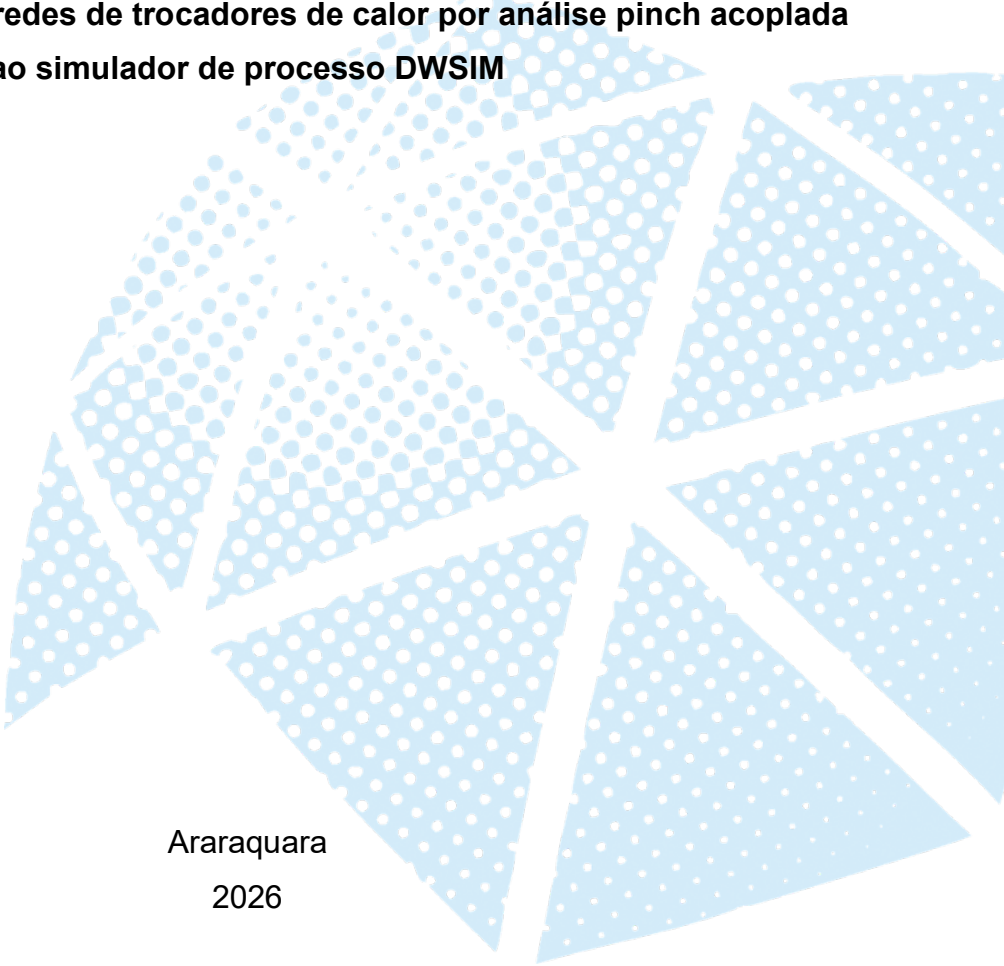


UNIVERSIDADE ESTADUAL PAULISTA – UNESP
Instituto de Química - Câmpus de Araraquara

Nathan Alves de Oliveira Framba

**Síntese automática de redes de trocadores de calor por análise pinch acoplada
ao simulador de processo DWSIM**

Araraquara
2026



Nathan Alves de Oliveira Framba

**Síntese automática de redes de trocadores de calor por análise pinch acoplada
ao simulador de processo DWSIM**

Trabalho de Conclusão de Curso
apresentado à Universidade Estadual
Paulista (UNESP), Instituto de Química,
Araraquara, para obtenção do título de
Bacharel em Engenharia Química.

Orientador(a): Prof. Dr. Elias de Souza
Monteiro Filho

Coorientador(a): Prof. Dr. Nicolas Spogis

Araraquara

2026

F813s

Framba, Nathan Alves de Oliveira

Síntese automática de redes de trocadores de calor por análise pinch
acoplada ao simulador de processo DWSIM / Nathan Alves de Oliveira Framba.

-- Araraquara, 2026

64 p. : il., tabs.

Trabalho de conclusão de curso (Bacharelado - Engenharia Química) -
Universidade Estadual Paulista (UNESP), Instituto de Química, Araraquara

Orientador: Elias de Souza Monteiro Filho

Coorientador: Nicolas Spogis

1. Engenharia química. 2. Software gratuito. 3. Recuperação do calor. 4.
Entalpia. 5. Linearização. I. Título.

Nathan Alves de Oliveira Framba

**Síntese automática de redes de trocadores de calor por análise pinch acoplada
ao simulador de processo DWSIM**

Trabalho de Conclusão de Curso apresentado à
Universidade Estadual Paulista (UNESP)
Instituto de Química, Araraquara, para obtenção
do título de Bacharel em Engenharia Química.

Data da defesa: 30 / 06 / 2026

Banca Examinadora:

Prof. Dr. Elias de Souza Monteiro Filho

Profa. Dra. Kelly Johana Dussán Medina

Prof. Dr. Leandro Martins

AGRADECIMENTOS

Agradeço aos docentes do Engenharia Química da Universidade Estadual Paulista “Júlio de Mesquita Filho”, em especial ao meu orientador Prof. Dr. Elias de Souza Monteiro Filho que me apoiou na confecção deste trabalho. Agradeço especialmente à paciência e compressão nesse período de formulação do trabalho. Estendo esse agradecimento ao meu coorientador Prof. Dr. Nicolas Spogis, não só pela paciência e compreensão, mas por todo apoio técnico com a parte de programação e interligação do código em Python com DWSIM. Esse trabalho não seria possível tal qual existe sem esse apoio e por isso agradeço imensamente.

Agradeço aos meus amigos e amigas da graduação por toda a companhia apoio e parceria, nos altos e baixos da vida universitária. São nessas conexões que eu encontrei a motivação de sempre seguir em frente

Por fim, agradeço à minha família, mas em especial aos meus pais, que me proporcionaram toda a estrutura que tive e tenho até hoje para poder me dedicar a estudar. Agradeço o apoio financeiro, logístico e principalmente emocional.

RESUMO

Este trabalho apresenta a fundamentação teórica, a formulação matemática e a metodologia computacional do aplicativo **PinchCraft**, uma ferramenta de síntese automatizada de redes de trocadores de calor (HEN, *Heat Exchanger Networks*) destinada a fluxogramas modelados no simulador de processos de código aberto DWSIM. Para garantir a possibilidade e realidade de simulação com correntes bifásicas, o sistema extrai perfis de temperatura-entalpia rigorosos diretamente do pacote termodinâmico do simulador, eliminando o uso de capacidade calorífica. A representação rigorosa é convertida em segmentos lineares por meio do algoritmo Ramer–Douglas–Peucker (RDP), com tolerância de afastamento perpendicular controlada pelo usuário, e os segmentos resultantes alimentam a biblioteca OpenPinch, que executa o algoritmo da tabela-problema e produz os alvos termodinâmicos da análise pinch. Sobre essa base, dois algoritmos heurísticos hierárquicos sintetizam a rede física: uma primeira camada constrói pares fixos a partir de preferências explícitas, calculando o calor máximo termodinamicamente permutável por varredura numérica das curvas T–H respeitando $\Delta T_{\min}$ ponto a ponto, e uma segunda camada amplia a recuperação por *cross-match* ganancioso sobre as utilidades residuais da primeira. A topologia resultante é traduzida em um novo fluxograma DWSIM, gravado em formato *.dwxml* com *layout* otimizado por força bruta para minimizar cruzamentos visuais, recarregado e resolvido em subprocesso isolado para a validação ponto-a-ponto das temperaturas-alvo. A ferramenta foi validada em cinco estudos de caso de complexidades variáveis. Em todos, o fluxograma DWSIM gerado convergiu com erro inferior a 0,1 K em relação às temperaturas-alvo, e a comparação entre os alvos de utilidade calculados pelo OpenPinch e os consumos efetivamente sintetizados pelas duas camadas heurísticas evidenciou que a estratégia adotada produz topologias coerentes com ganho de recuperação de até 5,4% entre as camadas 1 e 2. Os casos também evidenciaram as limitações da metodologia atual, apenas heurística, e quais os próximos passos de melhoria.

Palavras-chave: Integração Energética; Análise Pinch; Rede de Trocadores de Calor; Entalpia Rigorosa; DWSIM; OpenPinch; Ramer–Douglas–Peucker.

ABSTRACT

This work presents the theoretical foundation, mathematical formulation, and computational methodology of the PinchCraft application, an automated synthesis tool for heat exchanger networks (HENs) developed from flowcharts modeled in the open-source process simulator DWSIM. To ensure the possibility and realism of simulation with two-phase streams, the system extracts temperature-enthalpy profiles directly from the simulator's used thermodynamic package, eliminating the need for heat capacity. The rigorous representation is converted into linear segments using the Ramer–Douglas–Peucker (RDP) algorithm, with user-controlled perpendicular exclusion tolerance, and the resulting segments feed into the OpenPinch library, which executes the problem-table algorithm and produces the thermodynamic targets for the Pinch analysis. Based on this foundation, two hierarchical heuristic algorithms synthesize the physical network: a first layer built from fixed pairs derived from explicit specifications, calculating the maximum thermodynamically interchangeable heat by numerically scanning the T–H curves respecting $\Delta T_{\min}$ point by point, and a second layer extending the retrieval by greedy cross-matching over the residual utilities from the first layer. The resulting topology is translated into a new DWSIM flowchart, saved in .dwxmz format with a brute-force optimized layout to minimize visual intersections, reloaded, and resolved in an isolated subprocess for point-by-point validation of the target temperatures. The tool was validated in five case studies of varying complexities. In all cases, the DWSIM flowchart generated convergence with an error of less than 0.1 K in relation to the target temperatures, and the comparison between the usability targets calculated by OpenPinch and the consumption synthesized by the two heuristic layers showed that the implementation strategy produces consistent topologies with a recovery gain of up to 5.4% between layers 1 and 2. The cases also highlighted the limitations of the current, purely heuristic methodology, and the next steps for improvement.

Keywords: Energy Integration; Pinch Analysis; Heat Exchanger Network; Rigorous Enthalpy; DWSIM; OpenPinch; Ramer–Douglas–Peucker.

LISTA DE FIGURAS

Figura 01 –	Curvas compostas do Case 01	19
Figura 02 –	<i>Grand Composite Curves</i> do Case 02	20
Figura 03 –	Pipeline de processamento do <i>PinchCraft</i>	25
Figura 04 –	Perfis T-H do Case 05, com mudança de fase	28
Figura 05 –	Etapa inicial de criação de um projeto	35
Figura 06 –	Tela após criação do projeto no <i>PinchCraft</i>	35
Figura 07 –	Upload do arquivo .dwxmz	36
Figura 08 –	Página de configuração da topologia inicial	37
Figura 09 –	Parâmetros globais e continuação para próxima etapa	37
Figura 10 –	Página de sintetização da rede	38
Figura 11 –	Informações gerais e de convergência da síntese	38
Figura 12 –	Arquivo DWSIM da rede de trocadores de calor sintetizada	39
Figura 13 –	Informações de calor e eficiência da rede sintetizada	40
Figura 14 –	Perfis T-H e rede formada na síntese	40
Figura 15 –	Curvas compostas e informações dos trocadores sintetizados	41
Figura 16 –	Rede de trocadores de calor do Case 01	44
Figura 17 –	Rede otimizada sem divisões de corrente segundo literatura	45
Figura 18 –	Rede de trocadores de calor do Case 2	46
Figura 19 –	Rede otimizada do caso A do exemplo	47
Figura 20 –	Rede de trocadores de calor do Case 3	48
Figura 21 –	Rede de trocadores de calor do Case 3 apenas com a Camada 1	49
Figura 22 –	Rede de trocadores de calor do Case 4	50
Figura 23 –	Rede otimizada de trocadores de calor do artigo para $\Delta T_{\min} = 10\text{ }^{\circ}\text{C}$	51
Figura 24 –	Rede de trocadores de calor do Case 4 após mudança na Camada 1	51
Figura 25 –	Rede de trocadores de calor do Case 5	52
Figura 26 –	Rede otimizada da literatura que mais se assemelha à do aplicativo	54
Figura 27 –	Rede otimizada da literatura com menor número de trocadores de calor	55

LISTA DE QUADROS

Quadro 1 –	Especificações das correntes	42
Quadro 2 –	Comparativo entre alvos termodinâmicos calculados e consumos sintetizados na camada 2	43
Quadro 3 –	Métodos de obtenção dos materiais registrados na literatura	53

LISTA DE ABREVIATURAS E SIGLAS

Símbolos latinos

A — área de troca térmica [m^2]

C_p — capacidade calorífica específica a pressão constante [$\text{kJ kg}^{-1} \text{K}^{-1}$]

h — Entalpia específica [kJ kg^{-1}]

H — Entalpia total [kW] ou [kJ s^{-1}]

m — Vazão mássica [kg s^{-1}]

MC_p — vazão de capacidade calorífica [kW K^{-1}]

P — Pressão [Pa]

Q — Fluxo de calor [kW]

T — Temperatura [K] ou [$^{\circ}\text{C}$]

U — Coeficiente global de transferência de calor [$\text{kW m}^{-2} \text{K}^{-1}$]

z — Fração molar global da mistura [–]

Símbolos gregos

β_V — Fração molar vaporizada [–]

ΔT — Diferença de temperatura [K]

$\Delta T_{\text{mín}}$ — Diferença mínima de temperatura entre quente e fria [K]

ΔT_{cont} — Contribuição de uma corrente para $\Delta T_{\text{mín}}$ [K]

ε — Tolerância de afastamento perpendicular do algoritmo RDP [K]

Subscritos

h — Utilidade quente (hot utility)

c — utilidade fria (cold utility)

rec — Recuperado (recovered)

s — Suprimento (supply)

t — Alvo (target)

mín — valor mínimo (minimum)

Siglas

CGC — Grand Composite Curve

GCC — Grand Composite Curve

HEN — Heat Exchanger Network

HX — Heat Exchanger

MILP — Mixed Integer Linear Programming

MINLP — Mixed Integer Non-Linear Programming

PTA — Problem Table Algorithm

RDP — Ramer–Douglas–Peucker

SLSQP — Sequential Least-Squares Quadratic Programming

SUMÁRIO

1	INTRODUÇÃO.....	12
2	REVISÃO BIBLIOGRÁFICA.....	14
2.1	Origens e desenvolvimento histórico da análise pinch	14
2.2	Métodos heurísticos versus Métodos matemáticos	15
2.3	Tratamento de mudança de fase	16
2.4	Softwares de análise pinch e simuladores de processo	16
2.5	Fundamentação teórica: Cálculos	17
2.5.1	Balanco entálpico em correntes de processo.....	17
2.5.2	Curvas compostas e <i>Grand Composite Curve</i>	18
2.5.3	Algoritmo da tabela-problema	20
2.5.4	Linearização Ramer-Douglas-Peucker.....	21
2.5.5	Síntese de rede: heurísticas e programação matemática	22
3	OBJETIVOS.....	24
4	MATERIAL E MÉTODOS.....	25
4.1	Arquitetura computacional	25
4.2	Ponte Python-DWSIM via pythonnet/Automation3	26
4.3	Extração rigorosa de perfil T-H	27
4.4	Linearização piecewise e geração StreamSchema	29
4.5	Targeting via OpenPinch.....	29
4.6	Síntese – Camada 1: pares fixos com $Q_{\text{máx}}$ termodinâmico	30
4.7	Síntese – Camada 2: cross-match ganancioso	31
4.8	Geração do flowsheet do DWSIM com layout otimizado	32
4.9	Casos teste de validação	33
4.10	Procedimento geral dos testes.....	34
5	RESULTADOS.....	42

5.1	Case 1 – Quatro correntes sensíveis (Peng-Robinson)	43
5.2	Case 2 – Síntese de Amônia (Raoult modificado)	45
5.3	Case 3 – Sistema multifluxo de sete correntes	47
5.4	Case 4 – Fracionamento de ácidos graxos	49
5.5	Case 5 - Simulação de Petroquímica com 10 correntes	52
5.6	Discussão geral dos casos	55
6	CONCLUSÕES	57
7	SUGESTÕES PARA TRABALHOS FUTUROS	58
7.1	Camada 3: Síntese por Programação Matemática	58
7.2	Múltiplas Utilidades e Custo Associado	58
7.3	Heurística incremental de layout para $n > 8$ correntes	59
7.4	Validação por casos industriais reais	59
8	REFERÊNCIAS	60

1 INTRODUÇÃO

A produção industrial, seja de produtos químicos, petroquímicos, farmacêuticos ou de alimentos, sempre requer, em algum nível, de trocas de calor, seja cedendo ou retirando para uma corrente de processo. Para um fluxograma de média complexidade, é elevado o número de correntes que necessitam de resfriamento, aquecimento, condensação, vaporização ou qualquer outro tipo de troca de calor. Sem integração energética essas demandas teriam de ser atendidas exclusivamente e individualmente por utilidades, sem levar em conta o potencial energético inerente às próprias correntes do processo. Estudos da *International Energy Agency* indicam que o consumo de energia térmica é responsável por parcela majoritária — entre 30 % e 70 % — dos custos operacionais dessas instalações, e por fração comparável das emissões diretas de CO₂. A racionalização desse consumo é, portanto, um eixo central tanto da agenda econômica quanto da agenda ambiental do setor.

A tecnologia *pinch* surgiu, ao final da década de 1970, como o primeiro corpo sistemático de métodos termodinamicamente fundamentados para essa racionalização. Em sua formulação original, devida a Linnhoff e Flower (1978)¹, o problema de minimização do consumo de utilidades em um conjunto de correntes de processo é tratado independentemente da configuração específica dos trocadores: primeiro é estabelecido o *alvo* termodinâmico — o menor consumo de utilidades necessário respeitando uma diferença mínima de temperatura $\Delta T_{\min}$ entre quaisquer correntes quentes e frias em arranjo contracorrente — e, somente em uma segunda etapa, é projetada uma rede de trocadores de calor que realize esse alvo. O ponto onde a aproximação térmica entre as curvas compostas atinge $\Delta T_{\min}$, batizado por Linnhoff e Vredeveld (1984) como *pinch*, desempenha o papel de fronteira termodinâmica natural do problema, dividindo-o em duas regiões com lógicas opostas de demanda energética².

