Variables

pressure_ratio [---] pressure ratio of the compressor

CONDENSADOR

Model equations

```

# mass balance
f_mass:    feed.mass = drain.mass;
f_z:    feed.z = drain.z;

# energy balance
f_energy:  feed.mass * feed.h - q_trans = drain.mass * drain.h;

# pressure drop
f_p:    feed.p - delta_p = drain.p;

# temperature loss
f_t:    feed.t - delta_t = drain.t;

# test conditions
t1:    test (q_trans >= 0.0) warning "q_trans negative";
t2:    test (delta_p >= 0.0) warning "delta_p negative";

```

Parameters

delta_p [bar] (pressure_diff) absolute pressure drop

Variables

delta_t [°C] (temperatur_diff) temperature difference between feed and drain mass flow

q_trans [kW] (power) transferred heat

FONTE

Model equations

```
f_mass:    mass = drain.mass;  
f_p:    p = drain.p;  
f_t:    t = drain.t;  
f_z:    z = drain.z;
```

Variables

mass	[kg/s]	(mass_flow)	mass flow
p	[bar]	(pressure)	pressure
t	[°C]	(temperature)	temperature
z	[kg/kg]	(mass_fraction)	concentration

SUMIDOURO:

Model equations

```
f_mass:    mass = feed.mass;  
f_p:      p = feed.p;  
f_t:      t = feed.t;  
f_z:      z = feed.z;
```

Variables

mass	[kg/s]	(mass_flow)	mass flow
p	[bar]	(pressure)	outlet pressure
t	[°C]	(temperature)	outlet temperature
z	[kg/kg]	(mass_fraction)	concentration

EVAPORADOR

Model equations

```

# mass balance
f_mass:    feed.mass = drain.mass;
f_z:    feed.z = drain.z;

# energy balance
f_energy:    feed.mass * feed.h + q_trans = drain.mass * drain.h;

# pressure drop
f_p:    feed.p - delta_p = drain.p;

# temperature increase
f_t:    feed.t + delta_t = drain.t;

# test conditions
t1:    test (q_trans >= 0.0) warning "q_trans negative";
t2:    test (delta_p >= 0.0) warning "delta_p negative";

```

Parameters

delta_p [bar] (pressure_diff) absolute pressure drop

Variables

delta_t [°C] (temperatur_diff) temperature difference between feed and drain mass flow

q_trans [kW] (power) transferred heat

R_XPrescription

Model equations

```
# mass balance
f1:  feed.mass = drain.mass;

# pressure
f2:  feed.p = drain.p;

# enthalpy
f3:  feed.h = drain.h;
f4:  feed.h - feed.Composition.f_h_pq( feed.p, 0.0) - x* (feed.Composition.f_h_pq( feed.p,1.0) - feed.Composition.f_h_pq( feed.p, 0.0)) = 0.0;

# tests
t_conn:  test ( feed.Composition.FluidID == drain.Composition.FluidID )    error "different fluid at feed and drain!";
```

Variables

x[--] quality

DISPOSITIVO DE EXPANSÃO

Model equations

```

# mass balance
f_mass:    feed.mass = drain.mass;

# pressure drop
f_p_rel:    (1.0-pressure_drop)*feed.p=drain.p;
f_p_abs:    feed.p-delta_p=drain.p;

# enthalpy is constant
f_h:    feed.h = drain.h;

# test conditions
t1:    test (delta_p>=0.0) warning "delta_p < 0.0";
t_conn:    test ( feed.Composition.FluidID == drain.Composition.FluidID )    error "different fluid at feed and drain!";

```

Variables

```

pressure_drop [---]          relative pressure drop
delta_p        [bar] (pressure_diff) absolute pressure drop

```

MOTOR

Model equations

```

# shaft only from left
ifl ref(shaft_in) && !ref(shaft_out) then
    fla: eta_el*eta_m*power + shaft_in.power = 0.0;
endifl

# shaft only from right
ifl !ref(shaft_in) && ref(shaft_out) then
    flb: eta_el*eta_m*power - shaft_out.power = 0.0;
endifl

# shaft from both sides
ifl ref(shaft_in) && ref(shaft_out) then
    flc: eta_el*eta_m*power + shaft_in.power - shaft_out.power = 0.0;
endifl

# test conditions
t1: test(power >= 0.0)      warning "power < 0.0";
t2a: test(eta_el >= 0.0)   error "eta_el < 0.0";
t2b: test(eta_el <= 1.0)   error "eta_el > 1.0";
t3a: test(eta_m >= 0.0)   error "eta_m < 0.0";
t3b: test(eta_m <= 1.0)   error "eta_m > 1.0";

