Computers and Chemical Engineering 180 (2024) 108474

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Comparing optimization schemes for solving case studies with multiple heat exchangers using high-order pinch point temperature difference methods

Eivind Brodal *, Steven Jackson, Oddmar Eiksund

UiT – The Arctic University of Norway, Tromsø 9037, Norway

ARTICLE INFO

Keywords:
Optimization
Heat pump
Heat exchanger
High-order
Interpolation

ABSTRACT

Heat exchangers (HEs) are often modeled using pinch point temperature difference (ΔT_{pinch}) methods when optimizing systems with HEs. However, even small inaccuracies in model predictions of HEs will introduce numerical noise that can cause optimization algorithms to fail. A recent study of single HEs suggests that high-order interpolation methods can compute ΔT_{pinch} much faster than conventional methods. However, the performance of such methods when optimizing HE systems have not previously been tested.

Heat pumps with 2 and 3 HEs, with and without an ejector are optimized using different schemes. Results from these case studies show that non-linear constrained gradient-based optimization algorithms are more than 5 times faster than particle swarm (PS), and that the conventional genetic algorithm (GA) should not be used. However, the main conclusion is that the case study optimizations are solved 5–10 times faster if ΔT_{pinch} is calculated using hybrid high and low-order interpolation methods.

1. Introduction

Optimization of systems with multiple heat exchangers (HEs) is used in a large variety of disciplines (Kemp, 2007), e.g., to find the best heat exchanger network (HEN) with respect to minimization of HE area, capital expenses or energy efficiency of the whole system. The term 'HEN optimization' is typically used for the mixed-integer problem arising when studying how to best transfer heat between hot and cold streams (Kim and Bagajewicz, 2016). In the present work, the heat streams are matched a priori using a fixed process design without a HE superstructure. Hence, a more classical process optimization is conducted, more closely related to the simultaneous heat integration (SHI) approach by Duran and Grossmann (1986). However, many of the methods investigated here, such as the HE models, can also be applied directly in HEN optimization. The review by Austbø et al. (2014) identified 186 optimization articles just related to liquefied natural gas (LNG) refrigeration processes. The main idea is often that process alternatives are best compared if they are modelled using components with real equipment performance, and otherwise operated with optimal processes parameters such as optimal pressures and working fluid mixtures. SHIs and HENs can be large systems only including HEs. However, such systems can also include other types of equipment (Allen et al., 2009), where classical heat integration problems must be extended to include both heat and work (Fu and Gundersen, 2016). Heat pumps, refrigeration cycles and steam cycles can be considered as special cases of different SHIs where it is often possible to increase the energy efficiency by adding more HEs that improve heat integration between hot and cold streams (Brodal et al., 2019).

The optimization of SHIs can be considered to consist of two main parts: the optimization algorithm that is used to find optimal process parameters; and the process model which is solved to find the relevant process performance indicators in each optimization step. Both parts can present complex problems in their implementation since many different optimization algorithms and numerical process models can be applied. Also, the performance of optimization algorithms can inter-depend on the accuracy of the process model, especially in systems with HEs which can be difficult to model accurately. Optimization studies have typically focused on finding optimal solutions, instead of investigating how such problems are best solved. However, it is important to identify fast optimization schemes since such problems require significant computational effort, e.g., sensitivity studies can easily require a month of computer run time (Brodal et al., 2019). The main goal in this work is to

* Corresponding author.
E-mail address: eivind.brodal@uit.no (E. Brodal).

https://doi.org/10.1016/j.compchemeng.2023.108474
Received 27 January 2023; Received in revised form 20 October 2023; Accepted 21 October 2023
Available online 21 October 2023
0098-1354/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

find efficient schemes for optimizing SHIs using pinch point temperature difference to model HE performance.

Modeling HEs with increased accuracy generally requires increased computational work. When optimizing SHIs, however, this extra effort can reduce the required number of optimization steps and thereby improve the optimization scheme and the overall workload. Multiple HE models exist for different designs such as detailed plate-fin and shell-and-tube models. For example, the literature survey by Ayub et al. (2019) lists 22 different heat transfer models just for evaporation in plate HEs, and the review article by Eldeeb et al. (2016) describes 17 evaporation and 8 condensation models for plate HEs. Such models can be used in optimization studies (Rao et al., 2020), but more often they are considered too complex and difficult to solve in this type of study. Simplified HE models, such as the pinch point temperature difference approach, exclude all information about HE geometry and materials, and have gained much attention since they require far less numerical computations. The pinch point is the HE region that operates with the smallest temperature difference. The pinch point region requires a large portion of the HE area since heat transfer is proportional to the temperature difference (ΔT). Hence, the main assumption in pinch analysis is that the size of $\Delta T_{\text{pinch}} = \min(\Delta T)$ is directly related to total HE area and cost. However, this also indicates that pinch analysis can be highly inaccurate if the pinch point region has a similar temperature difference as in the rest of the HE. For HEs with smaller ΔT_{pinch}, the pinch point is more likely to require a larger portion of the HE area. The theoretical best HE processes, with zero (heat transfer) entropy production at the pinch point, have $\Delta T_{\text{pinch}} = 0$. Note that such theoretical processes require infinite HE areas; and that this can exclusively be explained by the pinch point. A fast pinch point HE modeling approach is particularly convenient when solving optimization problems, since HE models must be solved in each optimization step. This is especially true when optimizing complex SHIs involving many process calculations and optimization steps. Pinch analysis was introduced in the late 1970s (Linnhoff and Flower, 1978). Since then, a large variety different SHIs have been optimized using ΔT_{pinch} models, e.g., the review article by Austbø et al. (2014) identified 15 different articles related to the LNG industry. Pinch analysis has also been used to model industrial heat pump processes (Allen et al., 2009), water-to-water heat pump systems (Murr et al., 2011), CO_2-based trans-critical and supercritical cycles (Ren, 2020), optimal fluid mixtures (Sarkar and Bhattacharyya 2009; Dai et al., 2015), optimal waste heat utilization (He et al., 2015), to improve complex LNG value chains processes (Bouabidi et al., 2021) and find energy efficient and economical gas-steam combined cycles (Li et al., 2022). Since a temperature cross in a HE is impossible, $\Delta T_{\text{pinch}} = 0$ can also be used to model theoretical best HE processes. Google Scholar identifies 3520 "pinch point temperature difference" documents, a doubling in only five years. However, this approach represents a highly simplified HE configuration, and is based on pure parallel flows. Pinch analysis is therefore not suitable for detailed (equipment level) design work but can be used to set the optimum starting point for this work. For example, Watson et al. (2015) used logarithmic mean temperature difference (LMTD) to model HE area A, as an extension to the pinch analysis. Elias et al. (2019) have also combined ΔT_{pinch} and LMTD modeling, and used this model to estimate HE area and cost.

The conventional ΔT_{pinch} approach is described in the book 'Pinch Analysis and Process Integration: A User Guide on Process Integration' by Kemp (2007) and the book chapter 'Pinch Point Analysis' by Dimian et al. (2014). Neither of these books mention high-order interpolation-based methods. To the authors knowledge, previous optimization studies have only found ΔT_{pinch} estimates based on the minimum temperature approach found using a fixed grid, i.e., without using any interpolation. It is perhaps natural to assume that there is no reason to solve simplified equipment models using numerical models with higher precision, however, ΔT_{pinch} approximations introduce numerical noise that can cause optimizations to be inaccurate or increase the overall workload and run time. Hence, numerical approximations with higher accuracy can help to speed-up the optimization process, and also improve the results since it is also reasonable to assume that detailed equipment level design work is improved (on average) if they are based on better process parameters less affected by random numerical approximations. Calculating ΔT_{pinch} with high accuracy is particularly important in comparative and sensitivity analyses. High-order (spectral) interpolation methods have been used for decades (Hesthaven et al., 2007) to solve partial differential problems (Trefethen 2000) and integrals (Shampine 2008). To the author's knowledge, high-order ΔT_{pinch} modeling was first introduced in 2023 (Brodal et al., 2023). This study showed that high-order methods can be 15 times faster than the conventional methods. Such high-order ΔT_{pinch} methods have previously only been applied to HEs with fixed inlet and outlet conditions, but not in SHI and HEN optimization studies. Hundreds of different HE temperature profiles are typically computed during an optimization, and the smoothness of these profiles depends on the accuracy of the fluid property packages. Since high-order methods are only successful when applied to smooth functions (Hesthaven et al., 2007), it is not obvious from the Brodal et al. (2023) study that such methods will improve optimization schemes. The aim of this study is to address this. Since the pinch point is typically identified by finding the minimum temperature approach ΔT at different grid points, the ΔT_{pinch} approximations are always less or equal the real ΔT_{pinch} value. That is, an unphysical ΔT_{pinch} value less than 0 K can be estimated as e.g., 2 K by an inaccurate approximation. Hence, optimizations based on constraints using inaccurate ΔT_{pinch} approximations will overestimate the HE performance, i.e., underestimate the entropy production during the heat transfer. There is also some randomness to the precision of the ΔT_{pinch} approximation. For example, the approximation often becomes more accurate if one grid point is close to the actual pinch point. Such random noise can break down the search strategy used by optimization algorithms, e.g., making it difficult to compute numerical gradients. Also, optimization algorithms will eventually find unphysical improvements with as large ΔT_{pinch} approximation errors as possible since this improves the HE performance. In the end, these methods are optimizing the ΔT_{pinch} approximation error, i.e., finding a process where the constraints are violated the most.

SHIs are often complex optimization problems to solve as discussed by Rao et al. (2020), which is a review article presenting an overview of advanced optimization algorithms. Rao et al. (2020) concludes that traditional optimization techniques, such as derivative based, usually fail to solve large-scale non-linear problems with local minima. It therefore states that more advanced methods, such as genetic algorithm (GA) and particle swarm (PS), must be used to overcome these problems. However, there are also many studies of complex SHIs where gradient-based algorithms have been used successfully. For example, the review article by Austbø et al. (2014) identified 15 gradient-based sequential quadratic programming (sqp) articles among 186 LNG optimization articles, but this is much less than the 29 GA-based articles identified in the same study. Gradient-based optimization algorithms are particularly sensitive to noise. Statistically based optimization schemes can be used if deterministic solvers fail, but such approaches are also affected by noise. Like all other SHI optimization literature found by the authors, the review article by Rao et al. (2020) focuses on optimization algorithms to explain how to best optimize SHIs. How the optimization results are affected by the accuracy of the process model is not even mentioned. SHI studies typically focus on finding the optimal configuration, however, a few studies have compared performance of optimization algorithms (Austbø et al., 2014; Rao et al., 2020), but neither of these studies investigated the best ΔT_{pinch} approximation. The reason for this is perhaps that SHI problems are complex and time consuming to optimize, which makes it hard to compare different approaches. SHI problems related to non-smoothness have been earlier discussed when solving optimal heat integration problems with multi-stream HEs, where

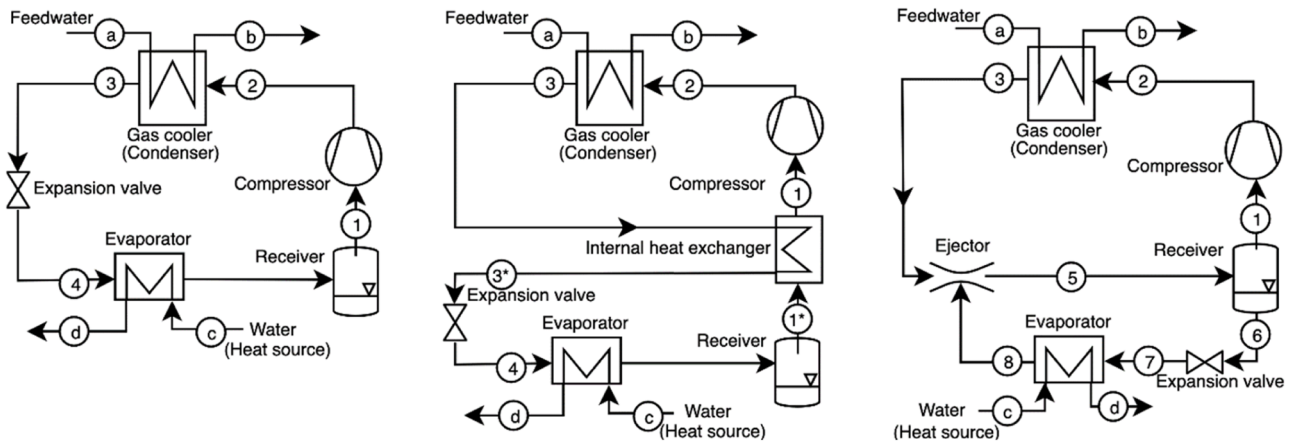

Fig. 1. Flow diagrams of heat pump A (left), heat pump B (middle) and heat pump C (right).

HE designs are allowed to change during the optimization. That is, new streams can be added, or previous streams can be removed, creating non-smooth processes schemes. Such problems are more complex and often simplified using constant heat capacity assumptions. For example, Fahr et al. (2022) discussed difficulties using gradient-based solvers for such problems and compared different optimization algorithms, using non-smooth, smooth, and mixed-integer pinch analysis formulations. An approach that avoids non-smooth functions has been proposed, but such schemes require additional continuous optimization variables (Cassanello et al., 2020). However, studies investigating smoothness related to systems only modeling HE with two streams, i.e., a fixed design, have not been found, even though approximations of such processes also can result in non-smooth problems due to numerical noise.

This study only looks at the modeling of HEs with two streams. To reduce complexity, HEs are modeled with the novel ΔT_{pinch} schemes previously recommended (Brodal et al., 2023), and only optimization algorithms available in MATLAB (2020) are tested, which include algorithms developed by Nelder and Mead (1965), Shanno (1970), Spellucci (1998), Kennedy and Eberhart (1995), Byrd et al. (1999) and Powell (2006). The main performance parameters in this article are optimization accuracy, success-rate and run time. The aim is to identify the best optimization algorithm and the best ΔT_{pinch} approximation method for optimizing heat pumps with 2 – 3 HEs, with or without an ejector. To the authors knowledge, this is the first optimization study to investigate high-order ΔT_{pinch} approximations, and also the first to systematically search for the overall best scheme with respect to optimization algorithm and HE approximation. That is, finding the best ΔT_{pinch} method and HE grid size.

2. Method

The method section includes a description of the heat pumps studied in this work. The details of the process and optimization models used are also explained, with the main focus being to describe the novel HE interpolation methods studied.

2.1. Process modeling

Models of SHIs include by definition heat exchangers, such as evaporators, condensers, and gas coolers, but can also include other components such as compressors, expansion valves and ejectors. Process models are calculated using fluid property packages, which include equation of state (EOS) models solved using numerical approximations. The fluid property package used in this article is named CoolProp (Bell et al., 2014), and implements a range of component specific EOSs with high accuracy, e.g., Span and Wagner (1996) for CO_2, Lemmon et al. (2009) for propane, Gao et al. (2020) for ammonia and Tillner-Roth and Baehr (1994) for R134a. In the process model developed, the performance of compressors, turbines and expansion valves is calculated directly from values calculated by the fluid property packages. Ejector performance can be modelled using an iterative method to find a set if conditions that coincides with the performance of a given ejector design. However, such numerical approximations generate noise. Iterative solvers can also be very time-consuming, since calculations have to be repeated in each optimization step (Brodal and Jackson 2019). In optimization studies, ejector performance can also be implemented through optimization constraints by increasing the dimension of the original optimization problem, i.e., without introducing additional numerical process noise (Brodal and Eiksund 2020). HEs can also be modeled through optimization constraints, however, HE constraints introduce noise since they are based on numerical approximations. The text below describes the heat pump designs and the equipment models in more detail.

2.1.1. Heat pump designs

Three different types of single-stage heat pumps are studied, as illustrated in Fig. 1. Heat pump A represents the simplest process with only the basic components. Heat pump B is modified with an additional internal suction gas HE, while heat pump C is modified with an ejector. Different working fluids (here referred to as refrigerants) are modeled for each heat pump. The heat exchanger where feedwater is warmed is sometimes operating as a gas cooler and other times as a condenser, depending on refrigerant and operating conditions such as the feedwater temperature.

2.1.2. Performance and equipment modelling

$$\text{COP} = \frac{P_{\text{heat}}}{P_{\text{comp}}} = \frac{h_2 - h_3}{h_2 - h_1}, \tag{1}$$

The coefficient of performance (COP) for heat pumps is defined as the ratio between the heating duty (P_{heat}) and the compressor duty (P_{comp}): where h is specific enthalpy. The numbering convention in Fig. 1 has been used. The refrigerant and feedwater mass flows ($\dot{m}_{\text{R}}$ and $\dot{m}_{\text{feedwater}}$) are calculated from the heating duty and process stream enthalpies:

$$\dot{m}_{\text{R}} = \frac{P_{\text{heat}}}{h_2 - h_3} \text{ and } \dot{m}_{\text{feedwater}} = \frac{P_{\text{heat}}}{h_{\text{b}} - h_{\text{a}}}. \tag{2}$$

For a system with a given design and equipment items, the set of process parameters that results in the most energy efficient operation ($\text{COP}_{\text{optimized}}$) is found using optimization schemes, as explained in Section 2.2.3. Equipment is described through performance parameters. Compressor performance is modelled using the isentropic efficiency parameter:

$$\eta_{\text{comp}} = \frac{P_{\text{comp,is}}}{P_{\text{comp}}} = \frac{h_{2,\text{is}} - h_1}{h_2 - h_1}, \tag{3}$$

where $P_{\text{comp,is}}$ is the power consumption of an isentropic compression process from state 1 to pressure p_2. The ejector is modeled using the ejector efficiency parameter introduced by Elbel and Hrnjak (2008), which compares the amount of work recovered by the ejector with the theoretical work that can be recovered with isentropic processes:

$$\eta_{\text{ejector}*} = \frac{\dot{m}_6\left(h_{8,\text{is}\to p_5} - h_8\right)}{\dot{m}_1\left(h_3 - h_{3,\text{is}\to p_5}\right)}, \tag{4}$$

where $h_{8,\text{is}\to p_5}$ and $h_{3,\text{is}\to p_5}$ are enthalpies obtained by isentropic processes from point 8 and 3 to the ejector outlet pressure (p_5). In optimization studies, it is often useful to compare different processes operating with the same ejector efficiency (η_{ejector}). An iterative search can be used to find a process pressure p_5 where the ejector operates with efficiency $\eta_{\text{ejector}*} \approx \eta_{\text{ejector}}$. However, this generates numerical noise, which can cause trouble for optimization algorithms. Finding p_5 accurately in each optimization step requires much computational effort (Brodal and Jackson 2019). Therefore, the ejector efficiency is instead implemented through optimization constraints while implementing the compressor pressure ratio $p_r = p_2/p_5$ as an optimization variable. The ejector efficiency ($\eta_{\text{ejector}*}$) for each optimization step process is calculated, and the ejector efficiency is implemented through optimization constraint $\eta_{\text{ejector}*} \leq \eta_{\text{ejector}}$ (Brodal and Eiksund 2020). Optimized results have the most energy efficient processes, i.e., the best allowed ejector efficiency: $\eta_{\text{ejector}*} \approx \eta_{\text{ejector}}$.

2.1.3. Heat exchanger modeling

HEs are typically the most time-consuming equipment to model since they must be calculated numerically using grid points and multiple fluid property evaluations. Implementing efficient HE models is therefore important in SHI and HEN optimization studies where the process model is evaluated in each step. The pinch point temperature difference is defined as $\Delta T_{\text{pinch}} = \min(\Delta T)$, where the temperature profile is the internal temperature differences between the warm heat source and the cold heat sink: $\Delta T = T_{\text{source}} - T_{\text{sink}}$. New high-order HE methods developed in the article Brodal et al. (2023) are applied. The temperature pinch ΔT_{pinch} is calculated for each HE, i.e., evaporator ($\Delta T_{\text{pinch evaporator}}$), gas cooler/condenser ($\Delta T_{\text{pinch gas cooler}}$) and IHE ($\Delta T_{\text{pinch IHE}}$). All HEs are assumed to operate in perfect counter current flow with zero pressure drop, which is a common simplification, e.g. used by Dai et al. (2015). To obtain a scheme where the temperature profile will be smooth functions, suitable for high-order interpolation methods, each HE is divided into sections (i) separated by internal bubble and dew points. The heating duty of HE section i is calculated: $\frac{\Delta P_i = \dot{m}_{\text{source}}}{dot} \cdot \Delta h_{\text{source},i}$, where $\Delta h_{\text{source},i}$ is the difference between the inlet and outlet enthalpies of the heat source in section i. The heating ratio $\Delta P_i / P$ is used to distribute N grid points evenly over each of the different HE sections, and the number of grid points in section i is named $N_{\text{grid},i}$. An equidistant distributed grid is often used, but not well suited for high-order interpolation methods (Hesthaven et al., 2007), where grid points should cluster at the ends, such as the Chebyshev grid distribution (Trefethen 2000):

$$x_i(j) = -\cos\left(\frac{\pi \cdot (j-1)}{N_{\text{grid},i} - 1}\right), \text{ for } j = 1, \ldots, N_{\text{grid},i}, \tag{5}$$

Internal enthalpies of the heat sink and heat source fluids in each HE section (i) are given by the linear transformations:

$$\begin{aligned} h_{\text{source},i}(j) &= \left(\frac{x_i(j)+1}{2}\right) \cdot \left(\frac{\Delta P_i}{\dot{m}_{\text{source}}}\right) + h_{\text{source},i,1} \text{ and } h_{\text{sink},i}(j) \\ &= \left(\frac{x_i(j)+1}{2}\right) \cdot \left(\frac{\Delta P_i}{\dot{m}_{\text{sink}}}\right) + h_{\text{sink},i,1}, \end{aligned} \tag{6}$$

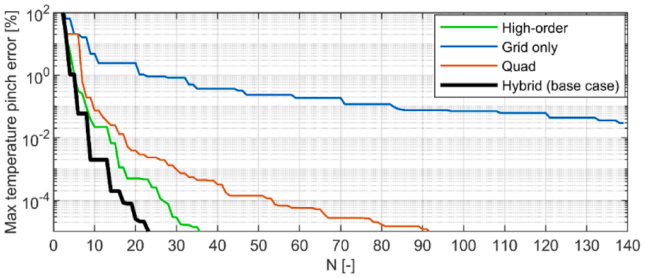

Fig. 2. ΔT_{pinch} errors using different HE models and grid sizes (N). (Data from Brodal et al. (2023).).

where the specific enthalpies $h_{\text{source},1,1}$ and $h_{\text{sink},1,1}$ are calculated directly from the inlet conditions. The temperatures $T_{\text{source},i}(j)$ and $T_{\text{sink},i}(j)$ are calculated using the properties package and the known pressures and enthalpies. The temperature profile is then found for the whole HE:

$$\Delta T_{i,j} = T_{\text{source},i}(j) - T_{\text{sink},i}(j). \tag{7}$$

The heating power related to each grid point is $P_i(j) = \dot{m}_{\text{source}} \cdot (h_{\text{source},i}(j) - h_2)$. Conventionally, the pinch point is found using an approximation only based on the grid values: $\Delta T_{\text{pinch}} = \min(\Delta T_{i,j})$. However, more accurate ΔT_{pinch} approximations can be used, as explained below.

High-Order ΔT_{pinch} Model

High-order (spectral) interpolations are here based on a Chebyshev grid distribution x_i, which is used to calculate the derivate of ΔT with respect to the heat transfer:

$$a_{i,j} = \left(\frac{d\ \Delta T}{dP}\right)_{i,j} = \left(\frac{dx}{dP} \cdot \frac{d\Delta T}{dx}\right)_{i,j} \approx \left(\frac{x_i(N_{\text{grid},i}) - x_i(1)}{P_i(N_{\text{grid},i}) - P_i(1)}\right) \cdot \mathbf{D}_{i,i} \Delta T_{i,j}, \tag{8}$$

where $x_i(N_{\text{grid},i}) - x_i(1) = 2$ from Eq. (5). The MATLAB function 'gallery' is used to calculate the $(N_{\text{grid},i} \times N_{\text{grid},i})$ spectral differentiation matrix $\mathbf{D}$, which can be calculated directly from an analytical expression derived from the behavior of Chebyshev polynomials (Trefethen 2000; Hesthaven et al., 2007). Pinch points are located at HE inlets and outlets, or internally at extremal points where the derivate of ΔT is zero. Such points are found using the cubic MATLAB function 'spline' to create a continuous polynomial function of the spectral derivative: $b_i = \text{spline}(\vec{P}_i, \vec{a}_i)$, while the extremal values $P_{i,\text{ex}}$ are found by solving $b_i(P_{i,\text{ex}}) = 0$ using the MATLAB function 'roots' and the 'coefs' and 'breaks' values created by the spline function:

$$\vec{P}_{i,\text{ex}} = \text{roots}(b_i \cdot \text{coefs}(j-1,:)) + b_i \cdot \text{breaks}(j-1), \text{ for } j = 2, \ldots, N_{\text{grid},i}. \tag{9}$$

To reduce run time, only regions where the derivative (a_i) is changing signs are investigated. Specific enthalpies $h_{i,\text{ex}}$ are calculated using Eq. (6), and CoolProp then finds $T_{\text{source},i,\text{ex}}$ and $T_{\text{sink},i,\text{ex}}$, using $h_{i,\text{ex}}$ and pressure as inputs. Finally, the pinch point is calculated as $\Delta T_{\text{pinch}} = \min(\Delta T_{i,j}, \Delta T_{i,\text{ex}})$, where $\Delta T_{i,\text{ex}} = T_{\text{source},i,\text{ex}} - T_{\text{sink},i,\text{ex}}$. This high-order approach was developed in (Brodal et al., 2023), where it was shown to be more stable and to obtain better precision than schemes based on pure high-order polynomial approximations.

Low-Order ΔT_{pinch} Model ('Quad')

A 1st-order approximation of the derivative can also be used to localize the pinch point. This approach is developed based on 2nd-order polynomial approximations for each set of three neighboring nodes values ($j-1, j, j+1$), for all HE section (i) using the MATLAB function 'polyfit'. Extremal points ($P_{i,\text{ex}}$) are roots of the derivative of these polynomial approximations, and are calculated using the MATLAB functions 'roots' and 'polyder'. $\Delta T_{i,\text{ex}}$ is found from $P_{i,\text{ex}}$, as explained above. This approach is named 'Quad'. A standard equidistant

distributed grid is best for low-order methods based on neighboring grid points, such as 'Quad'. The HE study by Brodal et al. (2023) showed that quadratic optimization required larger grids (N), see Fig. 2, and more run time than high-order ΔT_{pinch} methods to obtain similar precision.

Hybrid ΔT_{pinch} Model (Base Case)

Since $\Delta T_{i,\text{ex}}$ are calculated directly from $P_{i,\text{ex}}$, adding more points to the original grid $\Delta T_{i,j}$ can only improve the accuracy of ΔT_{pinch}, i.e., within the limits of the fluid property package. The method named 'Hybrid' combines low and high-order methods using Chebyshev grids; by first calculating $\Delta T_{i,\text{ex,Quad}}$ using 'Quad' and $\Delta T_{i,\text{ex,High}}$ using 'High-order'. These new points ($\Delta T_{i,\text{ex,Quad}}$ and $\Delta T_{i,\text{ex,High}}$) are added to the original Chebyshev grid ΔT_i, before 'Quad' is performed a second time to this new grid, calculating $\Delta T_{i,\text{ex,Quad2}}$. Brodal et al. (2023) recommends the 'Hybrid' ΔT_{pinch} approximation, $\Delta T_{\text{pinch}}=\min(\Delta T_{i,j},\ \Delta T_{i,\text{ex,Quad}},\ \Delta T_{i,\text{ex,High}},\ \Delta T_{i,\text{ex,Quad2}})$, since high-order methods are less accurate if the HE temperature profile is unsmooth, e.g., if fluid property packages introduce a lot of numerical noise or struggle to find dew and bubble points. However, such problems were only observed for HE with refrigerant mixtures, and not pure refrigerants which are studied here. Results from Brodal et al. (2023) are illustrated in Fig. 2, which illustrates the error in different ΔT_{pinch} approximations when modeling a single HE. Fig. 2 shows that 'Hybrid' and 'High-order' methods need significantly less grid points to obtain accurate results compared to the conventional 'Grid only' method. However, performance of these methods has not yet been studied and compared when solving SHI and HEN optimization problems, which is a main goal in this article.