Quase meio século após sua formulação, a tecnologia *pinch* permanece a abordagem mais difundida na prática industrial, sobretudo em sua forma codificada por Linnhoff et al. (1982), Kemp (2007) e Smith (2016)^{3,4,5}. No entanto, sua aplicação prática em problemas reais permanece dificultada por duas barreiras técnicas recorrentes. A primeira é a hipótese clássica de capacidade calorífica constante por corrente, que ignora as descontinuidades introduzidas pelos pontos de bolha e de

orvalho e a forte não-linearidade da entalpia em regiões bifásicas: Westphalen e Wolf Maciel (1999) demonstraram que tal simplificação pode produzir erros de até 240 % no alvo de utilidade quente em problemas envolvendo condensação ou evaporação parciais⁶. A segunda é o trabalho manual associado a passar os dados termodinâmicos do simulador para alguma ferramenta que faça a análise pinch (planilhas de *targeting*, por exemplo) e depois passar a rede encontrada para o simulador, já que isso representa um processo repetitivo e propenso a propagação de erros, ainda mais para problemas de grande porte.

O **PinchCraft**, sistema desenvolvido neste trabalho, foi concebido para remover ambas as barreiras simultaneamente. A primeira pela substituição completa da hipótese de capacidade calorífica constante: o sistema extrai os perfis temperatura–entalpia das correntes diretamente do *pacote termodinâmico* do caso (Peng–Robinson, Soave–Redlich–Kwong, NRTL, UNIQUAC, Raoult modificado, entre outros), executando cálculos rigorosos de equilíbrio multifásico em uma malha discreta de temperaturas. A segunda pela arquitetura cliente–servidor local, em que o usuário fornece apenas o arquivo *.dwxmz* (referente ao simulador de processos DWSIM) contendo os estados de suprimento (*Supply*) e alvo (*Target*) de cada corrente, e recebe automaticamente o fluxograma resultante com a rede de trocadores sintetizada, já resolvido e validado. O acoplamento com o DWSIM (MEDEIROS, 2024)⁷ é feito pela camada Automation3 via *pythonnet*, e a análise pinch propriamente dita é delegada à biblioteca OpenPinch (WALMSLEY, 2025)⁸, desenvolvida pelo programa Ahuora da Universidade de Waikato (Nova Zelândia).

2 REVISÃO BIBLIOGRÁFICA

2.1 Origens e desenvolvimento histórico da análise pinch

A análise pinch tem suas raízes no trabalho pioneiro de Hohmann (1971), que estabeleceu o conceito de *matched limit* para o consumo mínimo de utilidades em sistemas de trocadores⁹. A formulação sistemática em si, no entanto, se deve à série de dois artigos de Linnhoff e Flower (1978) na *AIChE Journal*, em que o problema é tratado em dois estágios: primeiro a geração de redes que atingem o máximo de recuperação interna de calor, e em seguida sua evolução por critérios secundários de otimalidade. Foi nesse contexto que apareceu, ainda sob a designação de *thermodynamic pinch*, a região do problema em que as composições agregadas das correntes quentes e frias se aproximam ao máximo¹. O *Pinch Design Method* propriamente dito, que é um conjunto de regras topológicas para construir a rede a partir do pinch para “fora”, foi consolidado por Linnhoff e Hindmarsh (1983), em artigo da *Chemical Engineering Science*¹⁰ que até hoje figura entre os mais citados da área.

A primeira sistematização editorial veio com o *User Guide on Process Integration for the Efficient Use of Energy*, publicado pelo *Institution of Chemical Engineers* (LINNHOF et al., 1982)¹, seguido pela formalização do termo *Pinch Technology* em Linnhoff e Vredevelt (1984), publicação na *Chemical Engineering Progress* que apresentou também a *Grand Composite Curve*, representação que permite identificar imediatamente os intervalos térmicos adequados a utilidades específicas². Linnhoff (1993) consolidou o estado da arte em uma revisão na *Chemical Engineering Research and Design*¹¹, e Kemp (2007) incorporou desenvolvimentos posteriores como integração em processos em batelada, minimização de água e integração de sítio total⁴. A obra mais recente de referência geral é Smith (2016), que dedica oito capítulos ao tema⁵. Klemeš e Kravanja (2013) apresentam revisão crítica de quarenta anos de integração energética, articulando análise pinch e programação matemática como vertentes complementares¹².

2.2 Métodos heurísticos versus Métodos matemáticos

Em paralelo à vertente heurística, isto é, ao uso de simplificações, regras práticas, métodos empíricos ou algoritmos menos complexos (como é o caso da metodologia pinch), desenvolveu-se uma vertente baseada em programação matemática. O grande marco dessa vertente é a sequência de três artigos de Papoulias e Grossmann (1983) na *Computers & Chemical Engineering*, que introduziram a formulação de *transshipment* para o problema de *targeting* multinível de utilidades¹³. Floudas, Ciric e Grossmann (1986) propuseram, na *AIChE Journal*, o primeiro modelo MINLP (*Mixed-Integer Nonlinear Programming* ou Programação Não Linear Inteira Mista) automatizado para a síntese ótima da rede com mínimo custo de capital sob restrição de mínimo custo de utilidades¹⁴. Yee e Grossmann (1990) consolidaram a abordagem com a superestrutura SYNHEAT, baseada em estágios, em que a divisão de correntes, a seleção de *matches* e o dimensionamento das áreas são otimizados simultaneamente em um único MINLP¹⁵. A revisão crítica e bibliografia anotada de Furman e Sahinidis (2002) classifica mais de quatrocentos artigos publicados no século XX, organizando-os em três famílias: heurísticos baseados em pinch, sequenciais baseados em MILP (*Mixed Integer Linear Programming* ou Programação Linear Inteira Mista) de *transshipment* e simultâneos baseados em MINLP¹⁶.

A vertente heurística, exemplificada por Linnhoff e Hindmarsh (1983)¹⁰, Trivedi et al. (1989)¹⁷ e por implementações comerciais como *SuperTarget* (Linnhoff March) e *Aspen Energy Analyzer*, apresenta a vantagem de ser interpretável pelo engenheiro de processo, podendo ser aplicada manualmente em problemas pequenos e fornecer, mesmo em problemas grandes, um caminho razoável a ser seguido. Seu custo é a incapacidade estrutural de encontrar topologias ótimas globais, isto é, redes otimizadas não só dos trocadores de calor, como também de relações entre correntes para casos de integração energética, em problemas com forte interdependência entre correntes. A vertente matemática, por outro lado, ainda enfrenta limitações computacionais expressivas em problemas com mais de algumas dezenas de correntes e múltiplas utilidades, principalmente pela alta não-linearidade introduzida pelas relações geométricas dos trocadores. Estratégias híbridas em que a solução heurística serve de ponta pé inicial para o MINLP tem sido amplamente exploradas.

2.3 Tratamento de mudança de fase

A hipótese de capacidade calorífica (C_p) constante, central à formulação da metodologia clássica de resolução de integração energética por análise pinch, deixa de ser aceitável a partir do momento em que qualquer corrente tenha mudança de fase, vá para uma região bifásica. Westphalen e Wolf Maciel (1999), publicado no *Brazilian Journal of Chemical Engineering*, demonstraram explicitamente, em um caso teste de absorção de amônia, que a hipótese de C_p contante pode produzir erro de até 240% no alvo de utilidade quente, e propuseram a substituição do método tradicional por uma versão que admite entalpias rigorosamente calculadas por relações termodinâmicas em cada intervalo⁶. Hasan, Jayaraman, Karimi e Alfadala (2010), na *AIChE Journal*, trabalharam no problema sob a perspectiva da síntese matemática, propondo modificações na superestrutura SYNHEAT capazes de tratar mudanças de fase não-isotérmicas e estabelecendo relações analíticas para a área de troca em regime de evaporação ou condensação parcial¹⁸. Kamath, Grossmann e Biegler (2012), também na *AIChE Journal*, avançaram com modelos de *multistream heat exchangers* com e sem mudança de fase, abrindo caminho para aplicações como a integração térmica de processos criogênicos¹⁹.

2.4 Softwares de análise pinch e simuladores de processo

Do lado comercial, as ferramentas mais difundidas são *Aspen Energy Analyzer* (originalmente *HX-Net* da Hyprotech, integrado ao *Aspen HYSYS*), *SuperTarget* (Linnhoff March / KBC) e *Petro-SIM* (KBC), todas proprietárias e com licenças tipicamente prestadas a grupos industriais ou consultorias de engenharia. Do lado acadêmico e de código aberto, destacam-se *OpenPinch* e *OpenHENS* desenvolvidos no programa Ahuora da Universidade de Waikato sob coordenação de Tim Walmsley (WALMSLEY, 2025)⁸, *CPI STAR* do *Centre for Process Integration* da Universidade de Manchester e o ecossistema *CAPE-OPEN* de unidades reutilizáveis (CAPE-OPEN Laboratories Network). O DWSIM, simulador de processos completo desenvolvido por Daniel Medeiros (MEDEIROS, 2024)⁷, oferece pacotes termodinâmicos comparáveis aos comerciais (Peng–Robinson, SRK, NRTL, UNIQUAC, GERG-2008, PC-SAFT) e

expõe as APIs (*Application Program Interface*) .NET pela camada Automation3, o que o torna candidato natural à integração com bibliotecas externas.

2.5 Fundamentação teórica: Cálculos

2.5.1 Balanço entálpico em correntes de processo

A unidade fundamental do problema é a corrente de processo: uma porção do fluxograma que precisa ser levada de um estado de suprimento ou *supply* ($T_s, P, z, \dot{m}$) a um estado-alvo ou *target* ($T_t, P, z, \dot{m}$), mantendo constantes a pressão, a composição molar global z e a vazão mássica $\dot{m}$, alterando as temperaturas de entrada e saída, de *supply* (s) e *target* (t). O fluxo de calor Q associado a essa transformação é dado, de modo geral pela Equação 01.

$$Q = \dot{m} \cdot h(T_t, P, z) - \dot{m} \cdot h(T_s, P, z) \quad 01$$

Nela, $h(T, P, z)$ é a entalpia específica obtida pela equação de estado ou modelo de coeficiente de atividade do pacote termodinâmico adotado. A direção do escoamento térmico é dado pelo sinal do calor Q : correntes quentes apresentam calor negativo, pois cedem calor ($T_t < T_s$), enquanto correntes frias apresentam calor positivo, pois recebem calor ($T_t > T_s$). É usual, no entanto, que se trate os calores em módulo e some-os ou subtraia-os conforme o necessário e contextualizado em equações.

Quando a entalpia varia linearmente com a temperatura, vale a relação clássica da Equação 02, em que $MC_p = \dot{m} \cdot C_p$ é a vazão da capacidade calorífica. Para qualquer corrente que atravessasse uma região bifásica, contudo, a entalpia possui forte não-linearidade e descontinuidade de *slope* em torno dos pontos de bolha e orvalho, decorrido da inclusão do calor latente integrado sobre a faixa do *glide*. Para esses casos é necessário usar a Equação 03, na forma integral.

$$Q = \dot{m} \cdot c_p \cdot (T_t - T_s) = MC_p \cdot \Delta T \quad 02$$

$$Q = \dot{m} \cdot \int_{T_s}^{T_t} \frac{\partial h}{\partial T_{P,z}} dT \quad 03$$

Na Equação 03, a derivada parcial $(\partial h / \partial T)_{P,z}$ coincide com o calor específico aparente $C_p(T)$, que é função da temperatura, e diverge nos pontos de mudança de fase.

2.5.2 Curvas compostas e *Grand Composite Curve*

A curva composta quente é construída pelo somatório, em cada faixa de temperatura delimitada pelas temperaturas de entrada e saída de todas as correntes quentes presentes no problema multiplicadas pelas suas vazões mássicas e diferenças de entalpias em cada ponto, conforme a Equação 04. Formalmente, considerando as correntes quentes em suas respectivas faixas $[T_{t,i}, T_{s,i}]$, a curva agregada acumula calor à medida que a temperatura cai.

$$Q_{hot}(T) = \sum_{i \in hot} \dot{m}_i \cdot [h_i(T_{s,i}) - h_i(\max(T, T_{t,i}))] \quad 04$$

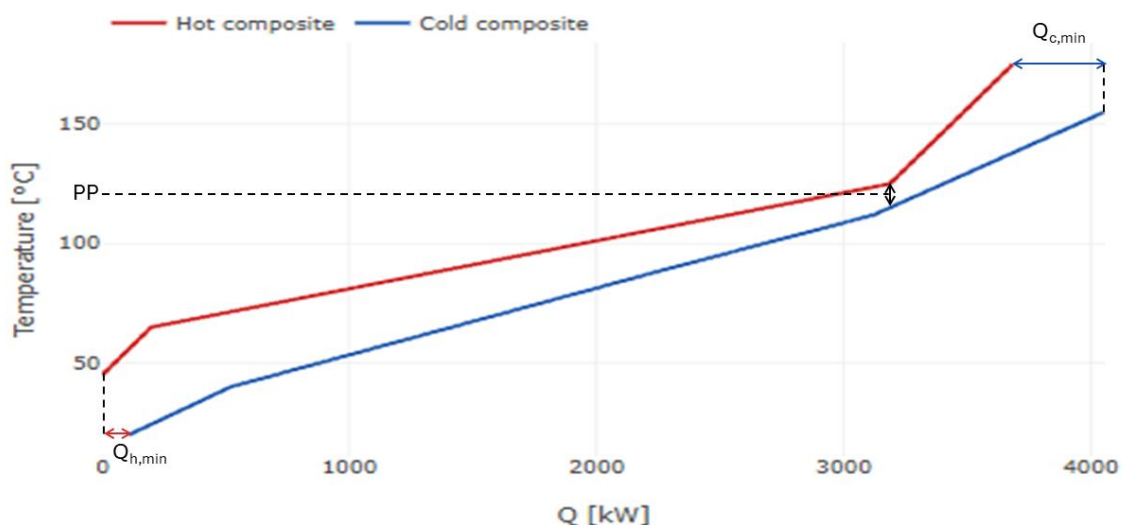
A curva composta fria é construída de maneira análoga para as correntes frias. Ao contrário da curva de correntes quentes, a curva de correntes frias tende a ganhar calor conforme a temperatura aumenta, de forma que elas tenham comportamentos opostos graficamente. Em representação gráfica (Q, T), as duas curvas são plotadas conjuntamente e a curva fria é deslocada horizontalmente em sentido a aproximá-la da quente até que a menor distância vertical entre as duas atinja o valor $\Delta T_{\min}$ prescrito, o valor da menor diferença de temperatura aceitável definido previamente.

O ponto em que essa distância é mínima é o ponto de **pinch**, o calor que ficar à esquerda da curva fria projetada até encontrar o final da curva quente corresponde à utilidade fria mínima $Q_{c,\min}$, o calor que ficar à direita da quente até o final da fria corresponde à utilidade quente mínima $Q_{h,\min}$, por fim o calor sobreposto entre as duas é a recuperação máxima $Q_{rec,\max}$. Esses três valores são, simultaneamente, alvos termodinâmicos do problema e função apenas das próprias correntes e de $\Delta T_{\min}$. Um exemplo pode ser visto na Figura 01.

Figura 01 – Curvas compostas do Case 01

Composite curves

Hot vs cold composite curves (shifted by $\Delta T_{min}/2$). The smaller the horizontal gap, the more heat is recoverable.



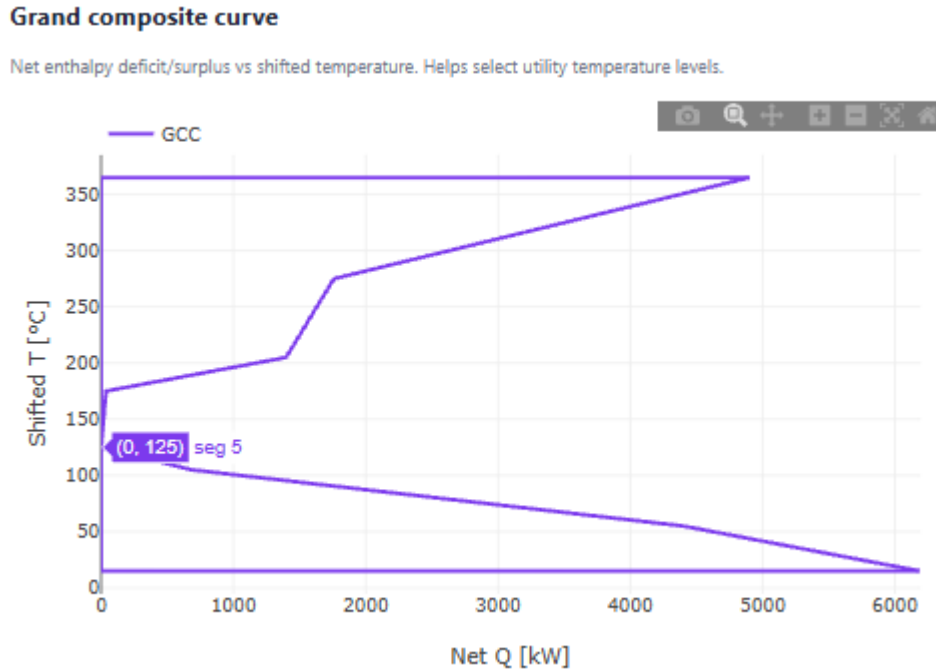
Fonte: elaborada pelo próprio autor.

Nesse exemplo, a mínima diferença de temperatura é atingida em 125°C e os calores mínimos para utilidade quente e fria são, respectivamente, 4900 kW e 6185 kW.

A *Grand Composite Curve* (GCC), introduzida por Linnhoff e Vredeveld (1984)², é uma representação derivada que plota, em uma temperatura deslocada T^* , o saldo líquido entre calor disponível pelas quentes e calor demandado pelas frias acumulado a partir do extremo superior, conforme a Equação 05, em que ΔT_{cont} é a contribuição individual de cada corrente para ΔT_{min} . O ponto em que $Q_{liq}(T^*) = 0$ é o pinch projetado sobre o eixo de temperaturas deslocadas. A GCC permite identificar diretamente, em cada faixa de temperatura, se o processo é um produtor (excesso) ou absorvedor (déficit) líquido de calor.

$$Q_{liq}(T^*) = Q_{hot}(T^* + \Delta T_{cont}) - Q_{cold}(T^* - \Delta T_{cont})$$

Figura 02 – Grand Composite Curves do Case 02



Fonte: elaborada pelo próprio autor.