```

Parameters

eta_el [---] electrical efficiency
eta_m [---] mechanical efficiency

Variables

power [kW] (power) electrical power input

CONNECTOR

Model equations

```
# mass balance is not defined!
```

```
f_p:    feed.p = drain.p;
```

```
f_h:    feed.h = drain.h;
```

```
# test for matching mass balance
```

```
t_mass:    test (abs(feed.mass - drain.mass) < 1e-3) error "mass balance not correct";
```

SEPARADOR DE LÍQUIDO

Model equations

```
# mass balance
f1:  feed.mass = drain_1.mass+drain_2.mass;

# Zero pressure drops
f2:  feed.p = drain_1.p;
f3:  feed.p = drain_2.p;

f4:  feed.mass * (1.0 - feed.Composition.f_q_ph(feed.p, feed.h)) * y_separation = drain_2.mass;

f5:  drain_2.h = drain_2.Composition.f_h_pq( drain_2.p, 0.0);

f6:  feed.h * feed.mass = drain_1.h * drain_1.mass + drain_2.h * drain_2.mass;
```

Variables

y_separation [----] degree of separation

MIXER

Model equations

```

# mass balance
f1: feed1.mass+feed2.mass-drain.mass = 0.0;
# energy balance
f2: feed1.h*feed1.mass+feed2.h*feed2.mass-drain.h*drain.mass = 0.0;
# pressure drop feed_1
f3: feed1.p-delta_p_1-drain.p = 0.0;
#pressure drop feed_2
f4: feed2.p-delta_p_2-drain.p = 0.0;

# test for positive pressure drops
t1: test (delta_p_1>=0.0)  warning "pressure drop delta_p_1 is negative";
t2: test (delta_p_2>=0.0)  warning "pressure drop delta_p_2 is negative";

#test for positive mass flows
t3: test (feed1.mass >= 0.0)      error "feed1 - mass flow is negative";
t4: test (feed2.mass >= 0.0)      error "feed2 - mass flow is negative";

# mixing is allowed for identical fluids only
t_conn1: test ( feed1.Composition.FluidID == feed2.Composition.FluidID ) error "different fluid at feed1 and feed2!";
t_conn2: test ( feed1.Composition.FluidID == drain.Composition.FluidID ) error "different fluid at feed and drain!";

```

Variables

delta_p_1 [bar] (pressure_diff) absolute pressure drop of a stream attached to feed_1
delta_p_2 [bar] (pressure_diff) absolute pressure drop of stream attached to feed_2

CONNECTOR

Model equations

```
# mass balance is not defined!
```

```
f_p:    feed.p = drain.p;
```

```
f_h:    feed.h = drain.h;
```

```
# test for matching mass balance
```

```
t_mass:    test (abs(feed.mass - drain.mass) < 1e-3) error "mass balance not correct";
```

```
t_z:    test (abs(feed.z - drain.z) < 1e-4) error "mass balance in z not correct";
```

ANEXO B - Condições operacionais não otimizadas do estudo similar de otimização desenvolvido por Dincer, Rosen e Ahmadi (2017)