2.2. Optimization and SHI modeling

The optimization schemes used for heat pump A, B and C are explained below.

2.2.1. Optimization algorithms

The goal is to find the best optimization algorithm for solving basic SHIs. However, this work only investigates algorithms available in the MATLAB (2020) Optimization Toolbox, which includes a large variety of popular optimization algorithms. The methods investigated here are Fminsearch, Fmincon, Fminunc, genetic algorithm (GA) and particle swarm (PS). Fminsearch is unconstrained, and uses a deterministic simplex algorithm developed by Nelder and Mead (1965). Fmincon is also deterministic and can solve non-linear constrained problems using a gradient-based approach. The default Fmincon optimization algorithm is the interior-point method (Byrd et al., 1999), but other options such as sequential quadratic programming (sqp) exist (Spellucci 1998; Powell 2006). Fmincon estimates gradients numerically, using either forward or central finite difference schemes. The central scheme requires twice as many function evaluations, but the increased number of calculations can result in better accuracy. Fminunc is an unconstrained deterministic gradient-based method and includes different optimization methods such as the default Quasi-Newton method (Shanno 1970). Both GA and PS are stochastic algorithms. PS is unconstrained, where the swarm members move and velocities are changed in each optimization step, depending on both the best location of close neighbors and the whole swarm, as described by Kennedy and Eberhart (1995). GA is inspired by genetics (Levy 1991), and uses population members to create the next generation, and includes strategies such as survival of the fittest, mutations and mixture of parents genes. Process modeling noise creates problems for the optimization algorithms. For example, optimization searches are likely to find a local minimum where the ΔT_{pinch} approximation error is the greatest, since a large ΔT_{pinch} error increases the objective function (COP). Numerical noise generated by the fluid property packages is typically small, but also effects the objective function (COP). Even a small amount of noise can affect optimization algorithms, e.g., creating an artificial local minimum at a point where the ΔT_{pinch} error is large due to grid point distribution in the numerical work. The region in which an optimization algorithm is affected by such a noise generated local minimum is tiny, and the region is further reduced by improving the accuracy of the ΔT_{pinch} approximation, making it almost random if such numerical noise will affect the optimization at all. Hence, a re-optimization where the first step is not identical to the last step in the previous optimization can be used to find real improvements. Sequential optimization schemes, where an old solution is reoptimized, can therefore be used to find improvements by slightly changing optimization parameters, such as step size, or using totally different optimization algorithms.

Table 1
Optimization schemes compared.

'Fminsearch'	'Fmincon(sqp,forward)'
'Fminunc'	'Fmincon(sqp,central)'
'PS(10)'	'Fmincon(interior,forward)'
'PS(50)'	'Fmincon(interior,central)'
'GA(50)'	'Fmincon(All)'

Table 2
Equipment constraints in the ΔT_{pinch} based optimization.

Heat pump A (Case 1 & 4)	Heat pump B (Case 2 & 5)	Heat pump C (Case 3 & 6)
$\Delta T_{\text{pinch evaporator}} \geq \Delta T_{\min}$	$\Delta T_{\text{pinch evaporator}} \geq \Delta T_{\min}$	$\Delta T_{\text{pinch evaporator}} \geq \Delta T_{\min}$
$\Delta T_{\text{pinch gas cooler}} \geq \Delta T_{\min}$	$\Delta T_{\text{pinch gas cooler}} \geq \Delta T_{\min}$	$\Delta T_{\text{pinch gas cooler}} \geq \Delta T_{\min}$
	$\Delta T_{\text{pinch IHE}} \geq \Delta T_{\min}$	$\eta_{\text{ejector}*} \leq \eta_{\text{ejector}}$

The 10 different optimization schemes, listed in Table 1, are compared in this study. Default MATLAB optimization values are used unless otherwise stated, i.e., function tolerance and constraint violation tolerance are set to 1E-6. GA is modeled with default population size of 50, named 'GA(50)', Particle swarm is computed with 10 swarm members, named 'PS(10)', and with 50 swarm members, named 'PS(50)'; Fmincon is studied with both interior-point and sqp algorithms, while using either forward or central finite differences schemes. These are named 'Fmincon(sqp,forward)', 'Fmincon(sqp,central)', 'Fmincon(interior,forward)' and 'Fmincon(interior,central)'. A sequence based optimization search, named 'Fmincon(All)', is also investigated. This approach first calls 'Fmincon(sqp,forward)' and 'Fmincon(interior,central)'. The best solution is then used as an initial guess in 'Fmincon(interior,forward)', and the solution is used as an initial guess in a final 'Fmincon(sqp,central)' optimization. Different Fmincon(All) sequences were tested. The spq algorithm was usually best, but this method sometimes found incorrect solutions, which were difficult to reoptimize. The success rate improved slightly by also doing an interior search from the original initial point, and then reoptimize the best solution obtained from the two approaches.

2.2.2. Equipment modeling with optimization constraints

Ejector and HE performance are implemented through optimization constraints. This article focuses on the performance of different ΔT_{pinch} approximations, and for simplicity, the same minimum allowed temperature approach ($\Delta T_{\min}$) is used for all the HEs. Ejectors are modeled with a maximum allowed ejector efficiency (η_{ejector}). The different optimization constraints for each heat pump design (A, B and C) are shown in Table 2. Note that instead of using '=' on all the performance constraints, less strict '$\leq$' or '$\geq$' constraints have been used to make it easier for the optimization algorithm to find the path to the optimized result. That is, it is not necessary to enforce that these equipment performance parameters cannot be worse during the optimization, since the final optimized results obtain the best performance values, e.g., $\eta_{\text{ejector}*} \approx \eta_{\text{ejector}}$ and $\Delta T_{\text{pinch IHE}} \approx \Delta T_{\min}$.

Other constraints, not directly related to component performance, are also applied to avoid impossible heat pump processes. That is, a constraint requiring that the pressure cannot increase in the expansion valve ($\Delta p_{\text{valve}} < 0$), and a constraint for systems with an internal suction

Table 3
Process variables being optimized, initial starting point x_0, lower bound **LB** and upper bound **UB**.

Heat pump A	Heat pump B	Heat pump C	Initial optimization guess x_0	LB	UB
T_1	T_1	T_8	$273.15K\cdot(1-0.01\cdot(0.5-rand))$	0 K	500 K
T_3	T_3	T_3	$223.15K\cdot(1-0.01\cdot(0.5-rand))$	0 K	500 K
p_2	p_2	p_2	$95bar\cdot(1-0.01\cdot(0.5-rand))$	1 bar	1000 bar
–	T_{1^*}	–	$15^{\circ}C\cdot(1-0.01\cdot(0.5-rand))$	1E-6 °C	100 °C
–	–	p_r	$4\cdot(1-0.01\cdot(0.5-rand))$	1	45

Table 4
Input parameters for the different heat pump designs. Initial guess values (x_0) are listed in Table 3.

Heat pump	Inputs process scheme	Inputs optimization scheme
A	$P_{\text{gas cooler}}$, R, T_a, T_b, T_c, T_d, T_1, T_3, p_2 and η_{comp}	$\Delta T_{\min}$ and x_0
B	$P_{\text{gas cooler}}$, R, T_a, T_b, T_c, T_d, T_1, T_3, T_{1^*}, p_2 and η_{comp}	$\Delta T_{\min}$ and x_0
C	$P_{\text{gas cooler}}$, R, T_a, T_b, T_c, T_d, T_8, T_3, p_2, p_r and η_{comp}	$\Delta T_{\min}$, η_{ejector} and x_0

Table 5
Output parameters for different heat pump designs. Optimization parameters x are listed in Table 3.

Heat pump	Outputs process scheme	Outputs optimization scheme
A	$\Delta T_{\text{pinch evaporator}}$, $\Delta T_{\text{pinch gas cooler}}$ and COP	$\text{COP}_{\text{optimized}}$ and $x_{\text{optimized}}$
B	$\Delta T_{\text{pinch evaporator}}$, $\Delta T_{\text{pinch gas cooler}}$, $\Delta T_{\text{pinch IHE}}$ and COP	$\text{COP}_{\text{optimized}}$ and $x_{\text{optimized}}$
C	$\Delta T_{\text{pinch evaporator}}$, $\Delta T_{\text{pinch gas cooler}}$, $\eta_{\text{ejector}*}$ and COP	$\text{COP}_{\text{optimized}}$ and $x_{\text{optimized}}$

gas HE to ensure that the inlet compressor temperature is increased ($T_1 - T_{1*} > 0$). Systems with ejectors are modeled with additional constraints requiring the different refrigerant mas flows to be positive.

2.2.3. The optimization scheme

Modeling work consists of two parts: the process model and the optimization search. The process model is used in each optimization step and calculates process values such as the objective function (COP) and the optimization constraints (ejector efficiency and ΔT_{pinch}) from nonlinear algebraic equations. However, calculating the ΔT_{pinch} value is special since it is based on approximations that requires significantly more numerical computations than the other performance parameters. The process model requires inputs related directly to the case study, but also the optimization variables ($\mathbf{x}$). For example, $\boldsymbol{x} = [T_1, T_3, p_2]$ for heat pump A. Table 3 shows the initial points (x_0) used for all cases. A small random noise is also added using the MATLAB function 'rand' to avoid identical optimization results for systems modeled with accurate HE schemes. The lower and upper bounds in Table 3 are only used by GA and PS, and the search region is large to allow efficient modeling of all the relevant systems using optimization sequences. That is, to speed up GA and PS algorithms the MATLAB settings 'InitialPopulationMatrix' and 'InitialSwarmMatrix' are used, respectively, with a shotgun algorithm that distributes the initial points randomly around the initial guess x_0 in the region: $[x_0 \cdot 0.95,\ x_0 \cdot 1.05]$. Due to the large upper and lower boundaries, a solution can be found even if the initial guess x_0 is bad. After relatively few optimization steps, all GA and PS points typically end up in a small region around the best point. Hence, it is often better to distribute the points around a smaller region that is assumed to be a good initial guess, than in a large region where most of the points will have unphysical values because they violate optimization constraints and therefore have large penalties. Compressor efficiency (η_{comp}) is implemented directly in the process model, while the HEs and ejector equipment performance parameters ($\Delta T_{\min}$, and η_{ejector}) are implemented through optimization constraints (see Table 2). Inputs and outputs of process and optimization schemes are summarized in Tables 4 and 5. More details are also given in the article (Brodal and Eiksund 2020).

The most energy efficient heat pump has the greatest COP. Since optimization algorithms find a minimum, the objective function to be minimized is therefore:

$$\text{COP}_{\text{optimized}} = \min(-\text{COP}), \tag{10}$$

for optimization algorithms that can implement the non-linear constraints in Table 2 directly. Of the optimization methods investigated here, only the Fmincon function has a direct (non-penalty based) implementation of the non-linear constraints. To solve unconstrained optimization algorithms, the constraints listed in Table 2 are added to Eq. (10) through penalty terms. The GA function in MATLAB already includes a penalty-based method, however, many prefer to make their own (Ding et al., 2017). For consistency, GA is here solved with the same penalty terms as the other unconstrained methods (Fminsearch, Fminunc, and PS). That is, for heat pump A with two HEs, the optimization problem has additional penalty terms for each HE:

$$\begin{aligned}\text{COP}_{\text{optimized}} = \min\Big(&-\text{COP} + k_2 \cdot \left[\max\left(0, \Delta T_{\min} - \Delta T_{\text{pinch evaporator}}\right)\right]^2 \\ &+ k_3 \cdot \left[\max\left(0, \Delta T_{\min} - \Delta T_{\text{pinch gas cooler}}\right)\right]^2\Big),\end{aligned} \tag{11}$$

where k_i are penalty factors, and $\Delta T_{\min}$ is the minimum allowed temperature approach in HEs. A squared penalty term is used. This approach generates a penalty term with a continuous derivative, however, other terms can be applied such as the penalty formula used by Ding et al. (2017). $\Delta T_{\text{pinch evaporator}}$ and $\Delta T_{\text{pinch gas cooler}}$ are approximations of the pinch point temperature difference in the evaporator and gas cooler, respectively. Heat pump B also includes an internal HE, which is modeled by adding $k_4 \cdot [\max(0, \Delta T_{\min} - \Delta T_{\text{pinch IHE}})]^2$ as a new penalty term to Eq. (11). Heat pump C is forced to operate with a given ejector efficiency (η_{ejector}) by calculating a process ejector efficiency ($\eta_{\text{ejector}*}$), and adding $k_5 \cdot [\max(0, \eta_{\text{ejector}*} - \eta_{\text{ejector}})]^2$ to the optimization problem described in Eq. (11). The best optimization parameters ($x_{\text{optimized}}$) are found using Eq. (11), and finally $\text{COP}_{\text{optimized}}$ is computed from the $x_{\text{optimized}}$ using Eq. (1), i.e., the final results are computed without penalty terms. All penalty factors are set to $k_i = 1000$. Large k_i factors are used to obtain optimization results with small constraint violations. Reducing the penalty factors will make it easier to solve Eq. (11), but the results are more likely to violate the constraints.

Constraint violations can indicate that an optimization was unsuccessful. To remove potentially bad solutions, only optimization results with constraint violations less than 0.001 for ejector efficiency, 0.001 K for temperature pinches are considered as successful optimization. That is, assuming that results with larger constraint violations should be reoptimized using a different initial guess.

Table 6
Definition of cases 1 – 6 with refrigerant (R).

	Case 1 (R)	Case 2 (R)	Case 3 (R)	Case 4 (R)	Case 5 (R)	Case 6 (R)
Heat pump	A	B	C	A	B	C
T_b	70 °C	70 °C	70 °C	35 °C	35 °C	35 °C

2.3. Error estimates and run time

Analytical solutions are not available, hence the optimal value (COP_{best}) must be estimated from the optimized results obtained with accurate HE schemes with large N. In this work, COP_{best} is found from 10 different optimization approaches using 'Hybrid' interpolation schemes with grid points $N = 46$ and 47, i.e.:

$$COP_{best} = \max\left(COP^{Fminsearch}_{optimized,N=46},\ \ldots,\ COP^{PS(10)}_{optimized,\ N=47}, COP^{Fmincon(All)}_{optimized,N=47}\right), \tag{12}$$

The error of a Fminsearch optimization with grid size N is defined as:

$$COP\ error = abs\left(\frac{COP_{best} - COP^{Fminsearch}_{optimized,N}}{COP_{best}}\right)\cdot 100\%. \tag{13}$$

The computer run time depends on the number of calculations, but also the computer hardware. However, run time is an inaccurate measurement of optimization workload since it also depends on programs running in the background. The results in this study are generated using a machine with Intel(R) Xeon(R) W-2123 CPU @ 3.60GHz processor and 32 GB RAM.

2.4. Success rates and mean run times

Success rates (s) and mean run times are used to describe optimization scheme performance. In this study, s is defined as the percentage of successful optimizations with respect to the total number of optimizations and the mean run time is calculated the average run time. Success rates (s) and mean run times are obtained for different HE grid intervals. Two different criteria for optimization success are studied, i.e., COP error $< 0.01\%$, and COP error $< 1\%$. The random starting point, defined in Table 3, is also an important factor implemented in s. The success rate of n different optimizations with a success rate s, is modeled statistically as the success rate where at least one of the n optimizations is successful:

$$s_{tot}(n) = (1 - (1 - s)^n), \tag{14}$$

where $(1 - s)^n$ is probability that all n optimizations failed.

2.5. Definition of case 1–6

The three heat pump designs A, B and C, as illustrated in Fig. 1, are modeled with different heating demand and refrigerants (R). The feed water is warmed from $T_a = 15$ °C to T_b, which is either 35 or 70 °C depending on the case as explained in Table 6. The heating duty is always assumed to be $P_{heat} = 10$ kW. In heat pump C, which has an ejector, refrigerant exits the evaporator at its dewpoint. Four refrigerants (carbon dioxide, propane, ammonia and R134a) are modeled. The water used as a heat source is cooled in the evaporator from $T_c = 10$ °C to $T_d = 5$ °C. Different equipment performances are modeled individually, i.e., with compressor efficiency $\eta_{comp} = 0.75$, ejector efficiency $\eta_{ejector} = 0.17$, and minimum allowed temperature approach $\Delta T_{min} = 2.0$ K for all HEs.

2.6. Model validation

Since an analytical solution does not exist for optimized SHI problems, optimization is validated by calculating the same problem multiple times using different optimization algorithms, starting values (x_0), HE schemes and number of grid points N. Convergence between the results obtained with different schemes, as N increases, is illustrated in Figs. 6 and A1. Grid sizes with $N \geq 20$ typically generate a large number of results that deviate less than 0.001% from the COP_{best} for all the systems modeled, indicating that an accurate optimization has been found in all cases. The novel 'Hybrid' ΔT_{pinch} approximation was also validated against the conventional 'Grid only' method. For example, Figs. 4 and 5 show that optimizations with 'Grid only' converge to the same COP value as the base case ('Hybrid') for large N, and that the difference in COP is less than 0.01% for 'Grid only' approximations with $N \geq 100$. Novel 'Hybrid' approximations have also been validated against 'Grid only' in a larger convergence study of ΔT_{pinch} for single HEs in schemes, where the difference in ΔT_{pinch} was found to be less than 0.0001% for schemes with $N \geq 1000$ (Brodal et al., 2023).

3. Results

Optimized heat pump processes for CO_2 are illustrated in Fig. 3 for the different cases defined in Table 6. With CO_2 as the working fluid, Case 1–3 are transcritical processes with gas cooler pressure around 100 bar, while Case 4–6 are subcritical processes with condenser pressures around 70 bar, just below critical pressure of CO_2. Optimized processes with propane, ammonia and R134a are subcritical processes with condenser pressures much less than critical pressure. Although this work focuses only on optimization, a detailed sensitivity study of heat pump A, B and C is presented by Brodal et al. (2019).

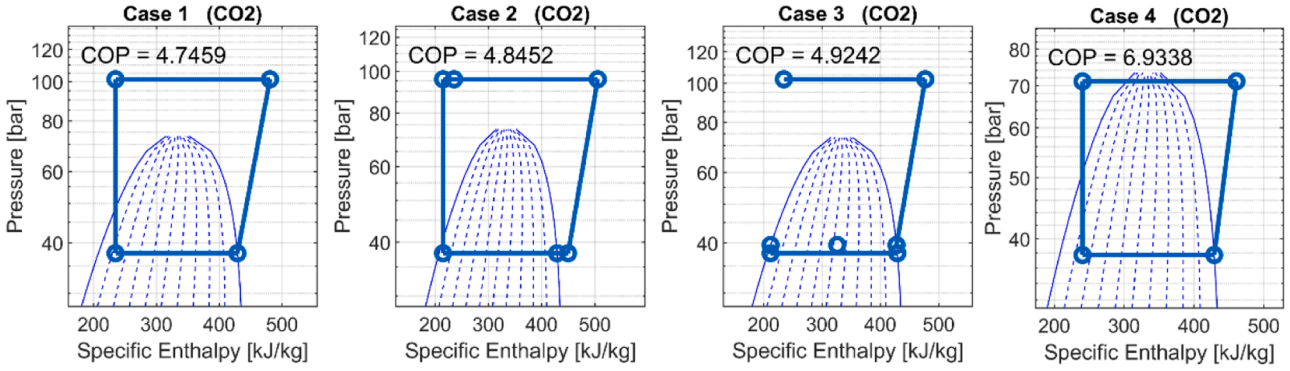

Fig. 3. Pressure-enthalpy (p-h) diagrams of optimized CO_2 processes.

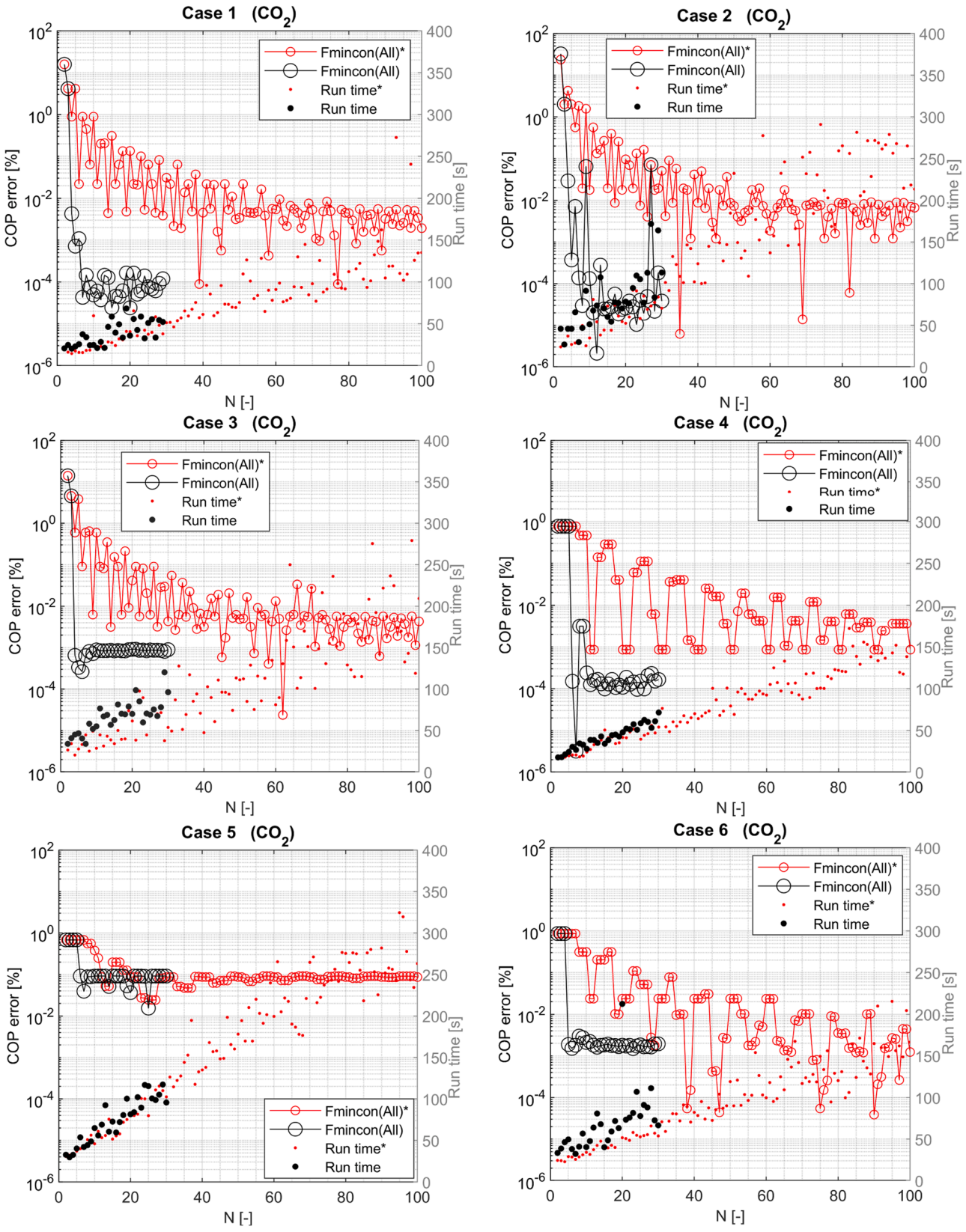

Fig. 4. 'Fmincon(All)' optimized COP errors and computer run times based on ΔT_{pinch} HE modeling. The '*' symbol indicates that the 'Grid only' method is applied instead of the 'Hybrid' (base case) method.

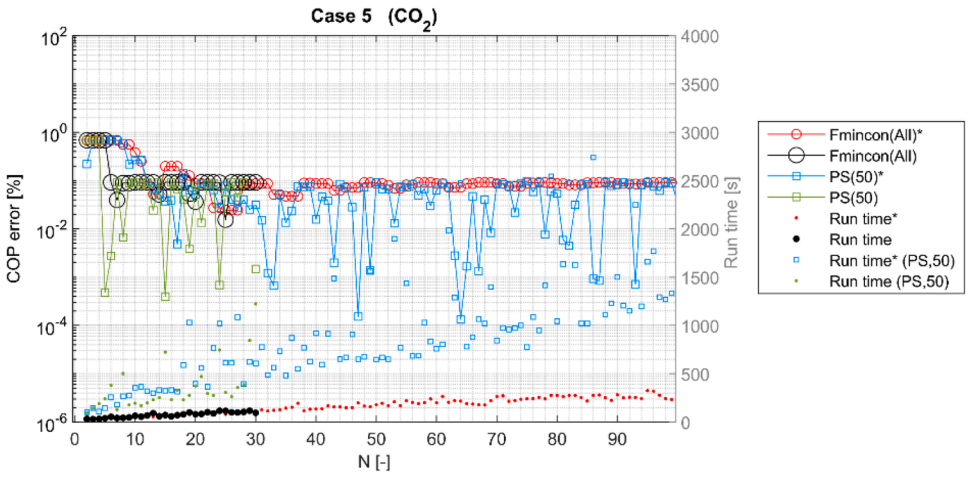

Fig. 5. COP errors in optimizations, where the '*' symbol is used for schemes where the 'Grid only' method is applied instead of the 'Hybrid' (base case) method.

Table 7
Grid points N needed by 'Grid only' to obtain the 'Hybrid' scheme accuracy with $N = 5$, 10 and 15.

ΔT_{pinch} error estimates in single HEs (See Fig. 2)		COP error estimates using Fmincon(All) (See Fig. 4)	
'Hybrid'	'Grid only'	'Hybrid'	'Grid only' (Increased run time)
$N \approx 5$	$N \geq 20$	$N \approx 5$	$N \approx 100$* (500 – 1000%)
$N \approx 10$	$N \geq 150$	$N \approx 10$	$N \gg 100$* (More than 500%)
$N \approx 15$	$N > 300$**	$N \approx 15$	$N \gg 100$* (More than 500%)

* $N \approx 30$ if Particle swarm is used (see Fig. 5).
** Estimate based on figures in Brodal et al. (2023).

3.1. Choice of ΔT_{pinch} scheme

Optimization errors solving the three transcritical and the three subcritical CO_2 cases are illustrated in Fig. 4, where ΔT_{pinch} is modeled with either the novel 'Hybrid' (base case) method using Chebyshev grids, or the conventional 'Grid only' method using equidistant grid distribution. Fig. 4 shows that the 'Hybrid' method typically needs far fewer HE grid point evaluations and less optimization run time to obtain similar results. For example, the 'Hybrid' method only requires $N \geq 4$ grid points (run time $\approx$ 20 s) to obtain results with COP error $< 0.01\%$, while 'Grid only' needs about $N \approx 60$ (run time $\approx$ 100 s) to obtain similar accuracy. For Case 2 with CO_2 the 'Grid only' method requires $N \approx 100$ (run time $\approx$ 250 s). Note that, there are no 'Fmincon(All)' results for Case 5 with COP error $< 0.01\%$. Since the optimization accuracy is not improved using more accurate ΔT_{pinch} approximation with large N, the Case 5 optimizations must be limited by other factors than the accuracy of the HE approximation. Fig. 5 illustrates that PS(50) is successful a few times for Case 5 for both 'Grid only' and 'Hybrid' HE approximations, but COP error $< 0.01\%$ only occur in less than 20% of the optimizations. The 'Grid only' approximations only require $N \approx 30$ to obtain similar results as the 'Hybrid' PS optimizations, however, such optimizations are slow (around 1000s). Performance of 'Grid only' and 'Hybrid' HE approximations in optimization problems are summarized in Table 7. Since the 'Hybrid' outperformed the conventional 'Grid only' approach, 'Hybrid' approximations are used when comparing different optimization algorithms in Section 3.2. Note that the run time for most optimization methods increases with N, but not all. This clearly illustrates that optimizations based on inaccurate and fast ΔT_{pinch} schemes with few grid points (N) sometimes will require more run time, which probably indicates that more optimization steps are required to find the solution due to the numerical noise.

3.2. Choice of optimization algorithm

The HE approximations, the numerical noise in fluid property packages and the optimization algorithm are all important for the overall success of optimization scheme. This section investigates how the grid size used in the ΔT_{pinch} approximation affects different optimization algorithms, in order to identify the best optimization scheme. Optimization schemes are modeled using the 'Hybrid' ΔT_{pinch} approximation, because Section 3.1 showed that this approach was better than

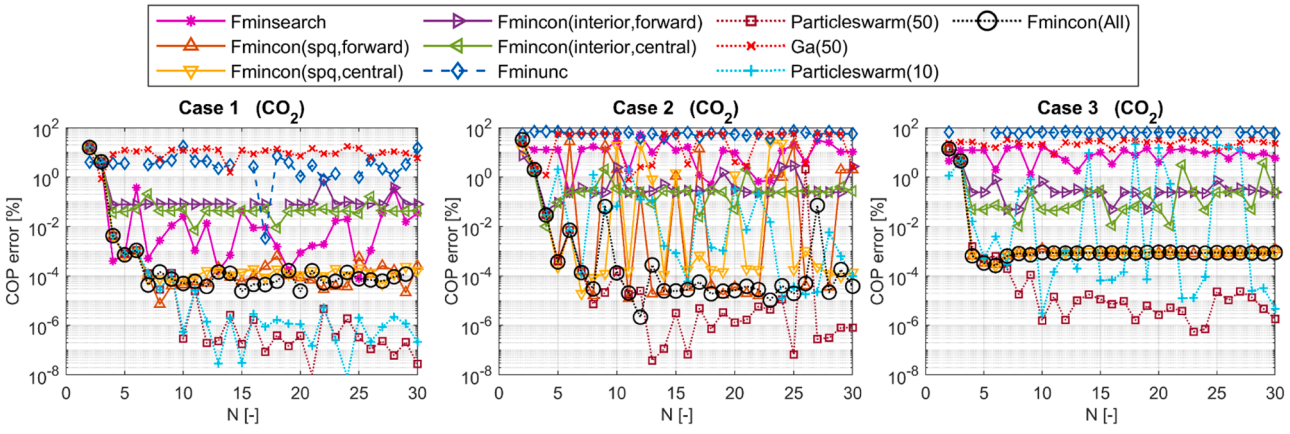

Fig. 6. Errors in optimized COP for Case 1 – 3 (CO_2) with respect to the number of HE grid points N.

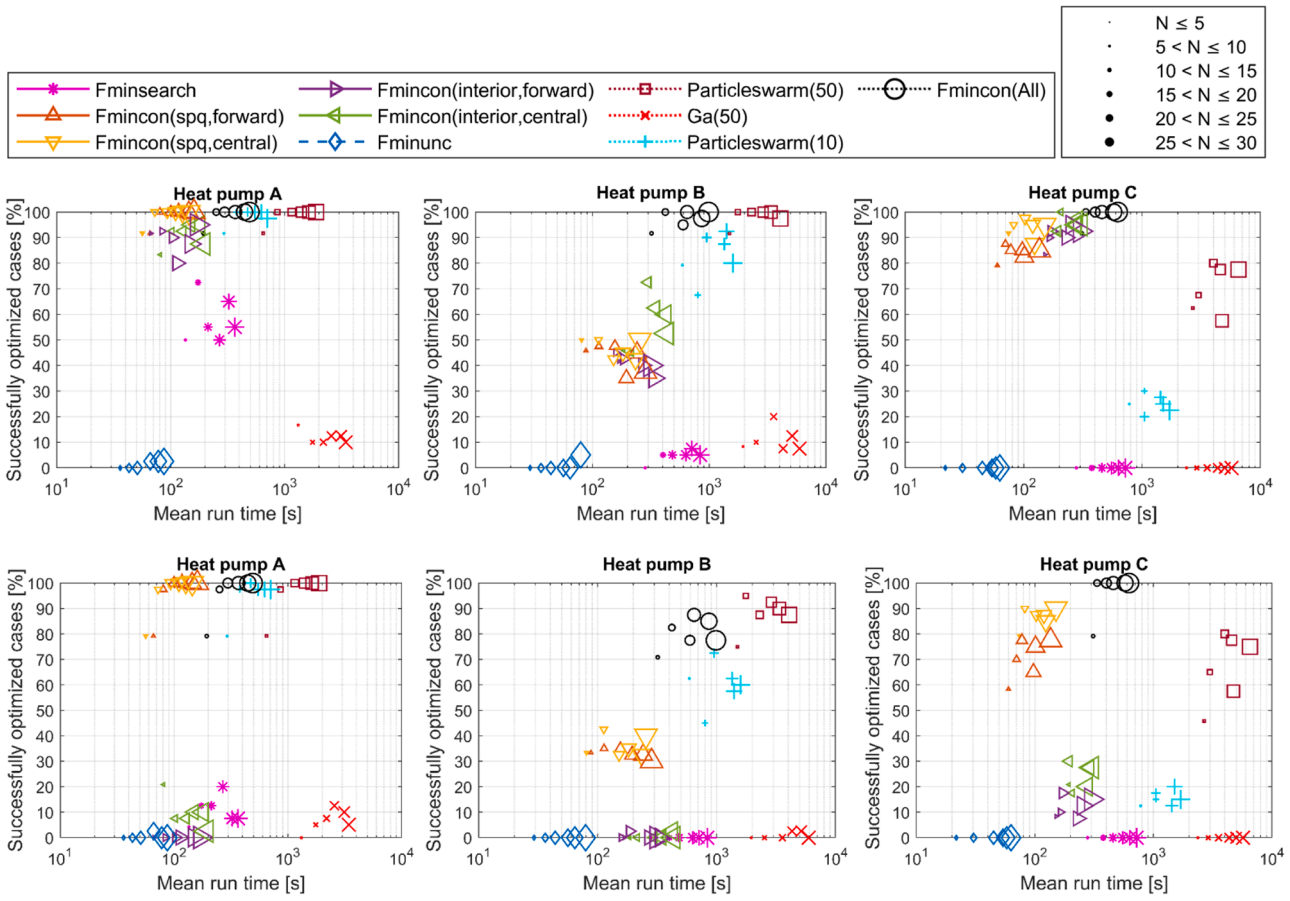

Fig. 7. Success rate (s) of optimized problems with respect to mean run time, using different optimization methods and HE grid sizes. Top: error COP $< 1\%$. Bottom: error COP $< 0.01\%$.

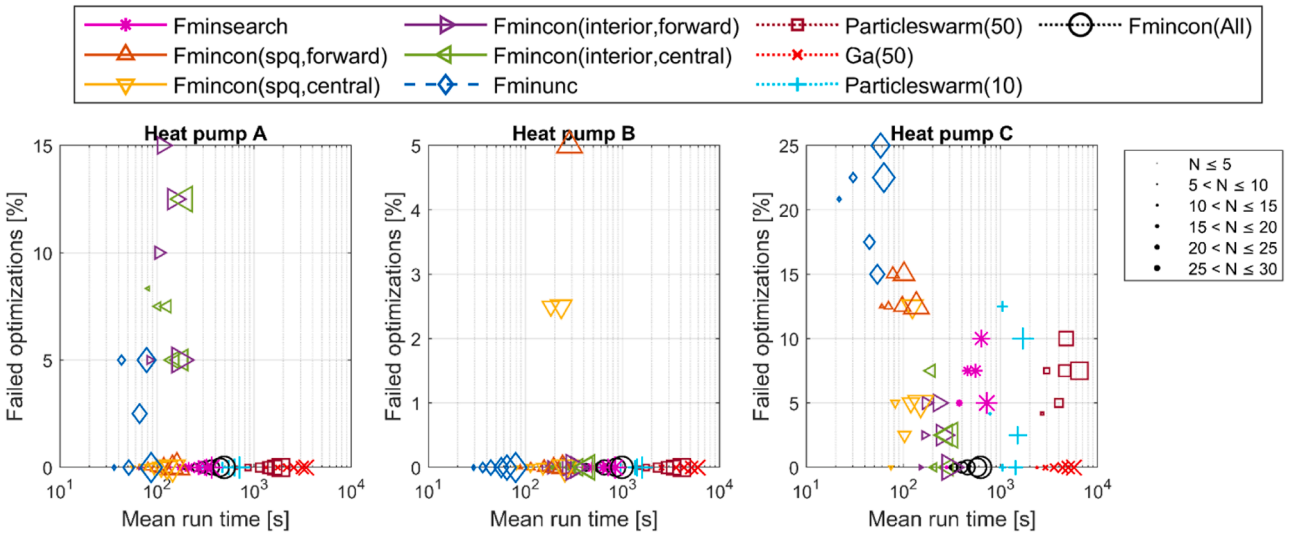

Fig. 8. Percentage of optimized problems that failed by violating default MATLAB optimization tolerances or the user defined constraint tolerances.

'Grid only'. COP optimization errors obtained in Cases 1 – 6 were solved using Fmincon(All) and nine other optimization searches for systems with CO_2, propane, ammonia and R134a. How the accuracy of the different optimization methods depends on grid size N, is illustrated for transcritical CO_2 processes in Fig. 6. Fig. A1, in the appendix, shows the optimalization run time for each grid size. Cases modeled with propane, ammonia and R134a are subcritical processes, where ΔT_{pinch} is located at bubble and dew points. Since these points always are calculated prior to the interpolation, there is no gain in accuracy by increasing the grid size or use interpolation, as shown in Fig. A1. The variation of these results is therefore directly related to the random initial guess (see Table 3). However, not all subcritical processes have a pinch point located at the bubble and dew points. For example, Figs. 4 and A1 show that the grid size and the interpolation method are vital when

Table 8
Recommended optimization schemes for heat pumps.

Heat pump	COP error less than	Best optimization scheme	
		Algorithm	HE grid size
A	1.00%	Fmincon(sqp)	$5 < N \leq 10$
B	1.00%	Fmincon(All)	$5 < N \leq 10$
C	1.00%	Fmincon(All)	$5 < N \leq 10$
A	0.01%	Fmincon(sqp)	$10 < N \leq 15$
B	0.01%	Fmincon(All)	$5 < N \leq 20$
C	0.01%	Fmincon(All)	$5 < N \leq 10$

optimizing the subcritical CO_2 processes (Cases 4–6).

Fig. 7 shows the optimization success rate for different HE grids and is created from the data presented in Figs. 6 and A1, as explained in the method Section 2.4. Since two cases are modeled for each heat pump (A, B and C) using four different refrigerants (CO_2, propane, R134a and NH_3), and each grid interval presented consists of five simulations, the percentages presented in Fig. 7 are based on $2 \bullet 4 \bullet 5 = 40$ different optimizations for each heat pump design. Fig. 7 shows that PS(50) in average requires about 10–40 times longer run time than Fmincon(sqp), and 5–10 times longer than Fmincon(All). However, Fmincon(sqp) is less successful in its optimization of the more complex heat pump processes (B and C). Figs. 4–6 show that the accuracy of the ΔT_{pinch} approximation is important for the optimization, until a certain precision is achieved, where other factors are limiting the optimization success. In general, Fig. 7 shows that improving the HE model resolution from $20 < N \leq 25$ to $25 < N \leq 30$ does not improve the optimization results, except for the unconstrained gradient method named Fminunc. However, for some of the optimization algorithms there is not a clear trend between N and the average success rate, which indicates that the ΔT_{pinch} approximation is not always the limiting factor. Numerical noise generated in fluid property packages and the methods used by the optimization algorithm itself are also important factors. Fig. 8 shows that the success rates in Fig. 7 are also influenced by cases where the optimization failed to find an acceptable solution, i.e., solutions where the solver itself identified problems, or solutions having ΔT_{pinch} estimate errors larger than 0.001 K. Such failed cases are detectable and can therefore be reoptimized. All the other failed cases in Fig. 7 were due to optimization inaccuracies, i.e., obtaining solutions where the global optimum was not found even though the MATLAB solvers reported 'minimization succeeded'. Note that such inaccurate solutions can only be detected by comparing results from multiple optimizations.

Table 8 lists the best optimization schemes for the different heat pumps, based on the results in Fig. 7. Table 8 shows that Fmincon is the best optimization algorithm for all the systems. Multiple optimizations based on different random initial guesses can be used if the success rate is insufficient (see Eq. (14)). Note that, even though PS(50) has a slightly better success rate than Fmincon(All) for heat pump B, it is not statistically better with respect to run time since it requires more computations. About 95% of the PS(50) calls and 82% of the Fmincon(All) calls with $5 < N \leq 10$ are successful (see Fig. 7). Statistically, see Eq. (14), only two Fmincon(All) optimizations with 82% success rate are required to obtain a $(1-(1-0.82)^2)\cdot 100\ \% \approx 97\ \%$ success rate, which is better than a single PS(50).

4. Discussion

The central problem that is addressed in this work is that even small inaccuracies in the model predictions for HEs will introduce numerical noise that can cause optimization algorithms to fail. The results presented above for optimization success rates with respect to HE grid size and computer run time are discussed below for different HE schemes and optimization algorithms.

4.1. Required accuracy

A 1% COP modeling accuracy is often sufficient for existing SHIs since equipment such as flow and temperature gage often have less precision. In optimization studies, it can even be interesting to obtain a higher numerical accuracy than the physical precision of the simplified equipment models, since they aim to set the optimum starting point for detailed (equipment level) design work. That is, finding an improvement of 1% by adjusting the optimization parameters can be useful, even if COP is calculated with an accuracy beyond the physical modeling precision. That is, one should not terminate the optimization when a 1% accuracy is reached or introduce unnecessary numerical approximations in the process model. The importance of accurate optimization is discussed in the LNG sensitivity study by Brodal et al. (2019), where several mixed fluid cascade (MFC) processes were optimized. In this study it was difficult to identify optimal refrigerant mixtures even though the optimization errors were relatively small. That is, for COP errors less than 0.3%, optimized mixtures had component fractions that variated randomly between 13 and 30%. A high optimization accuracy is also vital for creating smooth sensitivity plots, which is required to understand trends and design benefits. Based on experience from the study of similar heat pump processes (Brodal and Eiksund 2020), it is believed that a COP optimization error of less than 0.01% is sufficient in sensitivity studies. An accuracy of 0.01% can also give performance advantages during the design work, since it is reasonable to assume that processes can be slightly improved (on average) if they are based on better parameters. Despite setting the upper allowed constraint violation limit in the optimization to 0.001 K, the actual violation can be much higher if the ΔT_{pinch} estimation is inaccurate, e.g., as much as 10 K for a low-order $N = 2$ scheme (Brodal et al., 2023). That is, solutions based on low-order approximations can be unphysical with an actual ΔT_{pinch} value less than 0 K. Low-order schemes can often be solved quickly due to less computations, but it is difficult to use such solutions as initial guesses in more accurate optimization schemes because they may violate constraints if the ΔT_{pinch} approximations are improved. In our experience, it is typically better to use the initial points defined in Table 3 directly than to reoptimize a solution found with a less accurate optimization scheme.

4.1.1. Optimization tolerance settings

All optimizations use MATLAB's default termination tolerance, which is 1e-6. Such a high modeling precision of the objective function (COP) is not relevant for a real system, and optimizations are often terminated due to other criteria. For example, the results presented in Figs. 6 and A1, show that COP errors are often much larger than 1e-6. MATLAB's default finite difference step size setting is used in both Fminunc and Fmincon. There are different sources of errors in optimized schemes. The results presented here have neglectable equipment performance errors, since results with equipment constraint violations above certain thresholds are removed. That is, results with constraint violations larger than 0.001 for ejector efficiencies and 0.001 K for pinch points are removed, since optimizations terminated with even small constraint violations can indicate that the optimization has failed. However, the actual pinch point violation in solutions of inaccurate schemes can be much bigger than 0.001 K (Brodal et al., 2023). Fmincon is the only method in this study not using penalty terms to implement constraints. Relatively large penalty factors (k_i=1000) has been used to obtain accurate results with only small equipment constraints violations. That is, to avoid generating unphysical results significantly better that the optimal process. In this study, COP errors due to constraint violations are typically much less than 0.01 %. Note that, COP errors due to penalty factors are typically much greater than the 1e-6 default constraint violation tolerance used by Fmincon, which the unconstrained methods are being compared with. The optimization settings and penalty factors used in this study are based on experience from a

previous study (Brodal and Eiksund 2020), where penalty factors to some extend were adjusted to obtain accurate results generating smooth sensitivity plots. If the requirement is only a COP error less than 1%, the penalty factors can be reduced to improve computational speed. However, to compare unconstrained and constrained optimization algorithms fairly, the Fmincon constraint tolerance should also be modified. Finding solutions with such small constraint violations is relatively easy, as illustrated by continuous lines in Fig. 6, even in inaccurate optimizations where COP errors are larger than 10%. The default Fmincon constraint optimization tolerances in MATLAB are much greater than the user defined constraint tolerances, and Fig. 8 shows that only the Fmincon search failed to find heat pump A and B solutions within the optimization tolerance. However, Fig. 8 also shows that optimization tolerances problems were avoided when using the optimization sequence Fmincon(All), and that all optimization algorithms except GA and Fmincon(All) sometimes failed to solve heat pump C.

4.2. Choice of ΔT_{pinch} approximation and number of nodes

The performance of optimization schemes depends strongly on the HE approximations applied. The conventional temperature pinch method, here named 'Grid only', is simply finding the minimum grid value $\Delta T_i(j)$ using an equidistant distributed grid and Eq.(7). High-order interpolation methods have previously only been applied to single HEs (Brodal et al., 2023), which recommended hybrid high and low-order schemes since they were less sensitive to noise created by fluid property packages.

4.2.1. Conventional versus novel ΔT_{pinch} schemes

The novel optimization results of HE networks are discussed below, and compared with the single HE modeling results presented by Brodal et al. (2023).

Modeling of Single HEs

Fig. 2, which is created from data from a previous study of single HEs (Brodal et al., 2023), illustrates the accuracy of different ΔT_{pinch} approximation methods and that it is difficult to obtain accurate results with the 'Grid only' method. The 1st-order pinch point interpolation method ('Quad') is less accurate than the novel 'High-order' method for a given grid size N, but is much more accurate than 'Grid only'. Low-order approximations are less affected by numerical noise, and Brodal et al. (2023) identified the 'Hybrid' method, which combines 'Quad' and 'High-order', to be the most accurate method for a given grid size N. However, the 'Hybrid' method is a little more time consuming since it requires three additional grid point evaluations. Fig. 2 shows that the conventional 'Grid only' approximation need more than $N >>$ 140 grid points to obtain similar ΔT_{pinch} accuracy as the 'Hybrid' method with $N \approx 10$ (see Section 4.2.2). Fig. 2 shows that ΔT_{pinch} errors are less than 0.002% in such schemes. When modeling a single HE, the run time is almost linear with the number grid points (Brodal et al., 2023). Hence, the 'Hybrid' method is approximately 15 times faster than the 'Grid only' method to find ΔT_{pinch} with errors less than 0.06% (see Fig. 2).

Optimization Modeling SHIs

When solving SHI optimization problems, which is the novel aspect of this work, the run time gain realized by replacing 'Grid only' with 'Hybrid' is found to be slightly less than for modeling single HEs, as illustrated in Figs. 4 and 5. The results summarized in Table 7 shows that the 'Hybrid' method requires significantly less grid point evaluations and computer run time. To obtain Fmincon(All) optimized results with an COP error $< 0.01\%$, the 'Hybrid' method requires a grid size $N \approx 5$, while 'Grid only' requires $N \approx 100$. Fig. 4 shows that 'Hybrid' is 5 – 10 times faster than the conventional 'Grid only' method when Fmincon (All) is used to obtain accurate results with COP error $< 0.01\%$, which is a practical accuracy (see Section 4.1). If less accuracy is needed, the difference is less. Fig. 4 also shows that 'Hybrid' is only slightly better to produce optimizations with COP errors $\approx$ 1%, which requires $N \approx 5$ instead of $N \approx 10$. Note that the optimization method is also important when comparing different ΔT_{pinch} approximations. For example, Fig. 5 shows that 'Grid only' schemes only require $N \approx 30$ to generate similar results as 'Hybrid' schemes with $N = 5$ when using PS(50). However, the 'Grid only' PS(50) run time is still about 400% larger than the 'Hybrid' PS(50) run time.

4.2.2. Optimal grid size for hybrid ΔT_{pinch} schemes

The best ΔT_{pinch} approximation, i.e., the 'Hybrid' method (see Section 4.2.1), is used when comparing the different optimalization algorithms. The importance of the ΔT_{pinch} approximations accuracy is illustrated in Fig. 7. About 8% of the Fmincon(all) optimizations have COP errors larger than 1% for $N \leq 5$, and some of these solutions are far from optimal, e.g., having 10–50% COP errors (see Figs. 4 and 6). For larger grids $5 < N \leq 10$, where ΔT_{pinch} is calculated more accurately, Fmincon(all) produce only results with COP errors less than 1% (see Fig. 7).

The HE models applied in this study always calculate bubble or dew points, and then divide HEs into sections with smooth ΔT functions (see Section 2.1.3). That is, for subcritical processes where the pinch point is located at the bubble or dew point, there is no accuracy gained by introducing additional grid points. As illustrated in Fig. A1, such additional grid point evaluations only increase run time. For transcritical processes (see Fig. 6), and subcritical processes where the pinch point is not located at bubble or dew points (see Fig. 4), the grid size is very important. Fig. 2 shows that $N = 5$ and $N = 15$ correspond to ΔT_{pinch} errors less than 1% and 1e-3%, respectively. Fig. 7 shows that optimization accuracy improves if the HE grid size (N) increases, but that there is typically not much gained by increasing the grid size above 10. Fig. 7 shows that $N = 10$ is a relatively optimal 'Hybrid' grid size for all SHIs. The required grid size depends slightly on the design, and Table 8 shows that $5 < N \leq 20$ grid points can be recommended when optimizing SHIs with two HEs, three HEs and an ejector. That is, larger grids $10 < N \leq 20$ are typical required to obtain accurate results of complex SHIs with three HEs, while $5 < N \leq 15$ is optimal for SHIs with only two HEs. For SHIs with an ejector, a smaller grid size is sufficient ($5 < N \leq 10$) for Fmincon(all).

4.3. Choice of optimization algorithm

The purpose of this optimization study is to determine the best optimization scheme for simple SHIs (heat pump A, B and C) using different refrigerants and heating scenarios (Case 1–6). The optimized SHI results show that the success rate depends on the optimization algorithm and the heat exchanger scheme. However, the results also depend on other factors such as the process complexity, the size of the penalty factors k_i and the initial guess x_0. The results presented in Figs. 6 and A1 show that, for all cases, there is always more than one optimization algorithm that is able to find COP within a 0.01% error. PS(50) obtained such accurate results for all the cases modeled, while Fmincon (All) only failed obtaining such accuracy for Case 5 with CO_2. However, Fig. 7 illustrate that Fmincon(All) has a much better success rate than PS (50) for SHIs with an ejector. Fig. 7 also shows that Fminsearch, Fminunc and GA often fail. Using an unconstrained algorithm for a constrained problem is typically not ideal. Note that, other methods could be tested. For example, 'feasible path' optimization is widely used in large-sale nonlinear process modeling. However, a feasible path optimization algorithms can also fail as discussed by Ma et al. (2020), who suggested new methods to improve such algorithms.

Since the systems being optimized are very different, the initial guess is often far from the optimized results. Even optimization methods that fail, can often be able to find a better solution than the initial guess. Optimization algorithms that find slightly better solutions, can therefore be successful if used multiple times in a sequence continuously improving the result. That is, the best solution from the previous opti-

mizations is used as a basis for the initial guess in next optimization, either directly using it is an initial optimization guess, or indirectly by searching in the neighborhood. However, such schemes can be extremely time-consuming if they are based on optimization algorithms which are not well suited for a given problem. It is possible to include different optimization algorithms in the search, as done by the Fmincon (All) sequential optimization scheme. Fig. 7 shows that heat pump C success rates are 5–90% for single Fmincon optimizations, and 100% for Fmincon(All). The authors have previously used sequential based optimization schemes to find accurate solutions (Brodal and Jackson 2019; Brodal et al., 2019; Brodal and Eiksund 2020). Using the previous non-optimal solution as initial guess for the next optimization is the equivalent of increasing the number of iterations of the first optimization process, i.e., if the same optimization algorithm and optimization parameters are applied. However, random noise generated by fluid property packages and ΔT_{pinch} approximations can have a significant effect in optimizations, since this noise is sensitive to even microscopic changes irrelevant for the real physical process. This is illustrated by the Fmincon(All) results, which show that the solution can be improved in optimization sequences using different methods, i.e., spq/interior and forward and central finite difference methods.

If constrained gradient-based algorithms fail to optimize a certain SHI, the results show that PS is likely to be the best approach. Stochastic methods like PS and GA are often claimed to be better at finding the global minimum (Rao et al., 2020). GA is perhaps the most used method for SHI optimization, e.g., Austbø et al. (2014) identified 29 GA articles and only 2 PS articles. GA is often used to study complex process, such as industrial heat pump processes (Allen et al., 2009), but it is also used to model less complex systems, e.g. water-to-water heat pump systems (Murr et al., 2011) and CO_2-based transcritical cycles (Ren 2020). Li et al. (2022) states that GA has been widely used and has achieved good results. However, for the systems optimized here, PS is found to be the best stochastic approach. PS is overall the most accurate optimization method for heat pump A and B, but it is not particularly effective for heat pump C with an ejector. However, PS is not recommended since it is about 500–1000% more time consuming than Fmincon schemes (see Fig. 7). The conventional GA approach, on the other hand, is both slow and highly inaccurate. GA has a 0–20% success rate, which is low compared to PS(50) and Fmincon(All) success rates between 70 and 100%. GA should therefore probably never be used to optimize SHIs, even though long GA sequences can find an accurate result. For example, long sequences with more than 10 individual GA calls have earlier been used to optimize heat pump with five HEs (Brodal and Jackson 2019). However, Fig. 7 shows that the run time of 10 different GA(50) optimizations is more than 25,000% larger than the run time of a single Fmincon(spq), which has a 100% success rate for heat pump A. The authors have also experienced that it is difficult to optimize more complex HE systems, e.g., in the optimization study of a modified MFC process Brodal et al. (2019), the refrigerant process had 16 equipment constraints and 16 optimization parameters. However, more research is needed to investigate how GA and other optimization algorithms handle such complex systems.

Stochastic solvers struggle because the penalty term will 'blow up' for a point with even a small constraint violation. Additionally, the probability of picking a random point that has a better objective function (COP) is often significantly greater than the probability for this point not violating any of the ΔT_{pinch} constraints, especially in systems with many HEs. Therefore, PS and GA are probably not operating as intended, and all the points quickly end up in a tiny region around the best point, which is a point not violating the constraints. These methods are then typically only able to finds small improvements, since there is a higher probability of finding a point not violating the constraints close to another point not violating the constraints. Note that all the points recognized as good in GA and PS are not violating the constraints, while the optimal solution to some extent is hidden behind the boundary created by large artificial penalty terms. The GA method is inefficient since better genes/points are hidden by large penalty terms, even if a new improved point is found. PS on the other hand is more successful since finding a new improved point favors the other swarm members/points to move in this direction, and beyond. PS is therefore likely to explore a region further towards the optimal point during the next optimization steps. The size of the penalty constraint factors is important for unconstrained schemes such as PS and GA. Such schemes typically find the optimal solution much faster if penalty factors are reduced. However, if the penalty constraint factors are reduced, the optimal solutions will have greater constraint violations. These solutions are therefore likely to be identified as bad points if the penalty constraint factors are increased, making it very difficult to use such points to find a more optimal solution in a sequential search. Even though large penalty factors are used in unconstrained optimizations, constraint violations in Fmincon is often less, due to the 1e-6 constraint violation tolerance used by Fmincon. The best swarm size for PS was investigated by studying performance of swarms with 10 and 50 members, named PS(10) and PS(50), respectively. The results in Fig. 7 shows that PS(10) is typically 3 times faster than PS(50), which is the default swarm size in MATLAB. However, PS(10) is often less successful than PS(50). The success rate and run time of both these schemes could probably be improved by modifying the default MATLAB settings, such as the maximum allowed iterations and maximum allowed iterations without finding a better point, which are 200 and 20 respectively. More work is needed to find optimal swarm size and stopping criteria for SHIs.

Numerical noise affects fminunc and fmincon algorithms differently. However, for all noise levels, the results show the penalty approach implemented for the fminunc is far less efficient than the spq/interior point approaches implemented in the fmincon algorithm. Both Fmincon and Fminunc are gradient-based algorithms, but Fminunc can only implement non-linear constraints through penalty terms. Fig. 7 illustrates that Fminunc always fails for small grids ($N \leq 10$), and that a 5% success rate was obtained for larger grids ($25 < N \leq 30$). Implementing constraints through penalty terms in gradient-based algorithms, such as Fmincon and Fminunc, create problems due to the numerical noise generated by the ΔT_{pinch} approximations. That is, the penalty term, which includes HE approximation, is added directly to the parameter being optimized (COP), and such noise affects the finite-difference approximations. HE noise is reduced if more accurate ΔT_{pinch} approximations are applied, however, even highly accurate 'Hybrid' HE schemes with $N \approx 30$ is insufficient for Fminunc. Note that penalty terms can also be used by Fmincon, instead of using the default constrained methods available in Fmincon. However, such Fmincon optimizations were not included in this study since they, like Fminunc, were highly unsuccessful.

Fminsearch is relatively successful at finding a rough estimate of the simplest heat pump (A), but if a HE is added (heat pump B), the success rate is reduced from approximately 60% to less than 5%, and if an ejector is added the success rate is 0%. Fig. 7 shows that Fminsearch success rate is always less than all the different Fmincon methods, while the Fminsearch run time is typically three times longer than Fmincon. Hence, Fmincon is much better than Fminsearch for the problems studied here. However, Fminsearch is less affected by noise since is doesn't compute gradients and could be better than Fmincon if inaccurate process models or fluid packages are applied (see discussion in Section 4.5). Also note that Fminsearch to some extent is able to obtain better results, it is therefore often possible to improve the success rate by using an optimization sequence, such as in Fmincon(All), or calling Fminsearch multiple times using the 'GlobalSearch' function in MATLAB.

Fmincon is the only optimization algorithm in this study that can implement non-linear constrains directly. Fmincon is typically 5 – 10 times faster than the PS(50). Fig. 7 shows that forward and central finite difference methods overall have similar success rates and run time. The optimization algorithm applied, i.e., sqp or interior, has a much larger

impact on the success rate and run time. The sqp method is often the best, i.e., both faster and more accurate. In general, the interior approach is less efficient finding accurate solutions with COP error $<$ 0.01%. Fig. 7 shows that Fmincon(sqp) is the best optimization method for the most simple heat pumps (heat pump A), however, if the complexity of the heat pump is increased with either an additional internal HE or an ejector, the sequence based Fmincon(All) method is better. A more detailed discussion of the best optimization schemes is given in Section 4.4.

4.4. Recommendations

Recommended optimization schemes, based on optimization run time and success rates are presented in Table 8. The results shows that Fmincon based optimizations are recommended for all the SHIs. That is, Fmincon obtain the best success rate with respect to run time, as discussed below. Fig. 7 shows that Fmincon(sqp) is the best optimization method for the simplest heat pump designs with two HEs, where it obtains a 100% success rate for $N > 10$. Fmincon(sqp) is about 3 times faster than Fmincon(All), and more than 10 times faster than PS(50).

4.4.1. Adding a heat exchanger (Heat pump B)

Fig. 7 shows that the sequence based Fmincon(All) is the best approach for heat pumps with three HEs, especially when less accurate optimizations are required. Fmincon(All) is about 5 times faster than PS (50). Fmincon(All) has a 100% success rate finding solutions with COP error less than 0.1% for $5 < N \leq 10$. HE grid size $15 < N \leq 20$ is most successful when generating more accurate results with COP error $<$ 0.01%, but such Fmincon(All) shemes have a success rate less than 90%. The low success rate, compared to the other heat pumps designs where Fmincon(All) obtains a 100% success rate, perhaps indicates that a slightly larger Fmincon sequences, i.e. more than four Fmincon calls, could be used for heat pump B. The low success rate can be explained by difficulties optimizing Case 5 with CO_2, where only 5% of Fmincon(sqp) and less than 20% of PS(50) were successful. Fig. 7 illustrates that adding a HE reduces the optimization success rate for a single Fmincon (sqp). However, the sequence based search Fmincon(All) is less affected by a third HE. Also note that if a third HE is added, PS(50) run time increase with 104%, while Fmincon(All) run time only increase with 68%. This perhaps indicates that using larger sequences is the best optimization approach for systems with even more HEs.

Fig. A1 shows that one Fmincon(sqp, central) and one Fmincon(sqp, forward) runs with $10 < N \leq 30$ managed to optimized Case 5 with CO_2 with COP error $<$ 0.01%, and that the computer run time were 15 and 25 s, respectively. That is, 90 different Fmincon(spq) runs are required to obtain a statistical success rate of $(1-(1-0.05)^{90})\cdot 100\ \% \approx\ 99\ \%$, with an overall run time of approximately 1800s. Fig. 5 shows that PS (50) requires around 400 s run time to obtain a 20% success rate. That is, 21 different PS(50), with overall run time around 8400 s, are required to obtain a statistical success rate of $(1-(1-0.20)^{21})\cdot 100\ \% \approx 99\ \%$. Hence, PS(50) requires around 5 times more run time than Fmincon optimization to obtain a 99% success rate. The random starting points, defined in Table 3, are also important. Since both Fmincon(sqp, central) and Fmincon(sqp, forward) produced successful results, there must also be starting points where Fmincon(All) is successful. Fig. 5 shows that all Fmincon(All) optimizations of Case 5 with CO_2 were relatively well optimized for $N > 5$, i.e. obtaining a 100% success rate for finding solutions with COP error $<$ 0.1%. Also note that the optimizations of Case 5 with CO_2 represent 12.5% of the 40 cases optimized with $15 < N \leq 20$, hence, the Fmincon(All) success rate for heat pump B is 100% for all optimizations other than Case 5 with CO_2 (see Fig. 7).

4.4.2. Adding an ejector (Heat pump C)

Fmincon(All) is the best approach and obtains a 100% success rate for $5 < N \leq 10$, even when finding accurate solutions with COP error $<$ 0.01% (see Fig. 7). PS(50) and GA often obtain inaccurate results with COP error $>$ 1% that are far from optimized. For systems with an ejector, Fmincon(sqp) is much better than Fmincon(interior), however the success rate with Fmincon(All) is typically the best. The statistically based methods (PS and GA) require about 10 times as much run time than Fmincon(All), while the success rate is less than 80%. Note that if an ejector is added, PS(50) run time increase with 250%, while the Fmincon(All) run time only increase with 33%.

4.4.3. Optimization of complex SHIs

Fig. 7 illustrates that the run time increases significantly if a third HE or an ejector is added, showing the importance of using efficient schemes for more complex SHIs. The most complex system modeled here has only 4 optimization variables and 3 equipment constraints, while e.g., Fahr et al. (2022) used 91 optimization variables for a regenerative Rankine cycle. More complex systems with additional HEs, refrigerant mixtures and optimal heat integration of multi-streams HEs can deal with different potential sources of numerical noise which are difficult to quantify. The review by Rao et al. (2020) concluded that traditional optimization techniques, such as gradient-based methods, usually fail to solve large-scale non-linear SHI problems with local minima. However, Fmincon methods such as sqp have been used to model complex SHIs (Espatolero et al., 2014), and are also found to be the best approach for the simpler SHIs modeled here. The trend discovered, is that the Fmincon(All) run time is significantly less affected by additional equipment. For example, the results show that the Fmincon(All) run time only increase with 68% when adding a HE and 33% when adding an ejector, while the PS(50) run time increase with 104% and 250%, respectively. This trend could indicate that nonlinear-constrained gradient-based optimization sequences, such as Fmincon(All), also will be the best approach for more complex SHIs. However, more research is needed to determine if heat pumps with two or three HEs are representative of 'systems with multiple heat exchangers' in general, or if it is possible to extrapolate observations from such simple processes to systems with more HEs.

4.5. Process modeling precision versus optimization success

In this study, the process value being optimized (COP) is calculated directly without using equipment approximation, instead such approximations are introduced through optimization constraints. Unlike the ejector efficiency constraints, the HE performance constraints introduce numerical noise, due to numerical ΔT_{pinch} approximations. Deterministic solvers use a fixed strategy, where the whole optimization search can fail if only one single improvement is missed due to numerical noise. However, the HE accuracy is also important for stochastic optimization algorithms, as illustrated in Fig. 7. It is important to find the ΔT_{pinch} and COP accuracy level required by different optimization algorithms. Optimization algorithms based on gradient calculations are highly sensitive to noise. Fig. 7 shows that Fminunc is the least successful method, due to the penalty-noise problem that derives for unconstrained gradient-based methods. Fmincon was also tested using penalty terms instead of non-linear constrains, however, this unconstrained gradient-based approach was also highly unsuccessful and therefore not investigated further. The finite difference step-size used by Fmincon and Fminunc can be increased to make them less sensitive to noise, but such schemes will require more optimization steps since gradient approximations will be less accurate.

4.5.1. Numerical noise generated by fluid property packages

The noise in the COP and ΔT_{pinch} values are directly related to the numerical approximations implemented in the fluid property package. In this work only SHIs with pure refrigerants are studied using CoolProp. For gradient methods, it is important to avoid unnecessary numerical noise in the objective function (COP). Hence, the great success of Fmincon indicates that the numerical noise introduced by CoolProp is

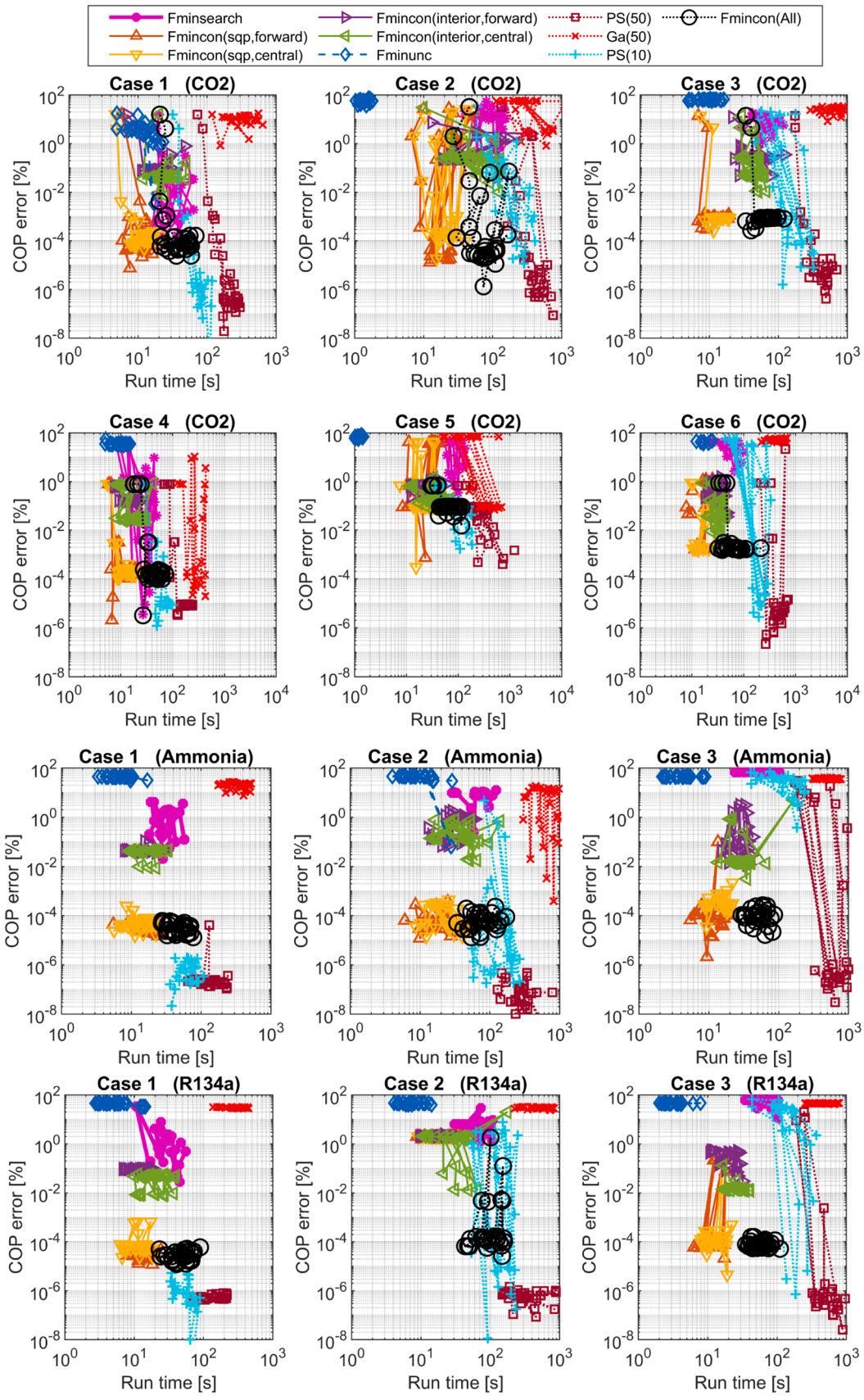

Fig. A1. Errors in optimized COP values with respect to computer run time for schemes with $N = 2, 3, \ldots, 30$.

relatively small. However, fluid properties of pure refrigerants are easier to compute accurately than refrigerant mixtures (Brodal et al., 2023), but Fmincon can also handle refrigerant mixture problems. For example, Fmincon has previously been applied when modeling mixed refrigerants SHIs used the fluid property package TREND (Brodal and Eiksund 2020).

4.5.2. Numerical noise generated by equipment models

In this study the main source of numerical noise is generated by the HE approximations, while the other process parameters, such as COP, are calculated accurately from fluid property package values. Some effort was put into modeling COP with as little noise as possible, e.g., modeling the ejector performance through optimization constraints. However, conventional process modeling programs primally designed to solve complex processes that do not involve optimization, such as HYSYS, are often calculating process parameter with a precision relevant for engineers to save computational time, e.g., using fast iterative solvers that calculate COP with only an 0.1% accuracy (Brodal et al., 2019). However, such a low precision can make gradient approximations useless, and can also cause problems for other optimizations schemes such as Fminsearch and PS (Brodal et al., 2019). More work is needed to investigate the optimal modeling accuracy for different optimization algorithms.

5. Conclusion

The accuracy of the process model affects the optimization results and the optimization run time. The ΔT_{pinch} HE constraints introduce numerical approximations in the optimization problem, which can be a dominant source of noise causing optimization algorithms to fail or be inefficient. Calculating ΔT_{pinch} accurately requires extra computational work, but this extra effort can reduce the overall workload in optimizations. HE approximations are time-consuming, hence, finding a fast ΔT_{pinch} scheme is important. A novel hybrid ΔT_{pinch} approximation, that combines low and high-order interpolation methods, is compared with the conventional 'Grid only' approach which does not involve any interpolation. The main finding is that the 'Hybrid' method is much better than conventional schemes when optimizing SHIs. Fig. 4 and Table 8 show that the novel 'Hybrid' method is 5 – 10 times faster than conventional 'Grid only' method to obtain accurate results that can be used in sensitivity studies and in detailed equipment level design work. Fig. 7 shows that the optimal number of grid points for the 'Hybrid' method is $5 < N \leq 20$, however, $N \approx 10$ is a relatively efficient grid size for all the different SHI designs.

Fmincon, which is a gradient-based method that can solve non-linear constrained optimization problems directly, is found to be the best optimization method for all the SHIs modeled here. Fmincon(spq) typically optimizes simple SHIs, with only two HE, accurately and is more than 10 times faster than PS, which is the best stochastically method identified. Single Fmincon optimizations are less successful for more complex SHIs with three HEs or an ejector. However, the Fmincon (All) sequence is still successful, and is about 5–10 times faster than PS (50). Fig. 7 shows that GA, which is perhaps the most popular SHI optimization method (Austbø et al., 2014), is both slow and inaccurate. For example, a sequence based Fmincon optimization using a 'Hybrid' ΔT_{pinch} model can potentially be 25 – 100 times faster than a PS scheme using 'Grid only' approximations, and perhaps more than 250 times faster than a conventional GA scheme using 'Grid only', since GA schemes must be called numerous times to obtain a sufficient success rate. That is, a sensitivity study that takes weeks to compute with conventional stochastic methods, such as Brodal and Jackson (2019), can be executed in hours with 'Hybrid' Fmincon based schemes.

It has often been argued, e.g., by Rao et al. (2020), that methods such as PS and GA must be used to solve large-scale SHI problems with local minima. However, this is not obvious from the results presented here. That is, PS is highly successful for SHIs without ejectors but requires much more computation time than Fmincon. The results also show that when the system complexity increases, adding a HE or an ejector, the run time increases more for PS than for Fmincon(all). This could indicate that sequences of non-linear constrained gradient-based optimization schemes also are the best alternative for more complex SHIs. The results also show that a single Fmincon(spq) can optimize the simplest system (heat pump A), while it is necessary to use a sequence multiple Fmincon calls to optimize the more complex systems (heat pump B and C). This could indicate that longer optimization sequences, than the four used by Fmincon(All), should be applied when studying SHIs with higher complexity. The methods recommended for the SHI modeling can perhaps be generalized for solving semi-infinite programs. It is also reasonable to assume that the findings are relevant for other applications such as HEN optimization, path constraints in dynamic optimization, and temperature constraints in plug flow reactors. However, more research is needed to answer such questions.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

CRediT authorship contribution statement

Eivind Brodal: Conceptualization, Methodology, Software, Formal analysis, Data curation, Writing – original draft, Visualization, Investigation. **Steven Jackson:** Writing – review & editing. **Oddmar Eiksund:** Methodology, Software, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Appendix

References

Allen, B., Savard-Goguen, M., Gosselin, L., 2009. Optimizing heat exchanger networks with genetic algorithms for designing each heat exchanger including condensers. Appl. Therm. Eng. 29 (16), 3437–3444.

Austbø, B., Løvseth, S.W., Gundersen, T., 2014. Annotated bibliography—use of optimization in LNG process design and operation. Comput. Chem. Eng. 71, 391–414.

Ayub, Z.H., Khan, T.S., Salam, S., Nawazc, K., Ayub, A.H., Khan, M.S., 2019. Literature survey and a universal evaporation correlation for plate type heat exchangers. Int. J. Refrig. Revue Int. Du Froid 99, 408–418.

Bell, I.H., Wronski, J., Quoilin, S., Lemort, V., 2014. Pure and pseudo-pure fluid thermophysical property evaluation and the open-source thermophysical property library CoolProp. Ind. Eng. Chem. Res. 53 (6), 2498–2508.

Bouabidi, Z., Katebah, M.A., Hussein, M.M., Shazed, A.R., Al-musleh, E.I., 2021. Towards improved and multi-scale liquefied natural gas supply chains: thermodynamic analysis. Comput. Chem. Eng. 151, 107359.

Brodal, E., Eiksund, O., 2020. Optimization study of heat pumps using refrigerant blends – Ejector versus expansion valve systems. Int. J. Refrig. 111, 136–146.

Brodal, E., Jackson, S., 2019. A comparative study of CO_2 heat pump performance for combined space and hot water heating. Int. J. Refrig. 108, 234–245.

Brodal, E., Jackson, S., Eiksund, O., 2019. Performance and design study of optimized LNG mixed fluid cascade processes. Energy 189, 116207.

Brodal, E., Jackson, S., Hailu, G., 2023. UA and pinch point temperature difference modeling — finding the best heat exchanger schemes. Comput. Chem. Eng. 169, 108085.

Byrd, R.H., Hribar, M.E., Nocedal, J., 1999. An interior point algorithm for large-scale nonlinear programming. SIAM J. Optim. 9 (4), 877–900.

Cassanello, M., Liang, Y., Hui, C.W., 2020. A shortcut method for simultaneous energy and heat exchange area optimization with variable stream conditions. Appl. Therm. Eng. 175, 115363.
Dai, B., Dang, C., Li, M., Tian, H., Ma, Y., 2015. Thermodynamic performance assessment of carbon dioxide blends with low-global warming potential (GWP) working fluids for a heat pump water heater. Int. J. Refrig. 56, 1–14.
Dimian, A.C., Bildea, C.S., Kiss, A.A., 2014. Chapter 13 - pinch point analysis. In: Computer Aided Chemical Engineering, 35. Elsevier, pp. 525–564. A.C. Dimian, C.S. Bildea and A.A. Kiss.
Ding, H., Sun, H., Sun, S., Chen, C., 2017. Analysis and optimisation of a mixed fluid cascade (MFC) process. Cryogenics 83, 35–49 (Guildf).
Duran, M.A., Grossmann, I.E., 1986. Simultaneous optimization and heat integration of chemical processes. AIChE J. 32 (1), 123–138.
Elbel, S., Hrnjak, P., 2008. Experimental validation of a prototype ejector designed to reduce throttling losses encountered in transcritical R744 system operation. Int. J. Refrig. 31 (3), 411–422.
Eldeeb, R., Aute, V., Radermacher, R., 2016. A survey of correlations for heat transfer and pressure drop for evaporation and condensation in plate heat exchangers. Int. J. Refrig. 65, 12–26. Revue Internationale Du Froid.
Elias, A.M., Giordano, R.d.C., Secchi, A.R., Furlan, F.F., 2019. Integrating pinch analysis and process simulation within equation-oriented simulators. Comput. Chem. Eng. 130, 106555.
Espatolero, S., Romeo, L.M., Cortés, C., 2014. Efficiency improvement strategies for the feedwater heaters network designing in supercritical coal-fired power plants. Appl. Therm. Eng. 73 (1), 449–460.
Fahr, S., Mitsos, A., Bongartz, D., 2022. Simultaneous deterministic global flowsheet optimization and heat integration: comparison of formulations. Comput. Chem. Eng. 162, 107790.
Fu, C., Gundersen, T., 2016. Heat and work integration: fundamental insights and applications to carbon dioxide capture processes. Energy Convers. Manag. 121, 36–48.
Gao, K., Wu, J., Bell, I., Lemmon, E., 2020. Thermodynamic properties of ammonia for temperatures from the melting line to 725K and pressures to 1000MPa. J. Phys. Chem. Ref. Data.
He, Y., Shebert, G.L., Chimowitz, E.H., 2015. An algorithm for optimal waste heat recovery from chemical processes. Comput. Chem. Eng. 73, 17–22.
Hesthaven, J.S., Gottlieb, S., Gottlieb, D., 2007. Spectral Methods for Time-Dependent Problems. Cambridge University Press, Cambridge. Cambridge.
Kemp, I.C., 2007. Preface. Pinch Analysis and Process Integration. Butterworth-Heinemann, Oxford (Second Edition). I.C. Kempxiv-xv.
Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of the ICNN'95 - International Conference on Neural Networks.
Kim, S.Y., Bagajewicz, M., 2016. Global optimization of heat exchanger networks using a new generalized superstructure. Chem. Eng. Sci. 147, 30–46.
Lemmon, E.W., McLinden, M.O., Wagner, W., 2009. Thermodynamic properties of propane. III. A reference equation of state for temperatures from the melting line to 650K and pressures up to 1000MPa. J. Chem. Eng. Data 54 (12), 3141–3180.
Levy, S., 1991. Genetic algorithms in search optimization and machine learning. New Whole Earth LLC 126.
Li, B., a. Deng, Y., Li, Z., Xu, J., Wang, H., 2022. Thermal-economy optimization for single/dual/triple-pressure HRSG of gas-steam combined cycle by multi-objective genetic algorithm. Energy Convers. Manag. 258, 115471.
Linnhoff, B., Flower, J.R., 1978. Synthesis of heat exchanger networks: I. Systematic generation of energy optimal networks. AIChE J. 24 (4), 633–642.
Ma, Y., McLaughlan, M., Zhang, N., Li, J., 2020. Novel feasible path optimisation algorithms using steady-state and/or pseudo-transient simulations. Comput. Chem. Eng. 143, 107058.
MATLAB (2020). 9.8.0.1323502 (R2020a), The MathWorks Inc.
Murr, R., Thieriot, H., Zoughaib, A., Clodic, D., 2011. Multi-objective optimization of a multi water-to-water heat pump system using evolutionary algorithm. Appl Energy 88 (11), 3580–3591.
Nelder, J.A., Mead, R., 1965. A simplex method for function minimization. Comput. J. 7 (4), 308–313.
Powell, M.J.D., 2006. A Fast Algorithm for Nonlinearly Constrained Optimization Calculations. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 144–157. Berlin, Heidelberg.
Rao, R.V., Saroj, A., Oclon, P., Taler, J., 2020. Design optimization of heat exchangers with advanced optimization techniques: a review. Arch. Comput. Methods Eng. 27 (2), 517–548.
Ren, L., 2020. Optimization and comparison of two combined cycles consisting of CO_2 and organic trans-critical cycle for waste heat recovery. Energies 13 (3), 724.
Sarkar, J., Bhattacharyya, S., 2009. Assessment of blends of CO_2 with butane and isobutane as working fluids for heat pump applications. Int. J. Therm. Sci. 48 (7), 1460–1465.
Shampine, L.F., 2008. Vectorized adaptive quadrature in MATLAB. J. Comput. Appl. Math. 211 (2), 131–140.
Shanno, D.F., 1970. Conditioning of quasi-newton methods for function minimization. Math. Comput. 24 (111), 647–656.
Span, R., Wagner, W., 1996. A new equation of state for carbon dioxide covering the fluid region from the triple-point temperature to 1100K at pressures up to 800MPa. J. Phys. Chem. Ref. Data 25 (6), 1509–1596.
Spellucci, P., 1998. A new technique for inconsistent QP problems in the SQP method. Math. Methods Oper. Res. 47 (3), 355–400 (Heidelberg, Germany).
Tillner-Roth, R., Baehr, H.D., 1994. An International standard formulation for the thermodynamic properties of 1,1,1,2-tetrafluoroethane (HFC-134a) for temperatures from 170K to 455K and pressures up to 70MPa. J. Phys. Chem. Ref. Data 23 (5), 657–729.
Trefethen, L.N., 2000. Spectral Methods in MATLAB. SIAM, Philadelphia, PA.
Watson, H.A.J., Khan, K.A., Barton, P.I., 2015. Multistream heat exchanger modeling and design. AIChE J. 61 (10), 3390–3403.