2.5.3 Algoritmo da tabela-problema

O Algoritmo da Tabela-Problema (PTA) é um procedimento numérico canônico de quatro etapas para calcular os alvos termodinâmicos sem a construção gráfica das curvas compostas. Sua versão original foi descrita por Linnhoff e Flower (1978)¹. A primeira etapa é o deslocamento das temperaturas, no qual para cada corrente define-se uma temperatura deslocada T^* em que tipicamente $\Delta T_{cont} = \Delta T_{min}/2$, conforme a Equação 06. A próxima etapa é a definição de intervalos, na qual ordenam-se todas as temperaturas deslocadas T^*_k em sentido crescente e identificam-se N intervalos térmicos $[T^*_{k+1}, T^*_k]$. Em seguida, é feito o balanço por intervalo, em que cada intervalo k calcula-se o saldo líquido de calor pela Equação 07. Por fim, é feita a cascata energética, construída a partir de $H_k = 0$, passando por cada H_k , conforme a Equação 08.

$$T^*_{hot} = T - \Delta T_{cont}, \quad T^*_{cold} = T + \Delta T_{cont} \quad 06$$

$$\Delta H_k = \left(\sum_{i \in hot \cap k} MCp_i - \sum_{j \in cold \cap k} MCp_j \right) \cdot (T^*_k - T^*_{k+1}) \quad 07$$

$$H_{k+1} = H_k + \Delta H_k \quad 08$$

O menor valor, mais negativo, atingido pela cascata, em módulo, é a quantidade de calor da utilidade quente mínima necessária para que não haja déficit de energia, conforme a Equação 09. A mínima utilidade fria é definida pelo valor no extremo inferior da cascata corrigida, após adição de $Q_{h,min}$ no topo, e o ponto de pinch é o intervalo cuja cascata corrigida atinge zero. A *Grand Composite Curve* é a própria curva $H_k(T^*)$ corrigida.

$$Q_{h,min} = \max(0, -\min_k H_k) \quad 10$$

2.5.4 Linearização Ramer-Douglas-Peucker

O algoritmo da tabela-problema admite, em sua forma canônica, segmentos lineares por trecho: cada corrente é caracterizada por uma sequência de segmentos ($T_{inicial}$, T_{final} , MCp). Quando a entalpia é obtida por flash rigoroso em uma malha discreta de temperaturas, a redução para essa representação requer uma *linearização piecewise* (linearização por partes) que respeite a fidelidade entálpica original. O algoritmo Ramer–Douglas–Peucker (RAMER, 1972; DOUGLAS; PEUCKER, 1973) é a escolha clássica para esse tipo de simplificação, e tem sido amplamente utilizado em compressão de polilinhas, simplificação de séries temporais e geração de cartografia digital^{20, 21}.

Dada uma sequência (T_0, H_0) a (T_n, H_n) e uma tolerância ε , o algoritmo procede recursivamente: traça o segmento entre os extremos e calcula, para cada ponto intermediário (T_k, H_k) , o afastamento perpendicular, de maneira a linearizar cada trecho, conforme a Equação 11.

$$d_k = \frac{|(T_n - T_0)(H_0 - H_k) - (T_0 - T_k)(H_n - H_0)|}{\sqrt{(T_n - T_0)^2 + (H_n - H_0)^2}} \quad 10$$

Se $\max_k d_k < \varepsilon$, todos os pontos intermediários são descartados e o segmento reto entre os extremos é aceito. Caso contrário, a sequência é particionada no ponto $k^* = \arg \max_k d_k$ e o procedimento é aplicado recursivamente aos dois subintervalos.

O algoritmo termina quando todas as regiões satisfazem o critério de tolerância. A complexidade no pior caso é $O(n^2)$, com média $O(n \log n)$.

No OpenPinch, o RDP é encadeado com um refinamento por programação não-linear sequencial (SLSQP) cujo objetivo é assegurar que os segmentos resultantes mantenham coerência quente/fria quando submetidos ao deslocamento ΔT_{cont} do PTA. Esse refinamento é particularmente importante na vizinhança de pontos de bolha e orvalho, onde uma simplificação poderia produzir segmentos que cruzam a fronteira de saturação e ferem o balanço entálpico cumulativo.

2.5.5 Síntese de rede: heurísticas e programação matemática

Uma vez calculados os alvos termodinâmicos, a síntese da rede consiste em definir três níveis de decisão: a topologia (quais correntes trocam calor entre si), a sequência (em que ordem os trocadores aparecem em cada corrente) e o dimensionamento (área, coeficiente global, número de cascos). O *Pinch Design Method* (LINNHOFF; HINDMARSH, 1983) opera sobre essas decisões por meio de regras topológicas explícitas, número mínimo de trocadores, prioridade para pares acima/abaixo do pinch, divisão obrigatória quando há violação de MCp^{10} , e atinge soluções normalmente subótimas em poucos minutos. Os métodos por programação matemática, exemplificados pela superestrutura SYNHEAT (YEE; GROSSMANN, 1990) e pelas formulações de Floudas, Ciric e Grossmann (1986), articulam essas três decisões em um único problema MINLP de minimização do custo anual total, em que a função objetivo combina custo de utilidades, custo de capital (proporcional a área elevada a 0,6 ou expressão similar) e custos operacionais^{15,14}. A relação $UA = Q / \Delta T_{LM}$, com ΔT_{LM} definida como a média logarítmica das diferenças nas duas extremidades, conforme Equação 11, fornece a base para o dimensionamento de cada trocador uma vez conhecida a topologia.

$$\Delta T_{LM} = \frac{\Delta T_1 - \Delta T_2}{\ln(\Delta T_1 / \Delta T_2)} \quad 11$$

Para o caso do simulador relevante para esse trabalho: os trocadores DWSIM são criados com modo de cálculo *CalcBothTemp* (especifica-se Q e o solver calcula as

temperaturas de saída), deixando a área como variável de projeto a ser calculada pelo usuário em uma etapa subsequente.

3 OBJETIVOS

Este trabalho tem por objetivo o desenvolvimento de uma síntese automática de redes de trocadores de calor por análise pinch acoplada ao simulador de processo DWSIM, isto é, desenvolver uma ferramenta que permita a integração energética completa, sem, ou com a menor quantidade possível por enquanto, de intervenção manual em qualquer etapa do processo, partindo de um conjunto de correntes definidas em um fluxograma no DWSIM até um arquivo com a rede ótima definida e funcional no simulador. Como pré-requisito ou suposição associado a esse objetivo, está a ideia de usar entalpia ao invés de capacidade calorífica, para permitir mudança de fase ao longo da integração energética.

4 MATERIAL E MÉTODOS

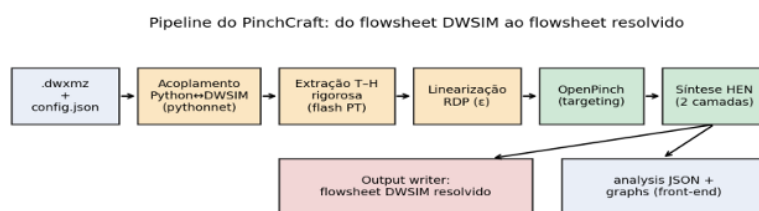
Boa parte das informações dessa seção do trabalho foram fornecidas pelo Prof. Dr. Nicolas Spogis, coorientador deste trabalho e responsável pela integração do DWSIM com a lógica da integração energética desenvolvida em grande parte pelo autor do trabalho (em parte adaptada pelo professor para possibilitar a integração). Quando necessário ele será referenciado como SPOGIS, 2026.

4.1 Arquitetura computacional

O PinchCraft é estruturado em três camadas computacionais distintas, organizadas segundo o padrão arquitetural *client-server* local: um núcleo de engenharia em Python puro (pacote *pinchhen*), um serviço HTTP local sobre FastAPI (RAMIREZ, 2018)²², e uma interface gráfica *single-page* em HTML/CSS/JavaScript. As três camadas comunicam-se pela porta TCP 8068 do *localhost*, dispensando autenticação na hipótese de operação por usuário único.

A Figura 3 sintetiza o fluxo de dados do sistema. A partir de um único arquivo *.dwxmlz* contendo as correntes em seus estados de suprimento e alvo, encadeiam-se: o acoplamento Python–DWSIM (carregamento de DLLs e leitura do flowsheet), a extração rigorosa de perfis T–H ponto-a-ponto via flash isobárico PT, a linearização Ramer–Douglas–Peucker, o *targeting* via OpenPinch, a síntese da rede em duas camadas heurísticas e, finalmente, a tradução da topologia em um novo flowsheet DWSIM resolvido por meio do *output writer*. A análise estatística completa (alvos, gráficos de Composite *Curve* e GCC, validação ponto-a-ponto das temperaturas etc.) é serializada em formato JSON e exposta à camada de apresentação.

Figura 03 – Pipeline de processamento do PinchCraft



Fonte: SPOGIS, 2026.

O núcleo *pinchhen* é deliberadamente desacoplado da camada de serviço, podendo ser usado tanto pela aplicação web quanto por *scripts* de linha de comando em *pipelines* de processamento em lote. Esse desacoplamento foi adotado para que a lógica de engenharia possa ser submetida a testes unitários e a benchmarks independentes da interface gráfica, e para que o sistema possa, em desenvolvimentos futuros, ser embutido em ambientes diferentes, como, por exemplo, *plug-in* nativo do DWSIM ou como *worker* Celery em arquitetura distribuída.

4.2 Ponte Python-DWSIM via pythonnet/Automation3

O acoplamento entre Python 3.11 e o motor termodinâmico do DWSIM Classic, implementado em .NET Framework 4.7.2, é construído sobre a biblioteca *pythonnet* 3.0.5, que expõe a *Common Language Runtime* como módulo Python nativo. A inicialização segue um padrão validado experimentalmente:

- (i) chamada a `pythoncom.Colnitalize()` antes de qualquer "import clr", para garantir a inicialização correta do apartamento COM utilizado por componentes do simulador;
- (ii) adição do diretório de instalação do DWSIM ao `sys.path` e mudança de diretório de trabalho para esse caminho, já que alguns pacotes termodinâmicos buscam arquivos de dados em caminhos relativos ao `cwd`;
- (iii) carregamento explícito por `clr.AddReference` dos assemblies essenciais: `DWSIM.Interfaces.dll`, `DWSIM.SharedClasses.dll`, `DWSIM.Thermodynamics.dll`, `DWSIM.UnitOperations.dll`, `DWSIM.Automation.dll` e diversos plugins de equações de estado avançadas;
- (iv) supressão dos avisos benignos emitidos pelos pacotes termodinâmicos opcionais (ThermoCS, Electrolytes), redirecionando `System.Console.Out` e `System.Console.Error` para um `TextWriter` nulo durante a janela de inicialização.

A Fluent API do DWSIM (`DWSIM.Automation.FluentAPI.dll`) foi avaliada como alternativa mais ergonômica, mas descartada após verificação experimental de um *bug* do *pythonnet* 3.0.5 com sub-namespaces compartilhados: ainda que o *assembly* carregue e os tipos sejam visíveis por reflexão .NET, o módulo Python "DWSIM.Automation.FluentAPI" não é exposto. Optou-se, em consequência, pela camada Automation3 clássica, robusta e plenamente funcional.

A criação de uma `MaterialStream` foi sempre feita via `flowsheet.AddObject(ObjectType.MaterialStream, x, y, name)`: a instanciação direta por construtor produz fases sem compostos e impossibilita qualquer flash subsequente. A referência `PropertyPackage.CurrentMaterialStream` foi sempre apontada explicitamente antes de cada `CalculateEquilibrium` e repontada após qualquer chamada à `MaterialStream.Calculate()`, método que pode silenciosamente redefinir essa referência em determinadas versões.

Um problema sutil identificado durante a validação, por meio de três cases de teste, é a contaminação do **estado global** do DWSIM entre chamadas sucessivas à `LoadFlowsheet` no mesmo processo Python. Detectou-se, ao executar os três casos em sequência sem reiniciar o *interpreter*, que as `MaterialStreams` do segundo e terceiro casos eram silenciosamente reposicionadas em estado 25 °C / 101325 Pa após *Solve*. A análise da causa-raiz identificou o cache de `PropertyPackages` e o singleton `GlobalSettings` como responsáveis. A solução estrutural adotada foi o despacho de cada execução de síntese para um subprocesso Python descartável, dedicado, lançado pelo módulo `runner_child.py`. O custo é da ordem de quinze segundos por execução (boot das DLLs), aceitável dada a confiabilidade obtida (SPOGIS, 2026)

4.3 Extração rigorosa de perfil T-H

Para cada corrente do problema, definida pelo par de *tags supply/target* da `MaterialStream`, o sistema constrói um perfil termodinâmico discreto entre as duas temperaturas. A varredura padrão usa $n = 60$ pontos linearmente espaçados, conforme a Equação 12.

$$T_i = T_s + \frac{i}{n-1} \cdot (T_t - T_s), \quad i = 0, 1, \dots, n-1 \quad 12$$

Em cada T_i é executado um flash isobárico PT, mantendo constantes P , $\dot{z}$ e $\dot{m}$, e são obtidas simultaneamente a entalpia específica h_i , a fração molar vaporizada $\beta_{v,i}$ e as entropias de cada fase. O fluxo de calor cumulativo, contado a partir do estado de suprimento, é dado na Equação 13.

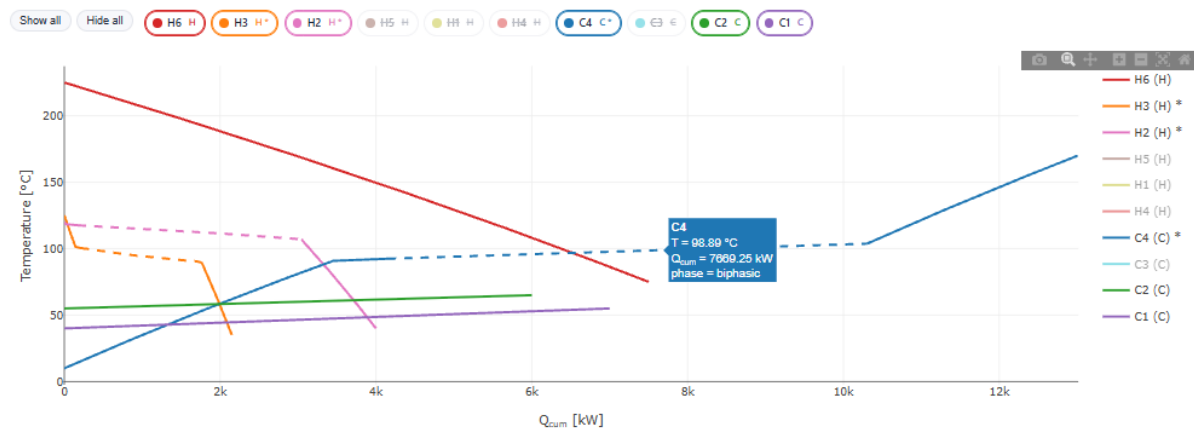
$$Q_{cum,i} = \dot{m} \cdot |h_i - h_s|$$

O módulo é usado porque o sentido físico (resfriamento para correntes quentes, aquecimento para correntes frias) é tratado em separado pela classificação *hot/cold*: a corrente é *hot* se $T_t < T_s$, *cold* caso contrário. A propriedade *has_phase_change* do perfil retorna verdadeiro se ao menos um ponto possuir $10^{-3} < \beta_V < 1 - 10^{-3}$, sinalizando às camadas seguintes que cuidados especiais devem ser tomados na interpretação das diferenças de temperatura no interior do trocador. Os perfis T-H do Case 05, que possui mudança de fase, podem ser vistos na Figura 04.

Figura 04 – Perfis T-H do Case 05, com mudança de fase

Stream T-H profiles (rigorous)

Individual temperature–enthalpy curves of each logical stream, extracted from DWSIM by isobaric PT flash. Solid lines: sensible region; dashed segments: two-phase region ($0 < \beta_V < 1$). The non-linearity is the reason PinchCraft uses rigorous enthalpy instead of constant c_p .



Fonte: elaborada pelo próprio autor.

Antes da extração, a consistência entre os dois MaterialStreams da mesma corrente lógica é verificada quanto a (a) pressão, com tolerância relativa de 0,1 %, (b) vazão mássica, com a mesma tolerância; e (c) composição molar global, com tolerância absoluta de 10^{-6} . Discrepâncias maiores resultam em ValueError explícito antes que qualquer flash seja executado, evitando que diferenças sistemáticas em fronteiras de processo sejam silenciosamente integradas como calor latente fictício. Tal verificação preventiva é central à confiabilidade do encadeamento subsequente.

4.4 Linearização piecewise e geração StreamSchema

O OpenPinch, à semelhança da maioria das implementações canônicas do PTA, opera sobre correntes representadas por segmentos lineares (T_{inicial} , T_{final} , MC_p). A conciliação entre a representação rigorosa T–H ponto-a-ponto e essa representação por segmentos é realizada pelo algoritmo RDP, com tolerância padrão $\varepsilon = 1$ K na coordenada T para fluxos de calor expressos em kW. O parâmetro é exposto na API pública do PinchCraft, permitindo o controle direto pelo usuário em casos em que o *glide* bifásico exija maior resolução.

A invocação do RDP é encadeada com o refinamento SLSQP exposto pelo OpenPinch em `get_piecewise_data_points()`, que aceita as flags `is_hot_stream` e `dt_diff_max` e devolve o conjunto reduzido de pontos. Cada corrente física é, portanto, mapeada em uma família de StreamSchema do OpenPinch — todas compartilhando o mesmo nome — cada uma representando um segmento (T_{supply} , T_{target} , heat_flow , dt_cont , etc). O padrão de múltiplos schemas com o mesmo identificador é o mecanismo nativo do OpenPinch para representar uma única corrente física com várias regiões termodinâmicas.

4.5 Targeting via OpenPinch

Os segmentos linearizados são organizados em uma instância de TargetInput do OpenPinch, complementada por uma lista de utilidades disponíveis (ar de resfriamento e vapor no padrão atual, com extensões previstas para água de resfriamento e múltiplos níveis de vapor). O serviço `pinch_analysis_service` do OpenPinch é então invocado, e produz um TargetOutput contendo: os alvos termodinâmicos $Q_{h,\text{mín}}$, $Q_{c,\text{mín}}$, $Q_{\text{rec},\text{máx}}$ e as temperaturas de pinch quente e fria, as curvas compostas em sua forma final parametrizadas por (Q, T) , a Grand Composite Curve e outras representações como Balanced Composite e Site Composite, esta última relevante em problemas de integração de sítio total.

A relação entre $\Delta T_{\text{mín}}$ total do problema e a contribuição ΔT_{cont} por corrente é estabelecida no PinchCraft pela convenção $\Delta T_{\text{cont}} = \Delta T_{\text{mín}}/2$, atribuindo a cada corrente metade da diferença mínima global, em consonância com a prática consolidada quando não se dispõe de informações específicas sobre os coeficientes

individuais de transferência de calor. O usuário define $\Delta T_{\min}$ diretamente na interface (padrão 10 K), e a conversão para ΔT_{cont} é executada de modo transparente antes da chamada ao OpenPinch.

4.6 Síntese – Camada 1: pares fixos com $Q_{\text{máx}}$ termodinâmico

A primeira camada de síntese opera sobre pares preferenciais quente–fria, definidos explicitamente pelo usuário ou inferidos automaticamente por convenções de nomeação (_supply/_target , _in/_out , etc.). Para cada par (HOT_i , $COLD_j$), o cálculo do calor máximo termodinamicamente trocável não pode utilizar a hipótese MC_p constante, caso contrário a vantagem da entalpia rigorosa seria anulada. O procedimento implementado, denominado *feasible_Q_max*, executa uma varredura numérica de Q sobre uma grade de 200 valores espaçados linearmente entre 0 e $Q_{\text{balanço}}$, com ele sendo igual ao mínimo entre o calor total de cada item do par.

Para cada Q candidato, as temperaturas instantâneas da quente e da fria no trocador são obtidas por interpolação inversa $T(h)$ na respectiva curva rigorosa do perfil, conforme equação 14.

$$T_h(Q) = T^{-1}(h_{s,h} - Q / \dot{m}_h), \quad T_c(Q) = T^{-1}(h_{s,c} + Q / \dot{m}_c) \quad 14$$

Em arranjo contracorrente, as diferenças de temperatura nas duas extremidades de um trocador encarregado de transferir o calor Q podem ser vistas nas Equações 15 e 16, nas quais A representa a extremidade quente do trocador (entrada quente e saída fria) e B a extremidade fria. O critério de viabilidade é a satisfação simultânea das extremidades serem maiores ou menores do que $\Delta T_{\min}$, conforme Equação 17 e $Q_{\text{máx}}$ é o maior valor de Q que satisfaz simultaneamente os dois critérios.

$$\Delta T_A(Q) = T_h(0) - T_c(Q) \quad 15$$

$$\Delta T_B(Q) = T_h(Q) - T_c(0) \quad 16$$

$$\Delta T_A(Q) \geq \Delta T_{\min} \quad \wedge \quad \Delta T_B(Q) \geq \Delta T_{\min} \quad 17$$

O procedimento permite identificar não apenas situações de cruzamento térmico globalmente inviável, em que o par é descartado e ambas as correntes recebem utilidade integral, mas também pinches internos a um único par com mudança de fase, em que a inversão de *slope* da curva $T(h)$ na região bifásica produz $\Delta T_{\min}$ no interior do trocador, fora das extremidades. Para a camada 1, a verificação ponto-a-ponto interna ao trocador é simplificada à inspeção das duas extremidades, justificada quando a derivada dT/dh permanece monótona ao longo da troca.

Uma vez determinado $Q_{\max}$ do par, com todas as condições satisfeitas, a topologia preliminar inclui um trocador HX entre as duas correntes, um aquecedor residual (HEATER) na corrente fria se $Q_{\max} < Q_{\text{cold,total}}$ e/ou um resfriador residual (COOLER) na quente se $Q_{\max} < Q_{\text{hot,total}}$. As correntes lógicas sem par preferencial são tratadas com utilidade integral. O resultado é uma estrutura de dados HENTopology contendo as listas de HXs e utilidades, com calores, temperaturas de entrada e saída e ordem na cadeia de cada corrente. Um exemplo de rede formada com a Camada 01 pode ser vista na Figura 21, na parte de Resultados deste trabalho.

4.7 Síntese – Camada 2: cross-match ganancioso

A camada 2 usa da topologia produzida pela camada 1 e procura ampliar a recuperação de calor por meio de *cross-matches*: trocas entre uma janela residual de uma corrente quente (atendida por um cooler) e uma janela residual de uma corrente fria (atendida por um heater) pertencentes a pares distintos. A iteração é gananciosa: a cada passo, todos os pares (cooler, heater) com Q residual superior a 0,1 kW são avaliados e o par com maior Q_{cross} viável é selecionado. Um novo trocador HX-CROSS-k é, então, inserido e os calores residuais são atualizados conforme a Equação 18.

$$Q_{\text{cooler,novo}} = Q_{\text{cooler,velho}} - Q_{\text{cross}}, \quad Q_{\text{heater,novo}} = Q_{\text{heater,velho}} - Q_{\text{cross}} \quad 18$$

Se o Q residual de uma utilidade cair abaixo de 1 mW (limiar prático para ponto flutuante), o bloco é removido. O processo se encerra quando não há mais nenhum par cooler/heater capaz de produzir Q_{cross} superior ao limiar mínimo de aperfeiçoamento ou quando se atinge o número máximo de iterações, que por padrão

é 10. A função `_feasible_cross_Q`, adaptação de `feasible_Q_max`, respeita não apenas as restrições de $\Delta T_{\min}$ nas extremidades, mas também as restrições de borda: a temperatura de saída da quente não pode ser inferior à temperatura-alvo original (a corrente não pode ser super-resfriada) e a temperatura de saída da fria não pode ultrapassar a temperatura-alvo original. A camada 2 não realiza divisão de corrente nem reordenação dos HX já existentes, apenas substitui, se possível, pares de utilidades por trocadores.

4.8 Geração do flowsheet do DWSIM com layout otimizado

A topologia produzida pelas camadas anteriores é traduzida, no módulo `output_writer`, em um novo flowsheet DWSIM funcional. O *layout* do canvas é o aspecto técnico mais delicado dessa etapa: o `NaturalLayout` e o `AutoLayout` nativos do simulador foram projetados para flowsheets de processo convencionais e produzem, em redes HEN, layouts visualmente confusos com cruzamentos excessivos. A solução adotada parte da convenção de Linnhoff — cada corrente em uma linha horizontal própria — e introduz uma **otimização combinatória** da ordem vertical das correntes para minimizar o custo de cruzamentos: $C(\sigma) = \sum_{hx \in HXS} (|\sigma(hx.hot) - \sigma(hx.cold)| - 1)$, em que σ é uma permutação das correntes (atribuição de índices verticais) e $\sigma(s)$ é o índice da corrente s . Pares adjacentes (distância vertical 1) contribuem com custo zero, refletindo o fato de que o trocador entre eles cabe no canal vazio entre suas linhas. Para problemas com até oito correntes, a permutação ótima é obtida por força bruta sobre as $8! = 40\,320$ permutações, prática viável e rápida. Para problemas maiores, uma heurística de ordenação por temperatura de suprimento (hots) e por temperatura-alvo (colds) é usada como *fallback*.

A posição horizontal dos blocos é determinada por um esquema *greedy* sobre a cadeia de cada corrente: cada bloco recebe a próxima coluna disponível, com HXs compartilhados entre quente e fria reutilizando a mesma coluna. As `MaterialStreams` intermediárias (estados pós-HX, pós-heater, pós-cooler) ficam em *offset* horizontal pequeno após o bloco que as gerou. Os `EnergyStreams` que alimentam ou descarregam calor dos heaters e coolers recebem *offset* vertical em relação ao bloco principal, ajustado conforme a porta de conexão de cada operação unitária — heaters

do DWSIM exigem o EnergyStream como porta de entrada 1 (Primary), enquanto coolers exigem-no como conexão de saída pelo EnergyConnector separado.

A replicação dos compostos e do pacote termodinâmico exige cuidado adicional. A criação direta de uma instância do PropertyPackage de origem por Activator.CreateInstance preserva o tipo (Peng–Robinson, Soave–Redlich–Kwong, NRTL, etc.), mas pode falhar na serialização do flowsheet de saída se o nome canônico não for o esperado pelo DWSIM. O output_writer resolve essa nuance percorrendo o dicionário auto.AvailablePropertyPackages e atribuindo ao novo pacote o nome canônico do tipo (por exemplo, "Peng-Robinson (PR)" e não apenas "PR"), o que evita erros silenciosos no LoadFlowsheet subsequente.

Após a montagem, o novo arquivo .dwxmz é gravado e imediatamente recarregado em um Automation3 limpo do mesmo subprocesso, com chamada a CalculateFlowsheet4 para validar a convergência. A consistência das temperaturas finais é então conferida contra as temperaturas-alvo originais com tolerância de 0,5 K, conforme detalhado na seção 5. Essa etapa de validação automatizada elimina a categoria mais frequente de falha silenciosa — flowsheets topologicamente corretos, mas termodinamicamente inconsistentes — antes da disponibilização do resultado ao usuário (SPOGIS, 2026).

4.9 Casos teste de validação

Foram selecionados três casos de complexidade crescente para validação do pipeline completo e (Cases 1 a 3) e depois mais dois casos (Cases 4 e 5) para testes adicionais principalmente da parte do aplicativo e sua funcionalidade para integração energética. São eles:

- 1) Case 1 - quatro correntes puramente sensíveis (duas quentes e duas frias), modeladas com Peng–Robinson (PENG; ROBINSON, 1976)²³, destinado a validar o encadeamento básico sem o complicador de mudança de fase;
- 2) Case 2 - quatro correntes representando parte do circuito de síntese de amônia (cinco componentes: N₂, H₂, NH₃, Ar, CH₄), com pacote termodinâmico baseado em Raoult modificado, destinado a validar o tratamento de mudança de fase parcial;

- 3) Case 3 - sistema multifluxo com sete correntes envolvendo Dowtherm A, água, hidrocarbonetos leves e mistura de glicóis, destinado a estressar a otimização combinatória de layout e o cross-match da camada 2;
- 4) Case 4 – cinco correntes de fracionamento de ácidos graxos (ácidos mirístico, palmítico, esteárico, oleico e linoleico), modeladas com Peng-Robinson, para validação da efetividade (ou da dependência) da topologia inicial da Camada 01;
- 5) Case 5 – simulação dez correntes de uma petroquímica, também modelado com Peng-Robinson, para analisar o comportamento do aplicativo com um número elevado de correntes e com região bifásica.

O arquivo original de validação contava com um conjunto de 14 correntes de entrada e 14 de saída, para os três primeiros casos em conjunto. Foi nesse arquivo que ocorreu o *bug* mencionado anteriormente. Os casos seguintes foram feitos em arquivos individuais. Os dois primeiros casos foram adaptados do trabalho *Benchmark solutions for small heat exchanger networks*, de Declerq (2018)²⁴. O terceiro caso foi modelo conforme o exemplo de Juan Diego em seu código *Método de punto de plieque*²⁵. O quarto caso foi modelado baseado no artigo *Initial Analysis on Heat Exchanger Networks of Fatty Acid Fractionation Plant to Optimize Energy Recovery and Controllability* de Bakar (2015)²⁶. Por fim, o quinto e último caso veio de uma adaptação de Declerq (2015)²⁷, *Insight in heat exchanger network synthesis: the ten streams literature case from Ahmad (1985) – Case 18*.

Como a maior parte dos artigos e textos que estudam integração energética não apresentam as composições das suas correntes, elas foram adaptadas com base em seus calores, *range* de temperaturas e MC_p , tal que se tenham casos próximos do real, mas que não condizem 100% com os exemplos.

4.10 Procedimento geral dos testes

Todos os testes seguiram o mesmo procedimento geral, mudando apenas para os casos em que se testou uma segunda configuração da camada 1, nos quais algumas etapas de teste foram repetidas ou alteradas. Segue o procedimento em seguida:

1. Abrir o aplicativo no link <https://pinchcraft.ai4tech.ai/>;
2. Dar um nome ao Projeto e clicar em *Create Project*, conforme Figura 05. A tela deve mudar para a da Figura 06.

Figura 05 – Etapa inicial de criação de um projeto

1. Project & Upload

Create a new project or open an existing one. Upload the source DWSIM flowsheet containing the material streams (supply and target states for each logical stream).

New project

Project name

Example Case - Case 2

Create project

Fonte: elaborado pelo próprio autor.

Figura 06 – Tela após criação do projeto no *PinchCraft*

1. Project & Upload

Create a new project or open an existing one. Upload the source DWSIM flowsheet containing the material streams (supply and target states for each logical stream).

New project

Project name

e.g. Refinery HX retrofit case

Create project

Existing projects

Project Name	Open
Example Case - Case 2 02/07/2026, 11:50:33	Open x
Case 3 - NH3 Plant Integrated 7-Stream Network 19/05/2026, 19:09:07 · input ok · config ok · layer 2 run	Open x
Case 4 - Fatty Acid Fractionation 09/06/2026, 20:01:12 · input ok · config ok · layer 2 run	Open x
Case 5 - 10 Stream Petrochemical Simulation 09/06/2026, 21:46:33 · input ok · config ok · layer 2 run	Open x
Case 1 - NH3 Plant Mid-Temp Heat Recovery 19/05/2026, 19:09:07 · input ok · config ok · layer 2 run	Open x

Upload DWSIM flowsheet (.dwxmz or .dwxml)

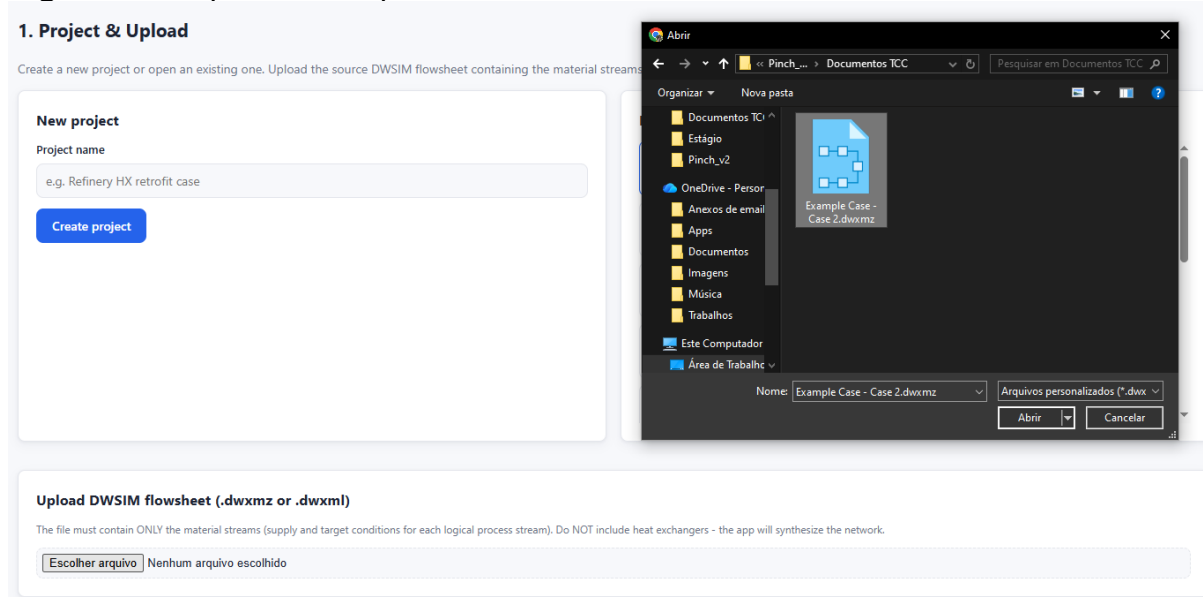
The file must contain ONLY the material streams (supply and target conditions for each logical process stream). Do NOT include heat exchangers - the app will synthesize the network.

Escolher arquivo | Nenhum arquivo escolhido

Fonte: elaborado pelo próprio autor.

3. Fazer o Upload do arquivo .dwxmz (DWSIM) apenas com as correntes, sem trocadores de calor ligados nelas, conforme a Figura 07;

Figura 07 – Upload do arquivo .dwxmz



Fonte: elaborado pelo próprio autor.

4. Ir para a próxima página *Configure Pairs*, conforme Figura 08. Esperar carregar automaticamente. A primeira tabela dará as correntes identificadas e suas informações. A segunda tabela é editável e indica a topologia. Por fim, pode-se modificar a temperatura de approach, o $\Delta T_{\min}$, conforme Figura 09;

Figura 08 – Página de configuração da topologia inicial

1 Project & Upload 2 **Configure Pairs** 3 Run Synthesis 4 Analysis & Network

2. Configure Pairs

Define which material streams form a logical process stream (supply + target). Auto-detected by tag suffix conventions (_supply / _target , _in / _out , _sup / _tgt). Adjust as needed.

Material streams in the flowsheet

Read-only view of all `MaterialStream` objects found in the uploaded .dwgmx.

Tag	T [°C]	P [bar]	m [kg/h]	β_V	Phase	Composition (mol)
C8_OUT	260.00	1.01	36273.00	1.000	V	Nitrogen: 25.0%, Hydrogen: 75.0%
C8_IN	120.00	1.01	36273.00	1.000	V	Nitrogen: 25.0%, Hydrogen: 75.0%
C5_OUT	60.00	1.01	35814.00	1.000	V	Nitrogen: 30.0%, Hydrogen: 70.0%
C5_IN	280.00	1.01	35814.00	1.000	V	Nitrogen: 30.0%, Hydrogen: 70.0%
C6_OUT	20.00	1.01	35503.00	1.000	V	Nitrogen: 17.0%, Hydrogen: 83.0%
C6_IN	180.00	1.01	35503.00	1.000	V	Nitrogen: 17.0%, Hydrogen: 83.0%
C7_OUT	200.00	1.01	35647.00	1.000	V	Nitrogen: 20.0%, Hydrogen: 80.0%
C7_IN	100.00	1.01	35647.00	1.000	V	Nitrogen: 20.0%, Hydrogen: 80.0%

Logical streams (editable) Re-run auto-detect

Each logical stream maps a **supply** material stream to a **target** material stream. Type (hot/cold) is inferred from supply vs target temperature. Add a **preferred pair** to tell the heuristic which hot/cold to match.

ID	Supply	Target	Type	Preferred pair	Notes
C5	C5_IN (280.0 °C)	C5_OUT (60.0 °C)	HOT	(none)	×
C6	C6_IN (180.0 °C)	C6_OUT (20.0 °C)	HOT	(none)	×
C8	C8_IN (120.0 °C)	C8_OUT (260.0 °C)	COLD	(none)	×
C7	C7_IN (100.0 °C)	C7_OUT (200.0 °C)	COLD	(none)	×

+ Add logical stream

Fonte: elaborado pelo próprio autor.

Figura 09 – Parâmetros globais e continuação para próxima etapa

Global parameters

Minimum approach temperature $\Delta T_{\min}$ [K] Minimum temperature difference allowed between hot and cold sides at any point of a heat exchanger.

Save config & continue Saved. You may now proceed to Tab 3.

Fonte: elaborado pelo próprio autor.

- Na segunda tabela, pode-se escolher os pares preferenciais de troca térmica na quinta coluna (*Preferred Pair*) e/ou adicionar mais correntes em + *Add logical stream*. Aqui deve-se fazer a modificação de $\Delta T_{\min}$ também se for necessário;
- Clicar em *Save config & continue* para ir para próxima etapa, conforme Figura 09, assim que a topologia inicial estiver conforme necessária;
- Ir para a próxima página *Run Synthesis*, conforme Figura 10. Escolher a Camada que se quer trabalhar (recomendado Layer 2 (Camada 2) que é a forma completa de síntese);

Figura 10 – Página de sintetização da rede

3. Run Synthesis

Choose the algorithm layer and run. The network is generated as a new DWSIM flowsheet ready to open in the simulator.

Algorithm

☐ Layer 1 - Fixed pairs heuristic
Each preferred hot/cold pair gets one heat exchanger; residual sensible duty falls to a heater or cooler. Simple and predictable. May leave room for improvement when one hot still has heat available where another cold still needs it.

☒ Layer 2 - Fixed pairs + cross-match **Recommended**
Same as Layer 1, then greedy cross-match: pairs of (residual cooler in a hot) + (residual heater in a cold) are inspected and, if feasible within ΔT_{min} , a cross heat exchanger is added to reduce both utilities at once.

Run synthesis

Fonte: elaborado pelo próprio autor.

8. Clicar em *Run Synthesis* e aguardar de 1 a 2 minutos. Assim que concluído, verificar os resultados imediatos, conforme Figura 11. Aqui o mais importante é o erro (*Error*) e o status (*Status*) da simulação, que indica convergência;

Figura 11 – Informações gerais e de convergência da síntese

Run synthesis ✓ Layer 2 done - validation OK

Results

Q recovered: **7260.32** kW

Hot utility (heaters): **1627.87** kW

Cold utility (coolers): **6419.96** kW

Heat exchangers: **2**

Utility blocks: **4**

Validation (target T vs DWSIM-solved T)

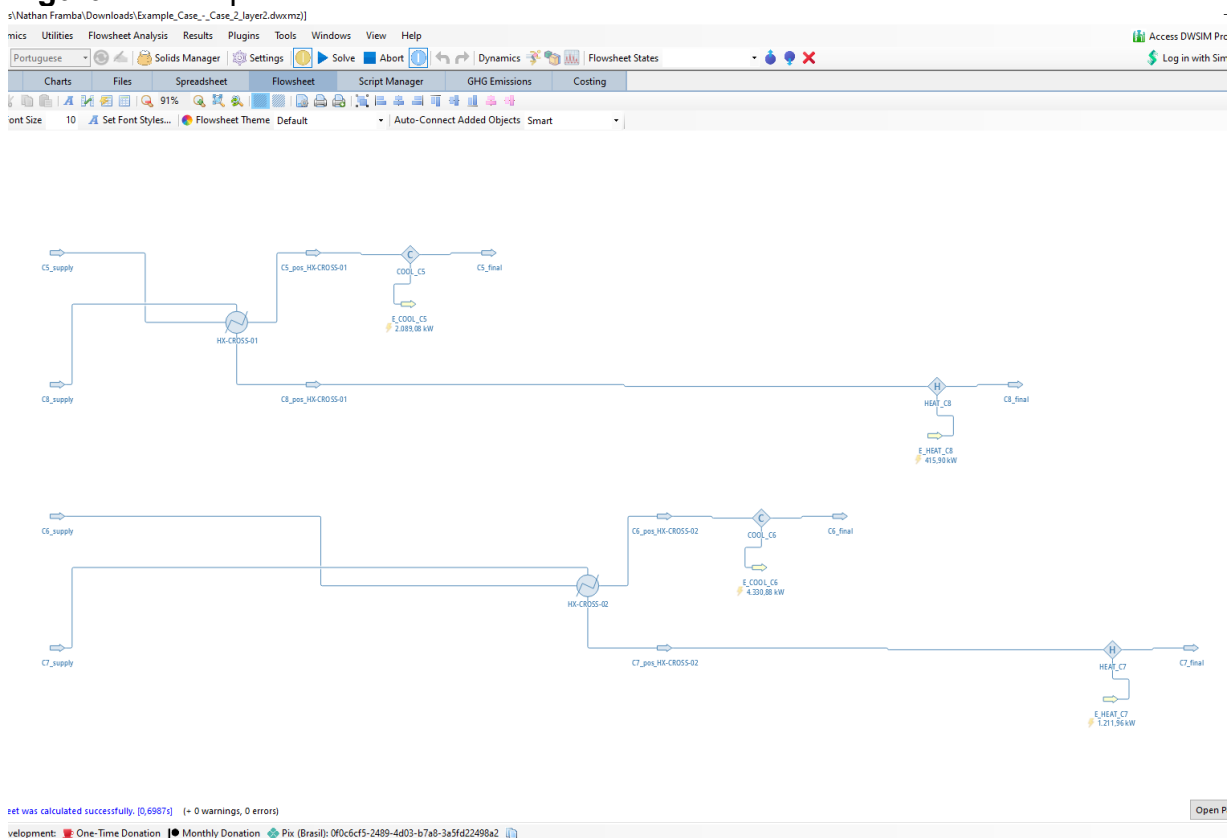
Stream	T expected [°C]	T calculated [°C]	Error [K]	Status
C5	60.00	60.00	+0.000	OK
C6	20.00	20.00	+0.000	OK
C8	260.00	260.00	+0.000	OK
C7	200.00	200.00	+0.000	OK

Download generated .dwxmz Open the file in DWSIM to inspect the synthesized heat exchanger network.

Fonte: elaborado pelo próprio autor.

9. Para se obter o *flowsheet* DWSIM, basta clicar em *Download generated .dwxmz*, resultando em um arquivo separado do original com a rede pronta, conforme Figura 12;

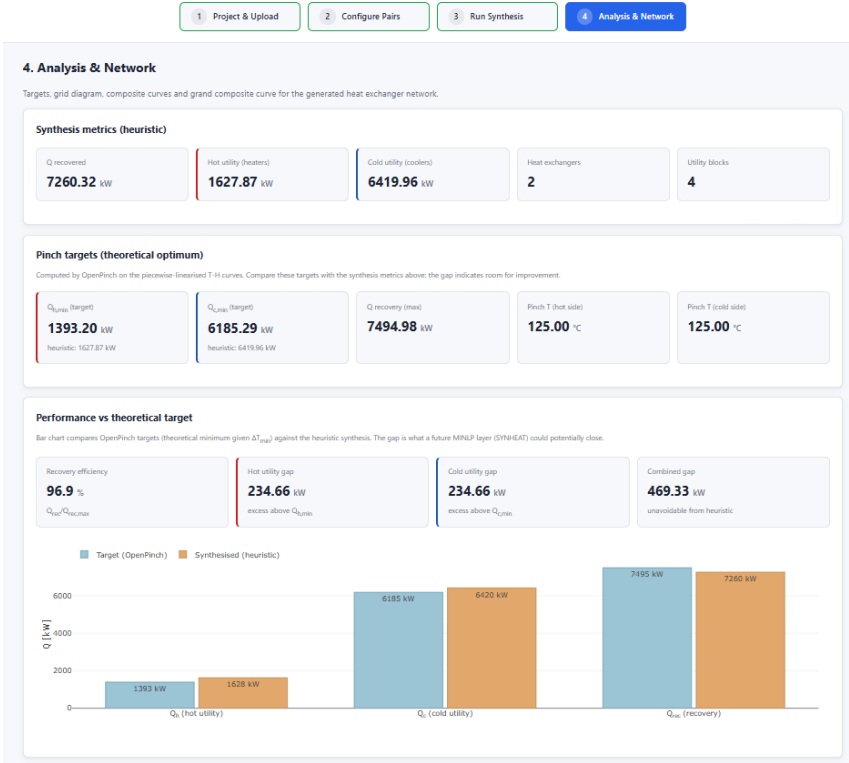
Figura 12 – Arquivo DWSIM da rede de trocadores de calor sintetizada



Fonte: elaborado pelo próprio autor.

10. A próxima página, *Analysis & Network*, traz todas as informações da rede criada: os calores e número de trocadores (Figura 13), os valores encontrados teoricamente pelo OpenPinch (Figura 13), a performance comparativa teórico vs. sintetizado (Figura 13), o perfil de entalpia-temperatura de todas as correntes (Figura 14), o diagrama de trocadores de calor (Figura 14), as curvas compostas (Figura 15), as informações dos trocadores de calor (Figura 15) e as informações dos aquecedores e resfriadores (Figura 15).

Figura 13 – Informações de calor e eficiência da rede sintetizada



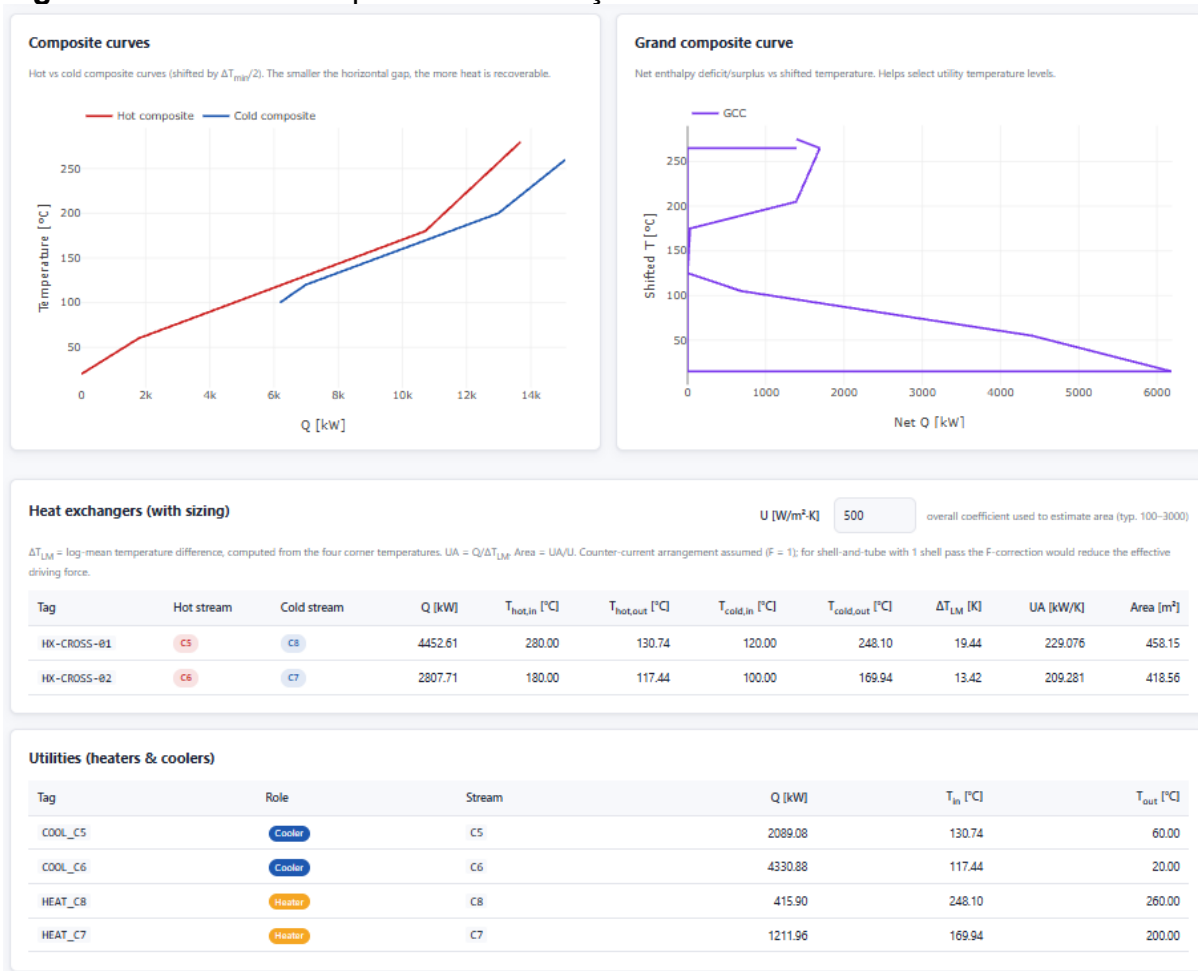
Fonte: elaborado pelo próprio autor.

Figura 14 – Perfis T-H e rede formada na síntese



Fonte: elaborado pelo próprio autor.

Figura 15 – Curvas compostas e informações dos trocadores sintetizados



Fonte: elaborado pelo próprio autor.

5 RESULTADOS

O Quadro 01 apresenta o conjunto completo de informações relativos às correntes dos cinco casos, com exceção de composição. O Quadro 2 apresenta os alvos termodinâmicos calculados pelo OpenPinch e os consumos efetivos pelo aplicativo.

Quadro 01 – Especificações das correntes

ID	Caso	Tipo	T _{supply} (°C)	T _{target} (°C)	$\dot{m}$ (kg/h)	Q _{tot} (kW)
C1	Case 1	Quente	175	45	34279	1292,8
C2	Case 1	Quente	125	65	35575	2391,6
C3	Case 1	Fria	20	155	34497	2845,8
C4	Case 1	Fria	40	112	36011	1099,2
C5	Case 2	Quente	280	60	35814	6541,7
C6	Case 2	Quente	180	20	35503	7138,6
C8	Case 2	Fria	120	360	36273	8375,6
C7	Case 2	Fria	100	200	35647	4019,7
C9	Case 3	Quente	353	313	3679	391,3
C10	Case 3	Quente	347	246	3902	284,3
C11	Case 3	Quente	255	80	3651	1067,1
C12	Case 3	Fria	224	340	3505	836,7
C13	Case 3	Fria	116	313	3575	126,0
C15	Case 3	Fria	40	293	3618	460,6
C14	Case 3	Fria	50	113	3660	454,3
H1	Case 4	Quente	128	60	101	3,9
H2	Case 4	Quente	140	65	14657	494,9
H3	Case 4	Quente	209	65	7000	639,8
C1	Case 4	Fria	70	130	37000	1147,2
C2	Case 4	Fria	128	193	22817	839,6
H1	Case 5	Quente	85	45	262202	6252,0
H2	Case 5	Quente	120	40	28771	4000,0
H3	Case 5	Quente	125	35	14900	2151,0
H4	Case 5	Quente	56	46	997561	12500,4
H5	Case 5	Quente	90	86	1184860	6000,3
H6	Case 5	Quente	225	75	77201	7500,0
C1	Case 5	Fria	40	55	824046	7000,4
C2	Case 5	Fria	55	65	989645	6000,0
C3	Case 5	Fria	65	465	279421	18000,0
C4	Case 5	Fria	10	170	70063	13008,0

Fonte: elaborado pelo próprio autor.

Quadro 02 – Comparativo entre alvos termodinâmicos calculados e consumos sintetizados na camada 2

Caso	Correntes	Q _{h,mín OP} (kW)	Q _{h sint.} (kW)	Q _{c,mín OP} (kW)	Q _{c sint.} (kW)	Gap (%)
Case 1	4	369,4	1462,1	108,8	1201,4	295,8
Case 2	4	4900,3	5116,9	6185,3	6401,9	4,4
Case 3	7	222,9	445,4	87,9	310,4	99,8
Case 4	5	1008,49	1932,5	160,28	244,7	91,6
Case 5	10	15927,3	20011,7	10322,6	14407,0	25,6

Fonte: elaborado pelo próprio autor.

No Quadro 02, o Gap (%) é a porcentagem considerando Q_{h,mín} como referência.

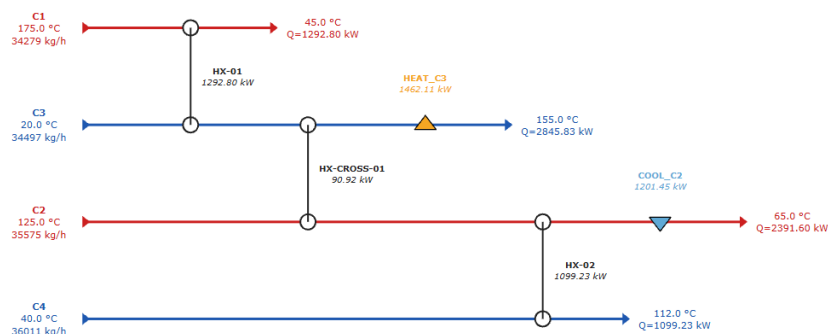
5.1 Case 1 – Quatro correntes sensíveis (Peng-Robinson)

O Case 1 envolve quatro correntes sensíveis: duas quentes (C1 e C2) e duas frias (C3 e C4), modeladas com Peng-Robinson. A análise pinch determina a temperatura de Pinch como $T_{\text{pinch}} = 120,0 \text{ }^{\circ}\text{C}$, com alvos $Q_{h,\text{mín}} = 369,4 \text{ kW}$ e $Q_{c,\text{mín}} = 108,8 \text{ kW}$ para $\Delta T_{\text{mín}} = 10 \text{ K}$. Essas correntes foram modeladas conforme o exemplo de Shenoy (exemplo 1), do artigo *Benchmark solutions for small heat exchanger networks*.²⁸, tal que as correntes C1, C2, C3 e C4 sejam análogas às correntes H1, H2, C1 e C2 do exemplo, respectivamente. A síntese da camada 1 produziu dois trocadores principais: HX-01 entre C1 e C3, transferindo todos os 1292,8 kW disponíveis na C1; e HX-02 entre C2 e C4, transferindo 1099,2 kW. A camada 2 introduziu adicionalmente o trocador HX-CROSS-01 entre C2 (cooler residual) e C3 (heater residual), recuperando 90,9 kW adicionais. O aquecedor HEAT_C3 cedeu 1462,11 kW, enquanto o resfriador COOL_C2 recebeu 1201,45 kW. O esquema de trocadores de calor pode ser visto na Figura 16.

Figura 16 – Rede de trocadores de calor do Case 01

HEN grid diagram

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.

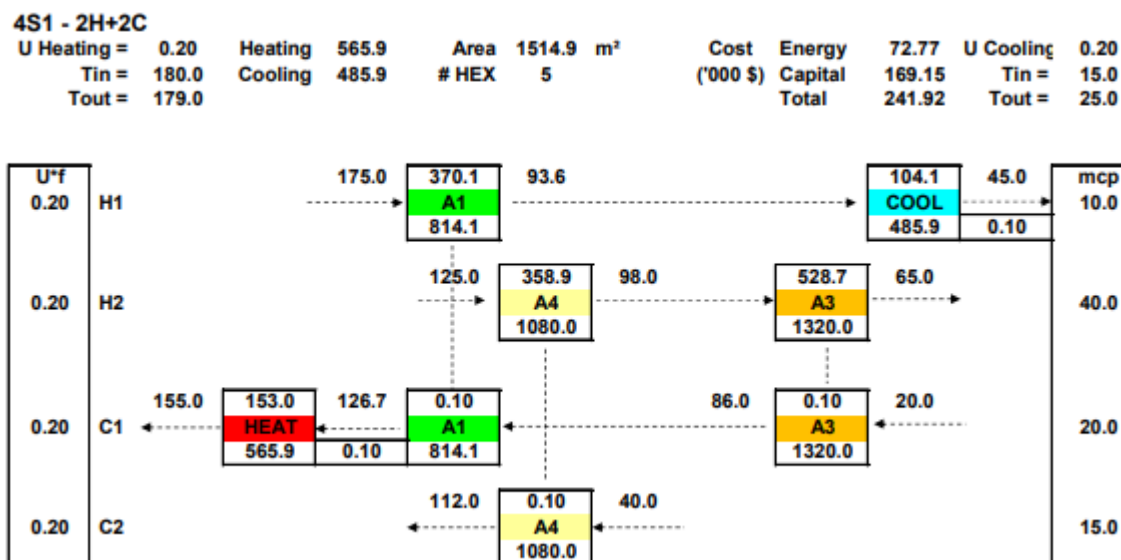


Fonte: elaborada pelo próprio autor.

O consumo total sintetizado pela camada 2 atinge $Q_h = 1462.1$ kW e $Q_c = 1201.4$ kW, significativamente acima dos alvos teóricos, representando uma eficiência de 69,4%. Esse gap reflete uma limitação estrutural da heurística de pares fixos: uma vez que a topologia inicial é fixada pelas preferências do usuário, recuperações cruzadas que exigiriam reordenação completa das correntes não são exploradas. O gap absoluto Q_h sintetizado - $Q_{h,mín} = 1093$ kW representa, em uma análise econômica, o custo da decisão de manter a topologia preferencial fornecida pelo usuário em vez de delegar inteiramente a topologia ao otimizador. A validação ponto-a-ponto das temperaturas-alvo confirmou erro máximo inferior a 0,1 K em todas as correntes, atestando a consistência termodinâmica do flowsheet gerado.

A Figura 17 apresenta o modelo ótimo de rede de trocadores de calor no exemplo de origem. Deve-se notar que o mesmo número de trocadores de calor (incluindo aquecedor e resfriador), foi escolhido, com a diferença sendo as correntes associadas a esses trocadores. A otimização do artigo, que considera como um dos grandes fatores o custo econômico que o simulador não possui, parte do princípio do resfriador estar associado à corrente H1, algo que o aplicativo não permite delimitar, apenas permitindo a escolha de pares entre correntes.

Figura 17 – Rede otimizada sem divisões de corrente segundo literatura



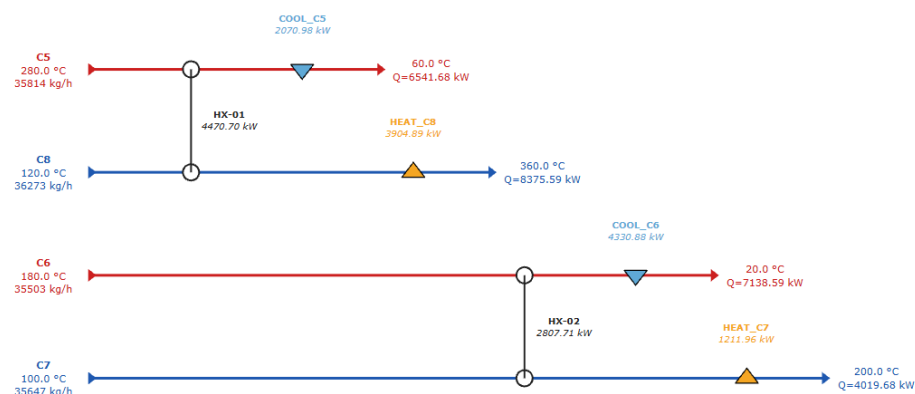
Fonte: DECLERQ, 2018.

5.2 Case 2 – Síntese de Amônia (Raoult modificado)

O Case 2 envolve quatro correntes do circuito de síntese de amônia, com cinco componentes (N_2 , H_2 , NH_3 , Ar , CH_4), modeladas com pacote de Raoult modificado. As temperaturas envolvidas vão de 20 °C a 360 °C, com vazões mássicas elevadas (≈ 35 t/h cada). A análise pinch identifica $T_{pinch} = 125,0$ °C, com alvos $Q_{h,mín} = 4900,3$ kW e $Q_{c,mín} = 6185,3$ kW para $\Delta T_{mín} = 20$ K. A presença de NH_3 em condições próximas à saturação introduz a região bifásica que motivou o aparato de entalpia rigorosa, e a linearização RDP com $\epsilon = 1$ K produziu em média doze segmentos por corrente fásica — número adequado para capturar o glide bifásico sem proliferação excessiva. Essas correntes foram modeladas conforme o exemplo de Trivedi et al. (exemplo 4), do artigo *Benchmark solutions for small heat exchanger networks*.²⁴, tal que as correntes C5, C6, C7 e C8 sejam análogas às correntes H1, H2, C1 e C2 do exemplo, respectivamente. A síntese gerou dois trocadores de calor: HX-01, transferindo 4470,7 kW entre C5 e C8, e HX-02, transferindo 2807,7 kW entre C6 e C7. Não houve cross-match adicional devido à incompatibilidade térmica entre os pares residuais. Foram gerados também dois aquecedores — HEAT_C8, fornecendo 3904,89 kW para C8, e HEAT_C7, fornecendo 1211,96 kW para C7 —, assim como dois resfriadores — COOL_C5, recebendo 2070,98 kW de C5, e COOL_C6, recebendo 4330,88 kW de C6. A rede pode ser vista na Figura 18.

Figura 18 - Rede de trocadores de calor do Case 2**HEN grid diagram**

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.

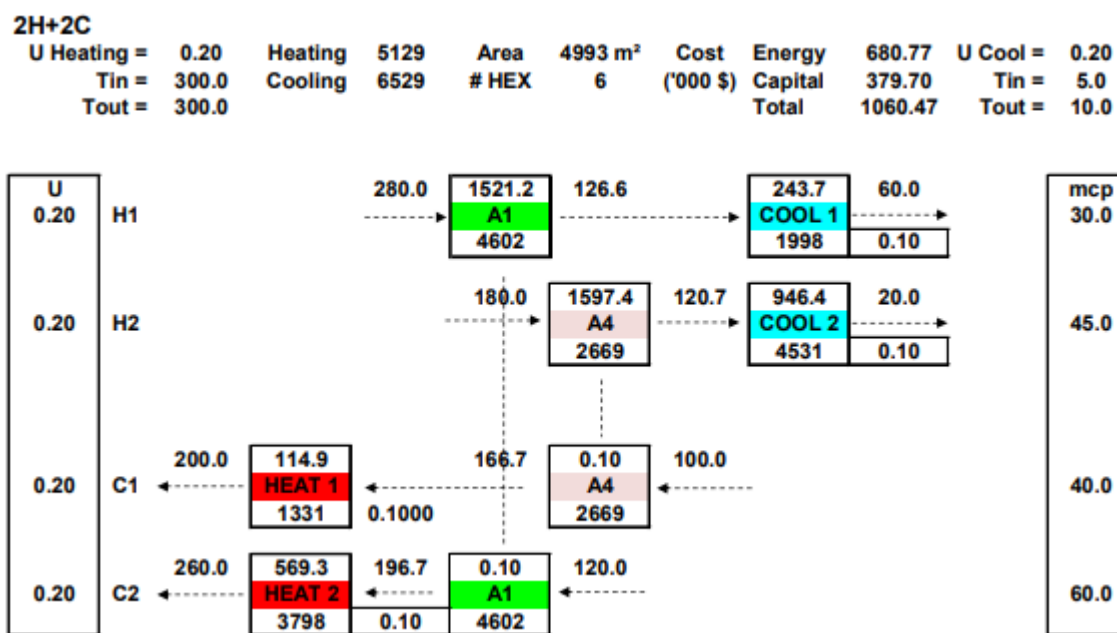


Fonte: elaborada pelo próprio autor.

O consumo total sintetizado foi $Q_h = 5116.9$ kW e $Q_c = 6401.9$ kW, este último apenas 3,5 % acima do alvo $Q_{c,mín}$, demonstrando que a topologia preferencial original já era termodinamicamente próxima do ótimo para a refrigeração, que é uma situação coerente com o fato de a indústria de síntese de amônia já incorporar décadas de heurísticas de integração energética em seu projeto. Isso também indica que a simulação a partir do exemplo original foi assertiva com a realidade. A performance do simulado em relação à teórica para esse caso foi de 97,1%.

A Figura 19 apresenta o modelo ótimo de rede de trocadores de calor no exemplo de origem que mais se aproximou do resultado obtido. Deve-se notar a corrente C8 foi modelada com temperatura final de 360 °C, diferentemente da corrente C2 original de 260 °C, entretanto isso não se mostrou um aspecto relevante pois em um segundo teste apenas mudando essa temperatura a eficiência subiu para 97,2% e a rede se manteve essencialmente igual em estrutura, mudando apenas as quantidades de calor trocadas. O modelo de redes do artigo utiliza a mesma estrutura da rede simulada, inclusive com as mesmas trocas entre correntes, de forma que até os calores trocados pelos trocadores sejam similares. Isso significa que o aplicativo conseguiu chegar a uma rede de trocadores de calor ótima, conforme era esperado pela sua eficiência, devido, em grande parte, à topologia original já próxima da ótima.

Figura 19 - Rede otimizada do caso A do exemplo



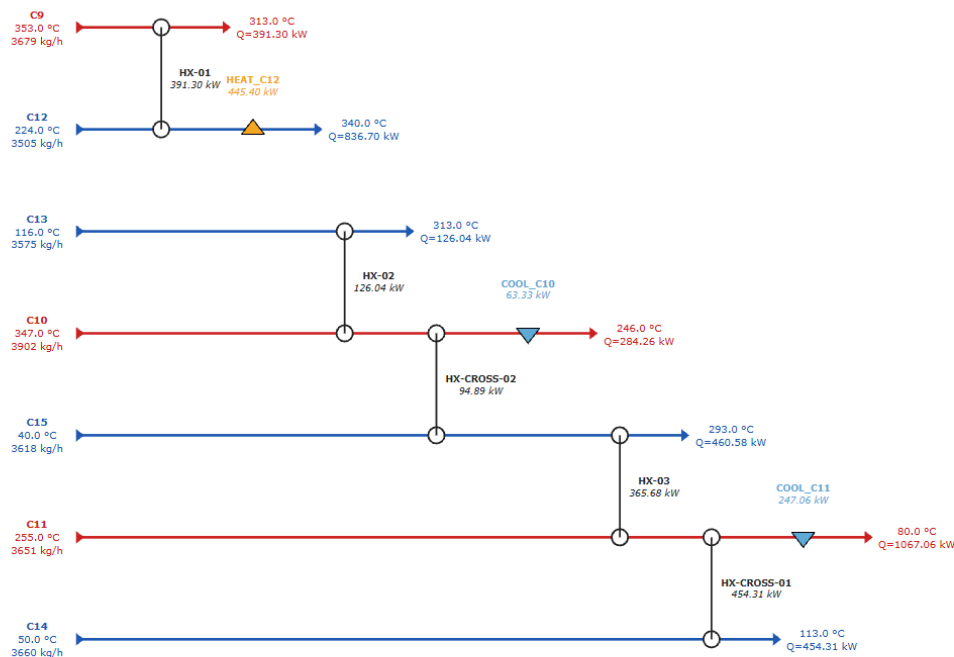
Fonte: DECLERQ, 2018.

5.3 Case 3 – Sistema multifluxo de sete correntes

O Case 3 é o mais complexo dos testes iniciais: sete correntes (três quentes, quatro frias) envolvendo Dowtherm A, água e hidrocarbonetos leves, com temperaturas entre 40 °C e 353 °C e vazões da ordem de 3,5 t/h. A análise pinch identifica $T_{\text{pinch}} = 229$ °C, com $Q_{h,\text{mín}} = 222,9$ kW e $Q_{c,\text{mín}} = 87,9$ kW $\Delta T_{\text{mín}} = 10$ K. Essas correntes foram modeladas conforme o exemplo de Juan Diego em seu código *Método de punto de pliegue*²⁵, usado como inspiração e base para a escrita da parte do código que calcula os dados da análise Pinch. A síntese da camada 2 gerou cinco trocadores (três da camada 1 e dois cross-matches), além de três blocos de utilidade residual: HX-01 trocando 391,3 kW entre C9 e C12, HX-02 trocando 126,04 kW entre C10 e C13, HX-03 trocando 365,68 kW entre C11 e C5, HX-CROSS-01 trocando 454,31 kW entre C11 e C14, HX-CROSS-02 trocando 94,89 kW entre C10 e C15, HEAT_C12 doando 445,4 kW para C12, COOL_C10 absorvendo 63,33 kW de C10 e COOL_C11 absorvendo 247,06 kW de C11. A rede pode ser vista na Figura 20.

Figura 20 - Rede de trocadores de calor do Case 3**HEN grid diagram**

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.

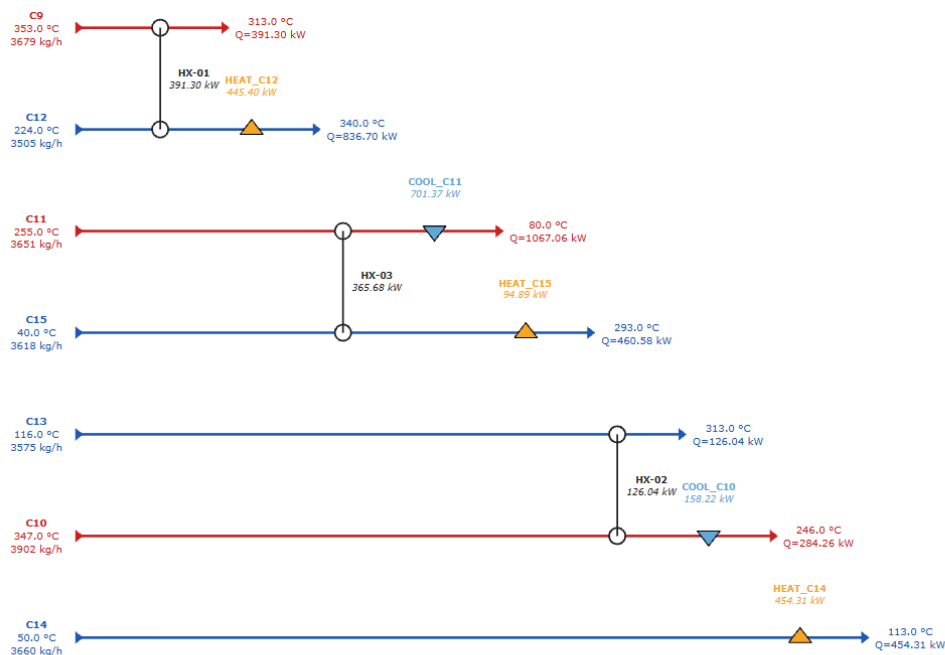
**Fonte:** elaborada pelo próprio autor.

Esse Case não possui uma rede otimizada para comparação, mas pode ser um caso de comparação entre os métodos de síntese do aplicativo, usando apenas a camada 1 ou incluindo a camada 2. A Figura 21 mostra a rede formada com a camada 1 apenas, com uma eficiência (performance vs teórico) de 53,4%, em comparação com a eficiência de 86,6% com a inclusão da camada 2. Ele apresenta o melhor gap observado nos três casos para a utilidade quente: o consumo $Q_h = 445,4$ kW está apenas 99,8 % acima do alvo termodinâmico 222,9 kW. A camada 2 recuperou conjuntamente entre seus dois trocadores cross-match 549,2 kW, eliminando completamente o aquecedor da C14 e o resfriador da C9. O cross-match HX-CROSS-01 é especialmente significativo: recupera mais calor do que qualquer um dos trocadores convencionais da camada 1, mostrando o potencial da segunda etapa heurística em problemas com mais de quatro correntes.

Figura 21 - Rede de trocadores de calor do Case 3 apenas com a Camada 1

HEN grid diagram

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.

**Fonte:** elaborada pelo próprio autor.

5.4 Case 4 – Fracionamento de ácidos graxos

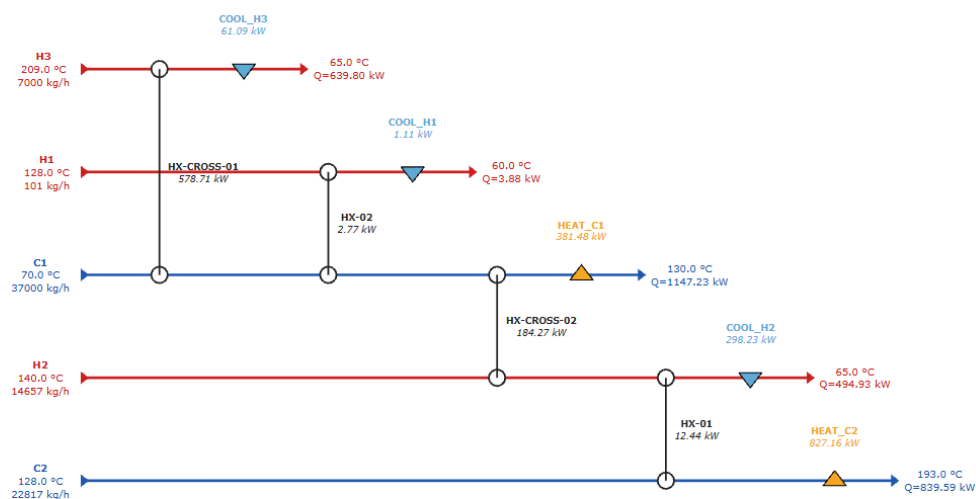
O Case 04 envolve cinco correntes com cinco componentes estipulados de um processo de fracionamento de ácidos graxos (ácidos mirístico, palmítico, esteárico, oleico e linoleico), modeladas com Peng-Robinson. A análise pinch determina a temperatura de Pinch como $T_{\text{pinch}} = 75,0 \text{ }^{\circ}\text{C}$, com alvos $Q_{\text{h.mín}} = 1008,49 \text{ kW}$ e $Q_{\text{c.mín}} = 160,28 \text{ kW}$ para $\Delta T_{\text{mín}} = 10 \text{ K}$. Essas correntes foram modeladas conforme o artigo de Abu Bakar et al: *Initial Analysis on Heat Exchanger Networks of Fatty Acid Fractionation Plant to Optimize Energy Recovery and Controllability*²⁶. A síntese da camada 1 produziu dois trocadores principais: HX-01 entre H2 e C2, transferindo 12,44 kW, e HX-02 entre H1 e C1, transferindo 2,77 kW. A camada 2 introduziu o trocador HX-CROSS-01 entre C1 e H3, recuperando 578,71 kW e o trocador HX-CROSS-02 recuperando 184,27 kW. O aquecedor HEAT_C1 cedeu 381,48 kW, o aquecedor HEAT_C2 827,16 kW, enquanto o resfriador COOL_H1

recebeu 1,11 kW, o resfriador COOL_H2 298,23 kW e o resfriador COOL_H3 61,09 kW. O esquema de trocadores de calor pode ser visto na Figura 22.

Figura 22 - Rede de trocadores de calor do Case 4

HEN grid diagram

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.

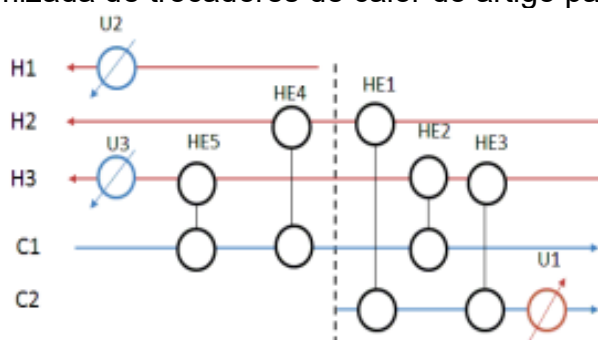


Fonte: elaborada pelo próprio autor.

A eficiência atingida foi de 79,5%. Foi utilizada 225% da utilidade fria teórica e 125% da utilidade quente teórica. A Figura 22 mostra as relações determinadas automaticamente pelo aplicativo, que resultaram nesses resultados de eficiência, novamente demonstrando que existe uma limitação estrutural da heurística de pares fixos por causa da topologia inicial ser fixada pelas preferências do usuário.

A Figura 23 apresenta o modelo ótimo de rede de trocadores de calor no exemplo de origem. Deve-se notar que o artigo decide usar um número maior de trocadores de calor do que o aplicativo determina, com dois resfriadores, apenas um aquecedor e cinco trocadores de calor.

Figura 23 – Rede otimizada de trocadores de calor do artigo para $\Delta T_{\min} = 10\text{ }^{\circ}\text{C}$



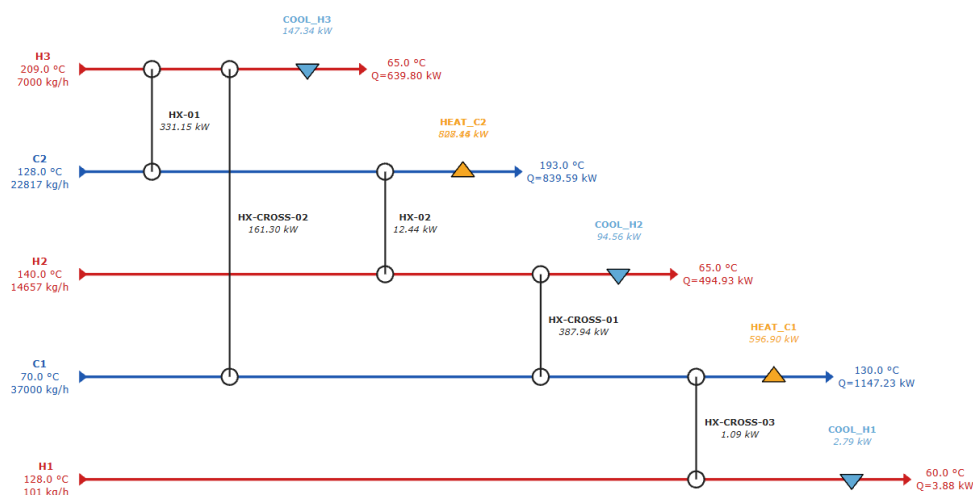
Fonte: ABU BAKAR, 2015.

A eficiência de 79,5% foi atingida estabelecendo como pares preferenciais para as correntes H1, H2 as correntes C1 e C2, respectivamente, com a corrente H3 sem par preferencial. Para tentar forçar uma aproximação do caso original, foram estabelecidos os pares preferenciais: H1-none, H2-C2, H3-C2, C1-H3 e C2-H2. Isso resultou em uma eficiência de 91,4%, com todas as correntes necessitando de utilidades e cinco trocadores de calor: três HX e dois HX-CROSS, conforme a Figura 24.

Figura 24 - Rede de trocadores de calor do Case 4 após mudança na Camada 1

HEN grid diagram

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.



Fonte: elaborada pelo próprio autor.

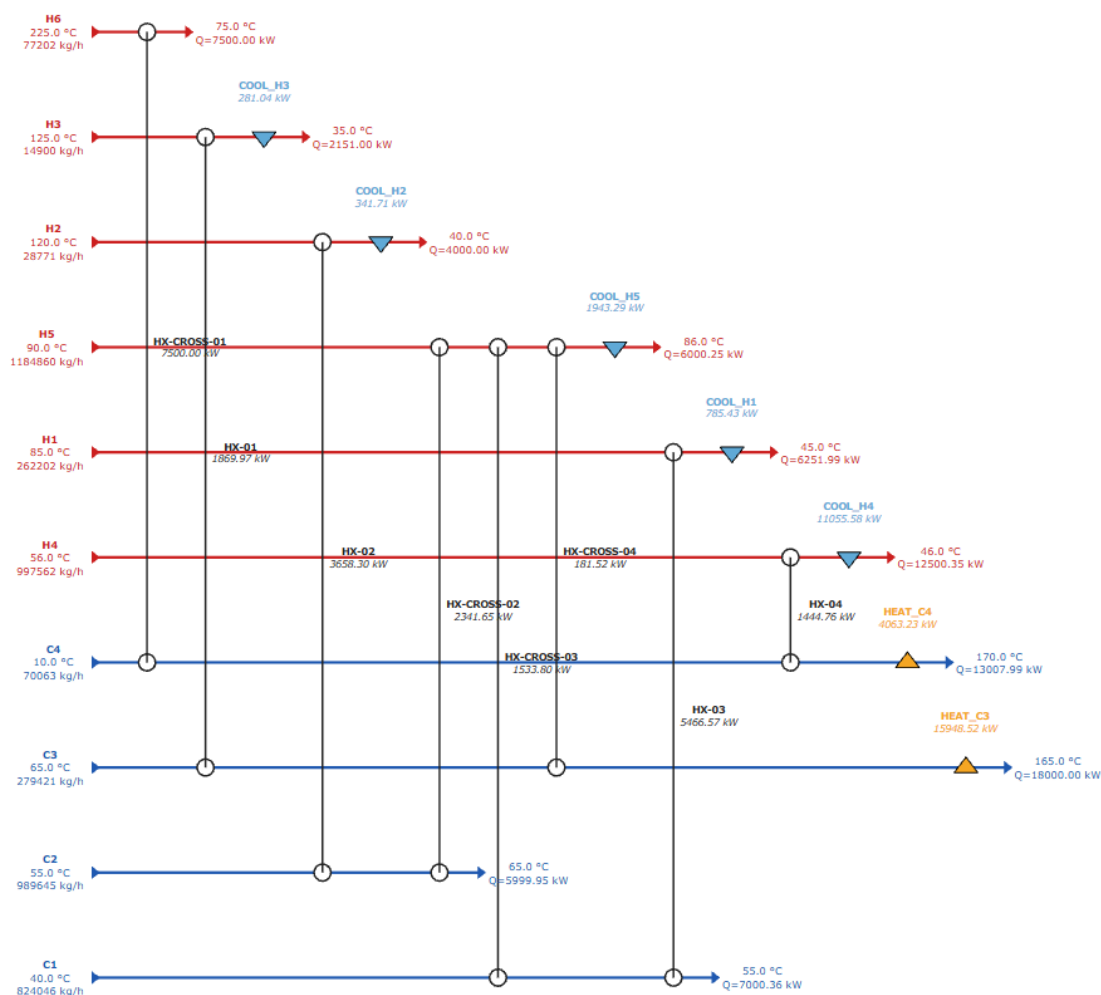
5.5 Case 5 - Simulação de Petroquímica com 10 correntes

O Case 5 conta como maior número de correntes dos testes, 10, com a complexidade adicionada de algumas correntes serem bifásicas, modelado com Peng-Robinson. Esse Case foi criado com base no Case 18 de Daniel Declecq, que por sua vez usa como base o caso de Ahmad (1985)²⁷. A simulação no aplicativo contou com 8 trocadores de calor, 4 da camada 1 e 4 da camada 2, 5 resfriadores e 2 aquecedores. O esquema pode ser visto na Figura 25. O Quadro 03 mostra as relações de correntes trocadores e energia.

Figura 25 - Rede de trocadores de calor do Case 5

HEN grid diagram

Each row is a process stream. Vertical bars are heat exchangers. Triangles are utility blocks (heater / cooler). Numbers show duty in kW.



Fonte: elaborada pelo próprio autor.

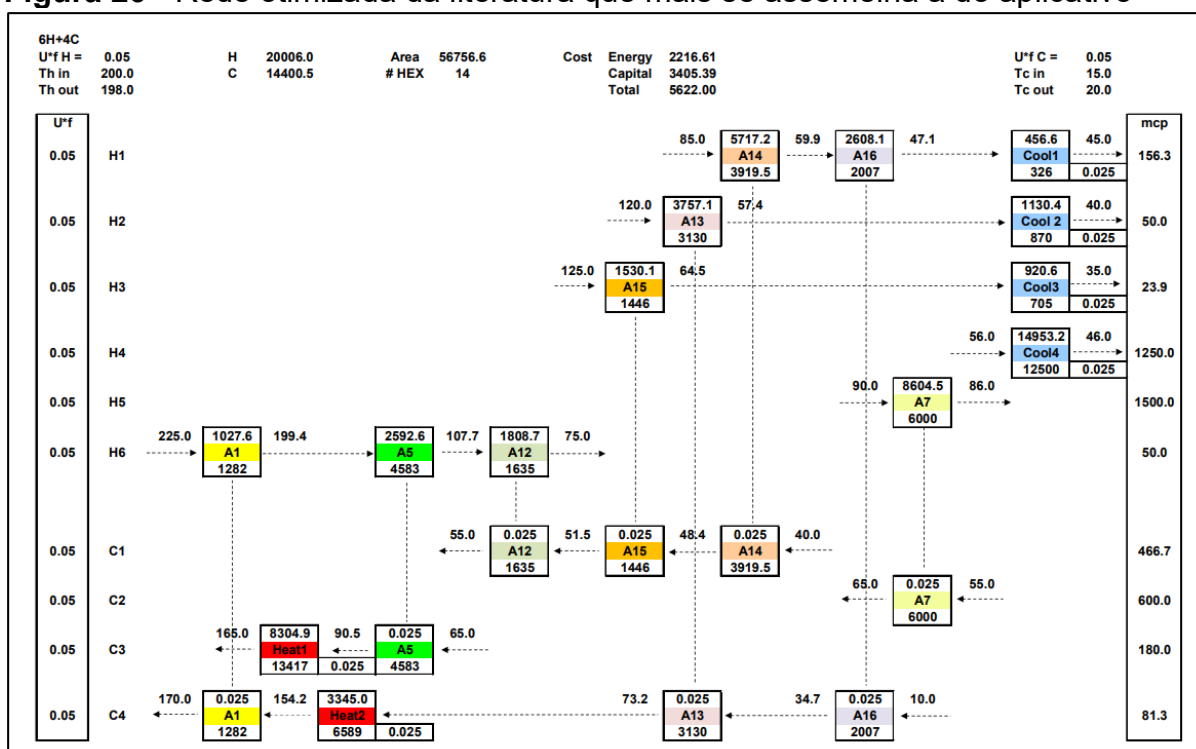
Quadro 03 – Métodos de obtenção dos materiais registrados na literatura

Trocador	Camada	Correntes (Quente-Fria)	Calor Trocado (kW)
HX-01	1	H3-C3	1869,97
HX-02	1	H2-C2	3658,30
HX-03	1	H1-C1	5466,57
HX-04	1	H5-C3	181,52
HX-CROSS-01	2	H6-C4	7500,00
HX-CROSS-02	2	H5-C2	2341,65
HX-CROSS-03	2	H5-C1	1533,80
HX-CROSS-04	2	H5-C3	181,52
COOL_H1	-	H1	785,43
COOL_H2	-	H2	341,71
COOL_H3	-	H3	281,04
COOL_H4	-	H4	11055,58
COOL_H5	-	H5	1943,29
HEAT_C3	-	C3	15948,52
HEAT_C3	-	C4	4063,25

Fonte: elaborado pelo próprio autor.

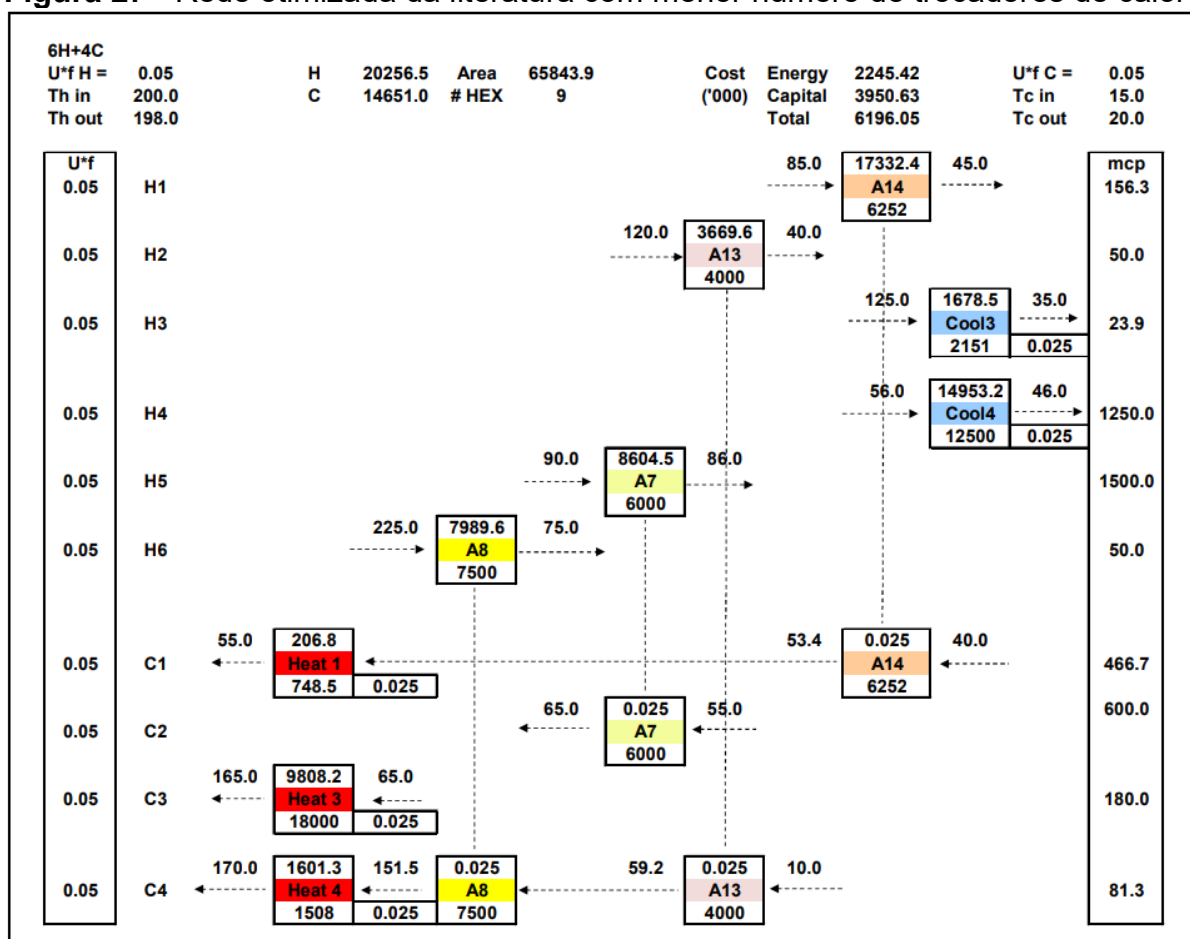
A análise pinch determina a temperatura de Pinch como $T_{\text{pinch}} = 51^{\circ}\text{C}$, com alvos $Q_{\text{h.mín}} = 15927,33 \text{ kW}$ e $Q_{\text{c.mín}} = 10322,63 \text{ kW}$ para $\Delta T_{\text{mín}} = 10 \text{ K}$. A eficiência atingida foi de 85,5%, com a utilidade quente utilizada sendo 126% da teórica e a fria sendo 140%. O caso mais comparável do artigo base é o de 14 trocadores de calor, sendo deles 4 resfriadores e 2 aquecedores, conforme a Figura 26.

Figura 26 - Rede otimizada da literatura que mais se assemelha à do aplicativo



Fonte: DECLERQ, 2015.

Alguns testes de repriorização de correntes (configuração de pares) foram feitos para tentar melhorar a eficiência de recuperação energética, mas sem grande sucesso. Mesmo uma tentativa de “forçar” a rede de trocadores da Figura 27 resultou em uma rede de 7 trocadores de calor e 7 utilidades com 88,8% de eficiência, mas com desvios consideráveis na validação de temperaturas, o que inviabiliza esse teste como uma possível rede. Percebe-se nesse Case que quanto maior o número de correntes, mais difícil se torna “ajustar” o chute inicial de pares, mesmo tendo um guia a ser seguido. Confirma-se aqui a necessidade de uma terceira camada ou de uma segunda etapa no aplicativo para *retrofit* de uma rede já criada.

Figura 27 - Rede otimizada da literatura com menor número de trocadores de calor

Fonte: DECLERQ, 2015.

5.6 Discussão geral dos casos

Baseado no comportamento dos três primeiros casos, é possível ver que existe de fato uma diferença entre a síntese heurística e os alvos termodinâmicos e ela varia fortemente conforme a qualidade da topologia preferencial e do potencial de cross-match inerente de cada caso. O Case 1, com apenas dois pares dados pelo usuário, atinge *gap* elevado entre heurística e alvos porque as correntes preferenciais não estão alinhadas ao pinch identificado pelo OpenPinch, isso é confirmado no Case 4 com a melhora significativa de eficiência (de 79,5% para 91,4%) com a mudança de topologia da Camada 1. O Case 2, onde as preferências do usuário coincidem aproximadamente com o ótimo termodinâmico, atinge *gap* pequeno (~3 % no Q_c) sem necessidade de cross-match. O Case 3, onde a camada 2 atua efetivamente, atinge o melhor compromisso entre número de trocadores (cinco) e consumo de utilidades.

O Case 5 serve como prova de fogo de todos os testes, possui o maior número de correntes e algumas delas são bifásicas, representando exatamente o caso que esse trabalho pretende resolver: um caso em que C_p constante não é aceitável e que o trabalho manual é grande devido ao número de correntes. Apesar de seu resultado não ter sido tão positivo quanto o do Case 2, tanto sua eficiência quanto seu *Gap* são aceitáveis em consideração à sua complexidade. Ele também serve como mais uma prova da limitação do método heurístico por causa sua da dificuldade de otimizar ainda mais iterativamente pela topologia da Camada 1.

Esse padrão é consistente com a literatura clássica (FURMAN; SAHINIDIS, 2002): a heurística por pares apresenta *gap* variável e pode falhar quando o usuário não fornece preferências consistentes com o pinch, isto é, quando o estabelecimento de pares inicial não for apropriado¹⁶. A camada 2 do PinchCraft mitiga essa limitação ao buscar cross-matches automaticamente, mas não pode reordenar a topologia herdada da camada 1, os pares pré-estabelecidos. Para *gaps* inferiores a 1 %, é necessária a passagem a uma terceira camada baseada em programação inteira-mista não-linear.

Quanto à robustez do encadeamento, a validação termodinâmica ponto-a-ponto convergiu em todos os cinco casos, com erro máximo de 0,027 K observado no Case 3 (corrente C13). Esse erro é numericamente desprezível e atribuível ao próprio solver Newton–Raphson interno do DWSIM, não às discrepâncias na metodologia de síntese. O *bug* previamente relatado de contaminação por estado global foi efetivamente mitigado pela arquitetura de subprocesso descartável, ao custo de aproximadamente quinze segundos por execução. Em nenhum dos cinco casos foi necessária intervenção manual sobre o fluxograma gerado, em algum deles foram apenas feitos testes de otimização, mas não por problemas com a criação da rede.

6 CONCLUSÕES

Este trabalho apresentou a fundamentação teórica, a formulação matemática, a implementação computacional e a validação experimental do **PinchCraft**, sistema síntese automática de redes de trocadores de calor acoplado ao simulador de processos DWSIM. A motivação central foi remover a hipótese de capacidade calorífica constante para permitir correntes com mudança de fase e o esforço manual associado à transcrição de dados entre simulador e ferramentas de *targeting*. Ambos os objetivos bem sucedidos.

Três contribuições caracterizam o estado atual do sistema. Primeiro, a robustez do acoplamento Python–DWSIM, que contornou um conjunto significativo de obstáculos técnicos previamente catalogados: limitações da Fluent API em pythonnet 3.0.5, contaminação por estado global do simulador, exigências específicas das portas de *EnergyStream* em heaters e coolers, replicação canônica de pacotes termodinâmicos. Segundo, a adoção sistemática de entalpia rigorosa em todas as etapas que envolvem balanço de calor, com o algoritmo Ramer–Douglas–Peucker como ponte matematicamente controlada entre a representação contínua e a discreta exigida pelo PTA. Terceiro, a entrega automatizada de um produto utilizável — o flowsheet DWSIM resolvido e validado, o que torna a ferramenta complementar ao ambiente de simulação em que o engenheiro já trabalha.

A validação em cinco casos de complexidade variável confirmou a convergência do flowsheet de saída com erro inferior a 0,1 K em todas as temperaturas-alvo, e demonstrou ganhos de recuperação de calor de até 5,4% entre as camadas heurísticas 1 e 2 do sistema. O gap entre síntese e alvos termodinâmicos varia entre 3,5% (Case 2, em que a topologia preferencial já é próxima do ótimo) e valores significativamente maiores em casos em que a topologia preferencial não está alinhada ao pinch (Case 1), que é uma limitação intrínseca da heurística por pares fixos. Isso justifica a evolução para uma terceira camada de síntese baseada em programação inteira-mista não-linear. Os resultados, no entanto, até o momento são satisfatórios e funcionais e cumprem com o proposto do trabalho.

7 SUGESTÕES PARA TRABALHOS FUTUROS

A presente versão do PinchCraft consolida o pipeline básico de extração rigorosa, *targeting* e síntese heurística. Sobre essa base, articula-se uma agenda de extensões organizadas por prioridade técnica e impacto esperado.

7.1 Camada 3: Síntese por Programação Matemática

A extensão técnica mais relevante é a implementação de uma terceira camada de síntese baseada na superestrutura SYNHEAT de Yee e Grossmann (1990)¹⁵, formulada como problema MINLP a estágios. A camada 3 deverá admitir divisão de correntes, otimização simultânea de áreas e seleção automática de *matches*. A integração com o OpenHENS (também do programa Ahuora) é a via natural, dado o já existente acoplamento com o OpenPinch. O custo estimado é da ordem de uma a duas semanas de trabalho para a implementação inicial, mais a calibração dos resolvedores (GEKKO com APOPT, IPOPT ou Couenne como *back-ends* MINLP candidatos). A topologia das camadas 1/2 do PinchCraft pode servir como *warm start* do MINLP, reduzindo o tempo de convergência em problemas grandes. Essa medida deve resolver o maior problema atual da ferramenta, que é a limitação do método heurístico.

7.2 Múltiplas Utilidades e Custo Associado

A estrutura UtilitySchema do PinchCraft já contempla a definição de múltiplas utilidades por nível térmico e custo unitário, mas atualmente apenas vapor e ar de resfriamento são pré-configurados. A extensão para vapor de baixa, média e alta pressão (LP, MP, HP), água de resfriamento, fluidos térmicos e gases criogênicos será apoiada por seleção automática de níveis a partir da *Grand Composite Curve* segundo as regras de Linnhoff e Vredeveld (1984)². O custeio anualizado total, combinando utilidades e capital, também passará a integrar a função objetivo da síntese,

permitindo a comparação direta entre topologias com mesmo Q_{rec} mas distribuições diferentes de utilidades.

7.3 Heurística incremental de layout para $n > 8$ correntes

A otimização combinatória por força bruta do *layout* deixa de ser viável para mais de oito correntes. Uma heurística incremental, baseada em *graph drawing* por camadas e iterativa por trocas adjacentes (estilo *barycenter heuristic* de Sugiyama), substituiria a força bruta com custo computacional polinomial e qualidade visual próxima do ótimo.

7.4 Validação por casos industriais reais

A última linha de continuidade é a aplicação do PinchCraft a casos industriais reais, com dados reais de correntes de processo, algo que foi notado de ser difícil de encontrar em literatura sobre integração energética. Tal validação permitirá calibrar parâmetros heurísticos, identificar nichos de aplicação em que a ferramenta agregue valor diferencial frente às alternativas comerciais e gerar conjuntos de *benchmarks* públicos para a comunidade acadêmica e industrial brasileira.

8 REFERÊNCIAS

¹LINNHOFF, B.; FLOWER, J. R. Synthesis of heat exchanger networks: I. Systematic generation of energy optimal networks. *AIChE Journal*, v. 24, n. 4, p. 633–642, 1978. DOI: <https://doi.org/10.1002/aic.690240411>.

²LINNHOFF, B.; VREDEVELD, D. R. Pinch technology has come of age. *Chemical Engineering Progress*, v. 80, n. 7, p. 33–40, 1984.

³LINNHOFF, B.; TOWNSEND, D. W.; BOLAND, D.; HEWITT, G. F.; THOMAS, B. E. A.; GUY, A. R.; MARSLAND, R. H. *A user guide on process integration for the efficient use of energy*. Rugby: Institution of Chemical Engineers, 1982.

⁴KEMP, I. C. *Pinch analysis and process integration: a user guide on process integration for the efficient use of energy*. 2. ed. Oxford: Butterworth-Heinemann, 2007. ISBN 978-0-7506-8260-2.

⁵SMITH, R. *Chemical process design and integration*. 2. ed. Chichester: John Wiley & Sons, 2016. ISBN 978-1-119-99014-7.

⁶WESTPHALEN, D. L.; WOLF MACIEL, M. R. Pinch analysis based on rigorous physical properties. *Brazilian Journal of Chemical Engineering*, v. 16, n. 3, p. 277–284, 1999. DOI: <https://doi.org/10.1590/S0104-66321999000300006>.

⁷MEDEIROS, D. W. O. *DWSIM — open source process simulator*. Versão 8.x. 2024. Disponível em: <https://dwsim.org>. Acesso em: 11 junho 2026.

⁸WALMSLEY, T. G. *OpenPinch — an open-source Python library for advanced pinch analysis and total site integration*. Hamilton: University of Waikato (Ahuora Programme), 2025. Disponível em: <https://github.com/waikato-ahuora-smart-energy-systems/OpenPinch>. Acesso em: 11 junho 2026.

⁹HOHMANN, E. C. *Optimum networks for heat exchange*. 1971. Ph.D. dissertation. University of Southern California, Los Angeles.

¹⁰LINNHOFF, B.; HINDMARSH, E. The pinch design method for heat exchanger networks. *Chemical Engineering Science*, v. 38, n. 5, p. 745–763, 1983. DOI: [https://doi.org/10.1016/0009-2509\(83\)80185-7](https://doi.org/10.1016/0009-2509(83)80185-7).

¹¹LINNHOFF, B. Pinch analysis: a state-of-the-art overview. *Chemical Engineering Research and Design*, v. 71, n. A5, p. 503–522, 1993.

¹²KLEMEŠ, J. J.; KRAVANJA, Z. Forty years of heat integration: pinch analysis (PA) and mathematical programming (MP). *Current Opinion in Chemical Engineering*, v. 2, n. 4, p. 461–474, 2013. DOI: <https://doi.org/10.1016/j.coche.2013.10.003>.

¹³PAPOULIAS, S. A.; GROSSMANN, I. E. A structural optimization approach in process synthesis — II: Heat recovery networks. *Computers & Chemical Engineering*, v. 7, n. 6, p. 707–721, 1983. DOI: [https://doi.org/10.1016/0098-1354\(83\)85023-6](https://doi.org/10.1016/0098-1354(83)85023-6).

¹⁴FLOUDAS, C. A.; CIRIC, A. R.; GROSSMANN, I. E. Automatic synthesis of optimum heat exchanger network configurations. *AIChE Journal*, v. 32, n. 2, p. 276–290, 1986. DOI: <https://doi.org/10.1002/aic.690320215>.

¹⁵YEE, T. F.; GROSSMANN, I. E. Simultaneous optimization models for heat integration — II. Heat exchanger network synthesis. *Computers & Chemical Engineering*, v. 14, n. 10, p. 1165–1184, 1990. DOI: [https://doi.org/10.1016/0098-1354\(90\)85010-8](https://doi.org/10.1016/0098-1354(90)85010-8).

¹⁶FURMAN, K. C.; SAHINIDIS, N. V. A critical review and annotated bibliography for heat exchanger network synthesis in the 20th century. *Industrial & Engineering Chemistry Research*, v. 41, n. 10, p. 2335–2370, 2002. DOI: <https://doi.org/10.1021/ie010389e>.

¹⁷TRIVEDI, K. K.; O'NEILL, B. K.; ROACH, J. R.; WOOD, R. M. A new dual-temperature design method for the synthesis of heat exchanger networks. *Computers & Chemical Engineering*, v. 13, n. 6, p. 667–685, 1989. DOI: [https://doi.org/10.1016/0098-1354\(89\)80006-0](https://doi.org/10.1016/0098-1354(89)80006-0).

¹⁸HASAN, M. M. F.; JAYARAMAN, G.; KARIMI, I. A.; ALFADALA, H. E. Synthesis of heat exchanger networks with nonisothermal phase changes. *AIChE Journal*, v. 56, n. 4, p. 930–945, 2010. DOI: <https://doi.org/10.1002/aic.12031>.

¹⁹KAMATH, R. S.; BIEGLER, L. T.; GROSSMANN, I. E. Modeling multistream heat exchangers with and without phase changes for simultaneous optimization and heat integration. *AIChE Journal*, v. 58, n. 1, p. 190–204, 2012. DOI: <https://doi.org/10.1002/aic.12565>.

²⁰RAMER, U. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, v. 1, n. 3, p. 244–256, 1972. DOI: [https://doi.org/10.1016/S0146-664X\(72\)80017-0](https://doi.org/10.1016/S0146-664X(72)80017-0).

²¹DOUGLAS, D. H.; PEUCKER, T. K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, v. 10, n. 2, p. 112–122, 1973. DOI: <https://doi.org/10.3138/FM57-6770-U75U-7727>.

²²RAMÍREZ, S. *FastAPI — modern, fast, web framework for building APIs with Python*. 2018. Disponível em: <https://fastapi.tiangolo.com>. Acesso em: 11 junho 2026.

²³PENG, D.-Y.; ROBINSON, D. B. A new two-constant equation of state. *Industrial & Engineering Chemistry Fundamentals*, v. 15, n. 1, p. 59–64, 1976. DOI: <https://doi.org/10.1021/i160057a011>.

²⁴DECLERCQ, Daniel. *Benchmark solutions for small heat exchanger networks*. [S. l.]: Pinchco, 2018. Disponível em: [Pinchco PDF](#). Acesso em: 10 jun. 2026.

²⁵JDSR1. *mpunto-plieue*. GitHub. Disponível em: <https://github.com/jdsr1/mpunto-plieue>. Acesso em: 11 junho 2026.

²⁶ABU BAKAR, S. H.; ABDUL HAMID, M. K.; WAN ALWI, S. R.; MANAN, Z. A. *Initial Analysis on Heat Exchanger Networks of Fatty Acid Fractionation Plant to Optimize Energy Recovery and Controllability*. **Chemical Engineering Transactions**, v. 45, p. 97–102, 2015. DOI: 10.3303/CET1545017. Disponível em: <https://www.cetjournal.it/index.php/cet/article/view/CET1545017>. Acesso em: 11 junho 2026.

²⁷DECLERCQ, Daniel. *Insight in heat exchanger network synthesis: the ten streams literature case from Ahmad (1985) – Case 18*. [S. l.]: Pinchco, 13 set. 2015. Disponível em: <https://www.pinchco.com/images/site/case18->

[the10streamsliteraturecaseofahmad\(1985\)-2015_09_13.pdf](#). Acesso em: 10 jun. 2026.

ANEXO A – REGISTRO DE PROGRAMA DE COMPUTADOR: PINCHCRAFT



REPÚBLICA FEDERATIVA DO BRASIL
MINISTÉRIO DO DESENVOLVIMENTO, INDÚSTRIA, COMÉRCIO E SERVIÇOS
INSTITUTO NACIONAL DA PROPRIEDADE INDUSTRIAL
DIRETORIA DE PATENTES, PROGRAMAS DE COMPUTADOR E TOPOGRAFIAS DE CIRCUITOS

Certificado de Registro de Programa de Computador

Processo Nº: **BR512026003720-4**

O Instituto Nacional da Propriedade Industrial expede o presente certificado de registro de programa de computador, válido por 50 anos a partir de 1º de janeiro subsequente à data de 22/05/2026, em conformidade com o §2º, art. 2º da Lei 9.609, de 19 de Fevereiro de 1998.

Título: DWSIM Pinch Plugin - PinchCraft

Data de publicação: 22/05/2026

Data de criação: 22/05/2026

Titular(es): NICOLAS SPOGIS; AI FOR TECHNOLOGY LTDA; NATHAN ALVES DE OLIVEIRA FRAMBA; ELIAS DE SOUZA MONTEIRO FILHO

Autor(es): NICOLAS SPOGIS; ELIAS DE SOUZA MONTEIRO FILHO; NATHAN ALVES DE OLIVEIRA FRAMBA

Linguagem: VISUAL BASIC; JAVA SCRIPT; PYTHON; C#

Campo de aplicação: FQ-07; IF-01; IN-01; IN-02; IN-03; IN-04; IN-05; MT-01

Tipo de programa: AP-01; AP-03; AT-01; CD-01; FA-04; GI-01; IA-01; IA-02; IA-03; IT-01; PD-05; SM-01; SO-02; SO-04; SO-07; TC-01; UT-06

Algoritmo hash: SHA-512

Resumo digital hash:

8cfcfe52f862967f64fcc94e9fea8800e789ca14e36def086b5852a42af0c15f8b0e35a81855e94c31c034887dc4f622a53641c5d4109d9a8a5edb81ae7dcf

Expedido em: 02/06/2026

Aprovado por:

ERICA GUIMARAES CORREA

Chefe da Divisão de Programas de Computador e Topografias de Circuitos Integrados