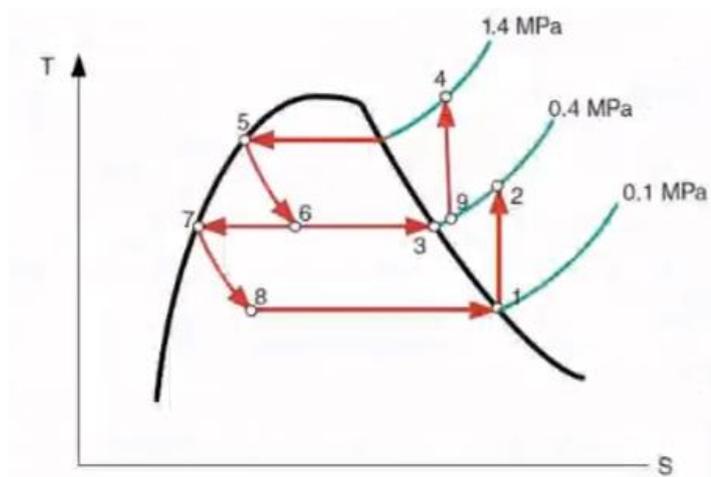


Diagrama T-s do estudo similar de otimização

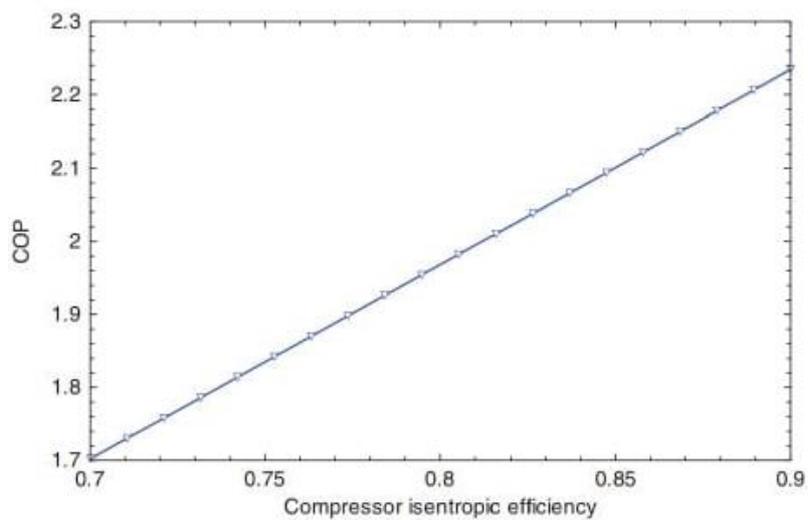
Fonte: Dincer, Rosen e Ahmadi (2017)

As condições operacionais do estudo similar de otimização são:

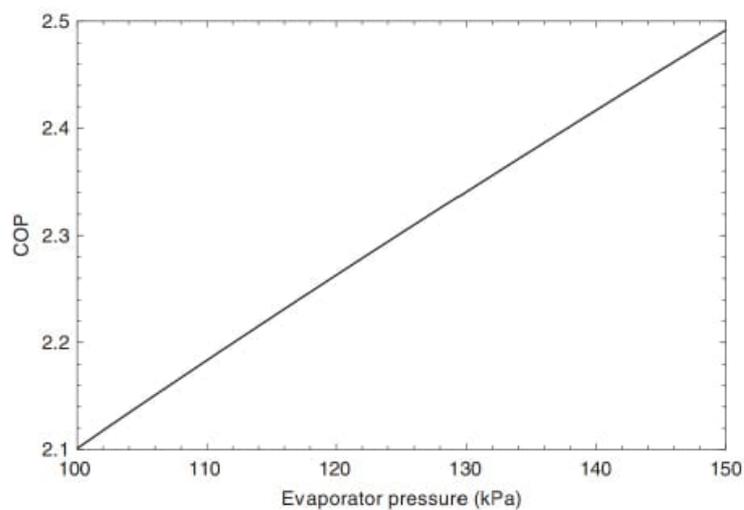
- Refrigerante: 134^a;
- Pressão de operação do evaporador: 0,1 Mpa;
- Pressão de operação do condensador: 1,4 Mpa;
- Estado do refrigerante após deixar o condensador: líquido saturado;
- Pressão de entrada no compressor de alta pressão: 0,4 Mpa;
- Estado do refrigerante após deixar o evaporador: vapor saturado;
- Compressores de alta e baixa pressões são isentrópicos
- Vazão mássica do refrigerante no condensador: 0,25 kg/s

ANEXO C – Efeitos das variáveis de decisão sobre o COP de acordo com estudo similar de otimização (Dincer, Rosen e Ahmadi, 2017)

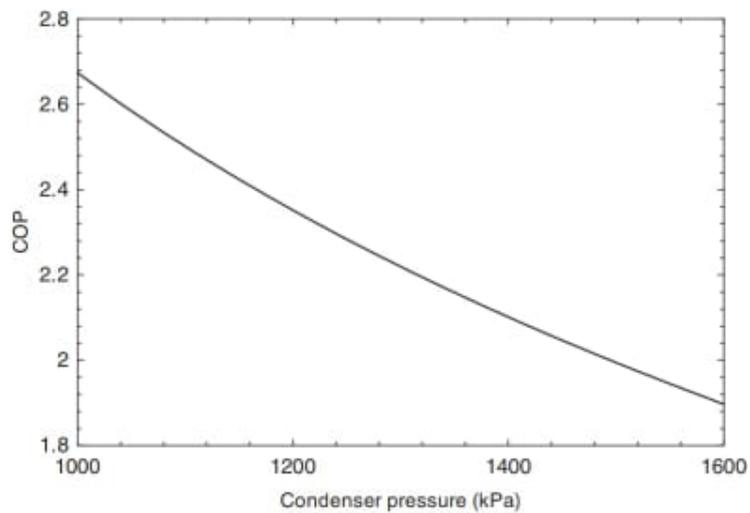
Efeito da eficiência isentrópica do compressor no COP:



Efeito da pressão do evaporador no COP:



Efeito da pressão do condensador no COP:



Efeito da pressão do separador de líquido no COP:

