Computers and Chemical Engineering 169 (2023) 108085

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

UA and pinch point temperature difference modeling — Finding the best heat exchanger schemes

Eivind Brodal[a,*], Steven Jackson[a], Getu Hailu[b]

[a] Engineering Science and Safety IVT, UiT – The Arctic University of Norway, Norway
[b] Department of Mechanical Engineering, University of Alaska Anchorage, United States

ARTICLE INFO

Keywords:
Heat exchanger
High-order
Interpolation
Chebyshev
UA-value
Pinch point

ABSTRACT

Process models using simplified heat exchanger (HE) models are often analyzed using methods based on derivatives or optimization procedures, where even small numerical errors can cause algorithms to fail. This article explores the use of numerical approximations for calculating pinch temperature and UA-value, including novel high-order polynomial methods using equidistant and Chebyshev grids. The results show that the mainstream methods, where LMTD and pinch temperature are calculated from grid values, are 2–5 times slower than the high-order methods if requiring accuracy better than 1%. If a 0.01% accuracy is needed high-order methods are often 10–20 times faster. Numerical errors in high-order schemes with pure fluids converge quickly to zero when increasing the grid size, and schemes with more than 30 grid points generate errors less than 1E-4%. High-order methods were less successful for fluid mixtures, where a novel hybrid high and low-order pinch temperature scheme is recommended.

1. Introduction

Heat exchangers (HEs), including evaporators and condensers, are key components in numerous process applications. Heat pumps and steam cycles are examples of heat exchanger networks (HENs) with multiple HEs, and more HEs are often introduced in such systems to maximize energy efficiency. There are many different types of HEs, e.g., plate-fin heat exchangers (PFHEs), shell-and-tube heat exchangers (STHEs) and fin-and-tube heat exchanger (FTHEs), and their size and design have a direct impact on the system performance, such as the compressor work in heat pump cycles. The rate of heat transfer depends on HE design-parameters such as geometry, size and construction material, as well as fluid properties, inlet temperatures and operating pressure. All these parameters can be incorporated into numerical HE models described by the heat transfer area (A) and the heat transfer coefficient (U), e.g., including parameters such as flow velocities, film coefficients and material thermal conductivity. Many different HE models have been suggested, such as the heat transfer plate HE models for supercritical fluids by Forooghi and Hooman (2014). The literature survey by Ayub et al. (2019) lists 22 different heat transfer models just for evaporation in plate HEs. The heat transfer and pressure drop review article by Eldeeb et al. (2016) describes 17 evaporation and 8 condensation models for plate HEs. Such HE models can be used for identifying suitable HE designs for a given process. HE models are also important for optimizing system performance in more complex HENs, which is often applied when comparing processes with different equipment with respect to energy usage and cost (Austbø et al. 2014). In many multi-variable optimization studies of HEN systems, detailed HE models are considered too complex and difficult to solve. Simplified models, excluding all information about exchanger geometry and materials have therefore gained much attention. The temperature profile inside HEs ($\Delta T = T_{\text{source}} - T_{\text{sink}}$), i.e., the internal temperature differences between the warm heat source and the cold heat sink, is central in different HE models. This work focuses on two simplified heat exchanger models based on the pinch point temperature difference, here named temperature pinch: $\Delta T_{\text{pinch}} = \min(\Delta T)$ and the UA-value, where the UA-value is given by an integral expression with $1/\Delta T$ as the integrand. These two simplified HE models require far fewer numerical computations, which is convenient in multi-variable optimization problems where the HE model must be solved in each optimization-step. Because heat transfer is proportional to ΔT, HEs operating with $\Delta T_{\text{pinch}} \rightarrow 0$ K are theoretically optimized. However, the region around the pinch point requires a large HE area for given heat transfer duty, and ΔT_{pinch} is therefore often recognized as the bottleneck, and therefore sometimes can be directly related to HE area (A) and cost. Therefore, ΔT_{pinch} has

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

https://doi.org/10.1016/j.compchemeng.2022.108085
Received 30 November 2021; Received in revised form 23 June 2022; Accepted 25 November 2022
Available online 28 November 2022

Nomenclature

A	Heat transfer area [m^2]
D	Spectral differentiation matrix [-]
h	Specific enthalpy [kJ kg^{-1}]
i	Heat exchanger section number i [-]
j	Node number in a heat exchanger section [-]
$\dot{m}$	Mass flow [g s^{-1}]
N	Total number of grid points [-]
$N_{\mathrm{grid},i}$	Number of grid points in HE section i [-]
n	Polynomial order of approximation [-]
p	Pressure [bar]
Q	Heat exchanger duty [kW]
q	Polynomial representation of Q [kW]
T	Temperature [°C]
ΔT	Temperature difference [°C]
$\Delta T_{\mathrm{pinch}}$	Minimum temperature difference [K]
U	Heat transfer coefficient [kW m^{-2} K^{-1}]
UA	Thermal conductance [kW K^{-1}]
$x_i(j)$	Grid point j in section i, where $x_i \in$ [-1,1]

Subscripts & Superscripts

ex	Extremal values
min	Minimum
pinch	Pinch point
source	Heat source fluid
sink	Heat sink fluid

Abbreviation

HE	Heat exchanger
HEN	Heat exchanger network
FTHE	Fin-and-tube heat exchanger
LNG	Liquefied natural gas
LMTD	Logarithmic mean temperature difference
PFHE	Plate-fin heat exchanger
STHE	Shell-and-tube heat exchangers

been used to model and improve complex processes such as liquefied natural gas (LNG) value chains (Bouabidi et al., 2021) and to find optimal waste heat recovery from chemical processes (He et al., 2015); to find optimal working fluid mixtures in simple systems such as heat pumps heating water (Sarkar and Bhattacharyya, 2009; Dai et al., 2015) and more complex LNG processes. UA calculations are also used when modeling U and the heat transfer for real HE designs where A is known, e.g., using heat transfer equations discussed by Eldeeb et al. (2016) and Ayub et al. (2019). $\Delta T_{\mathrm{pinch}}$ can also be implemented in all HE models to avoid unphysical temperature crossings, i.e., a constraint $\Delta T_{\mathrm{pinch}} > 0$. Watson et al. (2015) combines logarithmic mean temperature difference (LMTD) based UA calculations to model multi-stream HE area A as an extension to the pinch analysis. Elias et al. (2019) combines temperature pinch modeling with UA based on LMTD calculations to estimate HE area A and cost. Chen (2016) studied CO_2 gas coolers by modeling UA numerically, and based on the results, concluded that the HE could be undersized by a factor of 30–60 % if LMTD methods were applied.

The UA and $\Delta T_{\mathrm{pinch}}$ estimation methods that are the focus of this study are investigated using simple heat exchanger models (pure countercurrent flow configuration). Such simplified equipment models are often used in concept phase optimization studies that have the aim of finding the best set of operating parameters for a process and not necessarily the best design for an individual exchanger. That is, the methods set out in this work are not intended for application in detailed (equipment level) design work, rather they aim to set the optimum starting point for that work. When using simplified equipment models, it is natural to think that there is no reason to invest calculation effort in estimating performance parameters precisely. But this is not necessarily true when UA or $\Delta T_{\mathrm{pinch}}$ estimates are used in complex optimization. In many cases, more accurate estimates allow faster solution of optimization problems, especially when finite difference methods are used to find derivatives, but also when optimization algorithms without gradient calculations are used. The main problem is not the accuracy of the analytical equipment models, but the numerical approximations used to solve them which generate noise. That is, calculating UA or $\Delta T_{\mathrm{pinch}}$ using more accurate numerical approximations can speed up optimization searches. In optimization there are also other reasons for operating with a higher accuracy than the low-fidelity process models used, for example, it can be helpful to understand if a process has been correctly optimized by comparing results from different optimization runs since they will be more likely to tend to the same solution. It is also difficult to understand how the different simplified equipment models affect the overall process, but more accurate optimizations are probably more likely to find better process parameters to use in the equipment level design work.

Calculating UA or $\Delta T_{\mathrm{pinch}}$ accurately requires extra computational work, but such extra work can reduce the overall workload in optimization studies since HE inaccuracies introduce numerical noise that can cause the optimization search to fail or be inefficient. Even in the modeling of heat pump HENs that have relatively low complexity, numerical noise in combination with $\Delta T_{\mathrm{pinch}}$ constraints can create problems for both deterministic and statistically based optimization methods (Brodal and Jackson, 2019; Brodal et al., 2019). Optimization studies often focus on solving very complex HEN problems, e.g., found in the LNG industry (Austbø et al., 2014), which often deal with many potential sources of numerical noise, such as the use of multi-stream HEs with refrigerant mixtures, which are difficult to quantify. This perhaps explains why researchers in the past have focused on the development of efficient optimization approaches rather than the development of fast high-order numerical HE schemes that generate less noise for an equivalent number of computations, e.g., the review article by Rao et al. (2020) that summarizes advanced optimization techniques that have been applied successfully to solve HEN problems. Like all other temperature pinch related literature found by the authors, Rao et al. (2020) focuses on the optimization algorithms to explain how to best optimize HENs. The accuracy of the heat exchanger model and how this affects the optimization results is not even mentioned in this review.

High-order (spectral) interpolation methods have been used for decades (Hesthaven et al., 2007), e.g., to solve partial differential problems quickly as discussed by Trefethen (2000). A Chebyshev grid distribution is well suited for high-order interpolation of smooth functions since these grid-points cluster at the ends (Trefethen, 2000). However, for non-smooth functions, high-order methods can be unreliable. The authors have not been able to find any articles exploring, or even mentioning, high-order methods for computing temperature pinches, and it is not mentioned in any literature referenced in this article. It is also omitted in the literature that presents an overview of work in this field such as the book 'Pinch Analysis and Process Integration: A User Guide on Process Integration' (Kemp 2007); and the book chapter “Pinch Point Analysis” by Dimian et al. (2014). Studies using more sophisticated methods to evaluate the UA-value can be found, e.g., by Chen (2016) who investigated heat pumps with a CO_2 gas cooler using a step-size algorithm to improve the numerical integration of the UA-value. Such adaptive algorithms are designed to solve all kinds of integrals using moderately high-order methods (Shampine, 2008) and are therefore not directly optimized for solving HE integrals. It is also difficult to define an integration tolerance in adaptive algorithms that

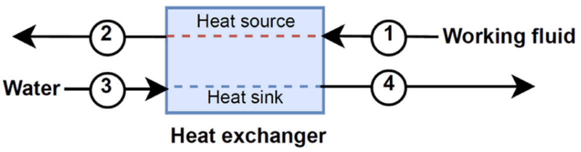

Fig. 1. Heat exchanger operating as condenser or gas-cooler.

Table 1
Input heat exchanger parameters defining Cases 1 – 6.

Input variable	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
Mass fraction CO_2 in the working fluid	1.0	1.0	1.0	0.5	0.5	0.95
Working fluid pressure ($p_1 = p_2$)	100 bar	75 bar	70 bar	50 bar	66 bar	68 bar
Outlet working fluid temperature (T_2)	25°C	25°C	25°C	20°C	20°C	20°C
Outlet water temperature (T_4)	60°C	50°C	45°C	70°C	80°C	45°C

will work well for a typical smooth problem, and at the same time does not waste a lot of time calculating special non-smooth problems. That is, spending a lot of time calculating *UA* integrals accurately does not make much sense if the numerical approximations done by the fluid property package are inaccurate. Instead, this work focuses on finding a fast and accurate scheme with a fixed approach that will be accurate for typical HE processes, while calculating special cases with enough accuracy, e.g., to avoid optimization algorithms to fail before converging. Even though accurate numerical integration methods have been applied to calculate *UA*, simplified and less accurate approaches based on the LMTD calculated between all neighboring grid points seems to be the most applied methods, e.g., by Watson et al. (2015), Elias et al. (2019) and Vikse et al. (2020). A comparative study of numerical integration methods for *UA*-calculations has not been found. That is, the state of the art *UA* or ΔT_{pinch} models have not used state of the art spectral interpolation methods.

This article explores different methods to calculate ΔT_{pinch} and *UA*, where the fluid properties are known at the inlets and outlets. The main concept investigated is that these methods can be solved faster using high-order schemes that evaluate ΔT on fewer grid points, i.e., less calls to the fluid property package which can be time consuming. The *UA*-value is often applied in engineering work since it can be further simplified into analytical expressions related to the LMTD, which is valid if the fluid's specific heat is constant during the heat exchange process and the HE flows are pure parallel (co-current or counter-current). The focus in this article, however, is to investigate numerical methods for computing ΔT_{pinch} and *UA* using both novel high-order methods and conventional low-order methods to compute the HE temperature profile (ΔT). The determination of the optimal method depends on the accuracy of the fluid property packages used in the calculations. Mixed fluid properties are mathematically more challenging to compute than for pure (one-component) fluids, and it is therefore interesting to study both. That is, also HE with mixtures where the fluid property packages introduce more numerical noise and dew/bubble point errors which can affect the accuracy of high-order methods negatively. Mixed fluids and pure fluids are computed using two different 'state of the art' properties packages: CoolProp (Bell et al., 2014) is used for pure components and TREND (Span et al., 2016) is applied for mixtures. Temperature profiles are calculated with both equidistant and Chebyshev distributed grid points, where different algorithms are applied to calculate $\min(\Delta T)$ and the integral of $1/\Delta T$. How the grid distributions and number of grid-points affect the accuracy and run time is investigated for mixtures and pure fluids separately, and the focus of this study is to compare different methods to calculate ΔT_{pinch} and *UA*. There are many articles calculating these HE parameters, but little is done in improving the methods or systematically exploring the best numerical approach, which is the focus of this article. The goal is to find ΔT_{pinch} and *UA* estimation methods that are both fast and accurate for both pure and mixed working fluids, which can be used as a basis in accurate and time-efficient HEN optimization schemes, e.g., applied to study basic heat pump and refrigeration cycles and more complex industrial systems.

2. Method

The different *UA* and ΔT_{pinch} heat exchanger models used in this study are described below. General high-order polynomial-based methods for calculating ΔT_{pinch} have not been identified in the literature. Neither have spectral methods for calculating *UA*, e.g., based on a Chebyshev distributed grid. The focus here is, therefore, to show how these methods can be implemented in standard heat transfer equations. The error and computer run time estimates used to compare the different approaches, and the HE cases being modelled, are also explained.

2.1. Heat exchanger cases

This article presents studies of several different heat exchanger cases where a working fluid is cooled by water in a heat exchanger, as shown in Fig. 1. To study the accuracy of the different HE approximation methods, both transcritical and subcritical cases are used. It is irrelevant for this study whether the HE is used in a refrigeration cycle, heat pump, steam cycle, or organic Rankine cycle. However, in refrigeration systems, working fluids are typically referred to as refrigerants.

The inlet water has temperature $T_3 = 5°\text{C}$ and pressure $p_3 = 2$ bar, while the inlet working fluid has temperature $T_1 = 100°\text{C}$. The heat transferred between the fluids in the heat exchanger, *Q*, is 12 kW in all cases. A HE pressure drop of zero bar is assumed for simplicity, i.e., $p_2 = p_1$ and $p_4 = p_3$, which is a common assumption in heat exchanger studies (Dai et al., 2015). Table 1 summarizes the six different cases that are studied, and Fig. 2 shows the temperature profiles for each case in detail. The heat exchanger is operating as a gas-cooler in Case 1 and 2, as illustrated in Fig. 3. In the other cases, it operates as a condenser. Cases 1 – 3 describe cooling of a pure CO_2 stream, while a mixture of CO_2 and propane is cooled in Cases 4 – 6. Processes around the critical point, and a working fluid mixture with a high concentration of CO_2 (Case 6) have been chosen since the accurate prediction of properties is more challenging to model, as experienced in the optimization study with mixed refrigerants by Brodal and Eiksund (2020).

2.2. Heat exchanger modeling

This section describes the different *UA* and ΔT_{pinch} models and defines the error estimates and run time smoothening approximations.

2.2.1. Fluid property calculations

Errors and numerical accuracy in the numerical routines of the fluid property packages can affect high and low-order polynomial HE schemes differently. Mixtures are generally more difficult to solve than one component fluids, hence, pure working fluids and mixtures are studied separately to test the different methods. Both CoolProp (Bell et al., 2014) and TREND (Span et al., 2016) are state of the art fluid property packages that implement the equation of state (EOS) by Span and Wagner (1996) for pure CO_2. For CO_2 and propane mixtures, they use the GERG-2008 equation of state by Kunz and Wagner (2012). TREND and CoolProp use different numerical algorithms to solve these EOSs. However, the pure fluid property outputs are almost identical for TREND and CoolProp, but for mixtures the difference is much larger, e.g., for CO_2 and propane mixtures they typically calculate enthalpy (*h*) or entropy (*s*) values with only the first five digits identical (Brodal and

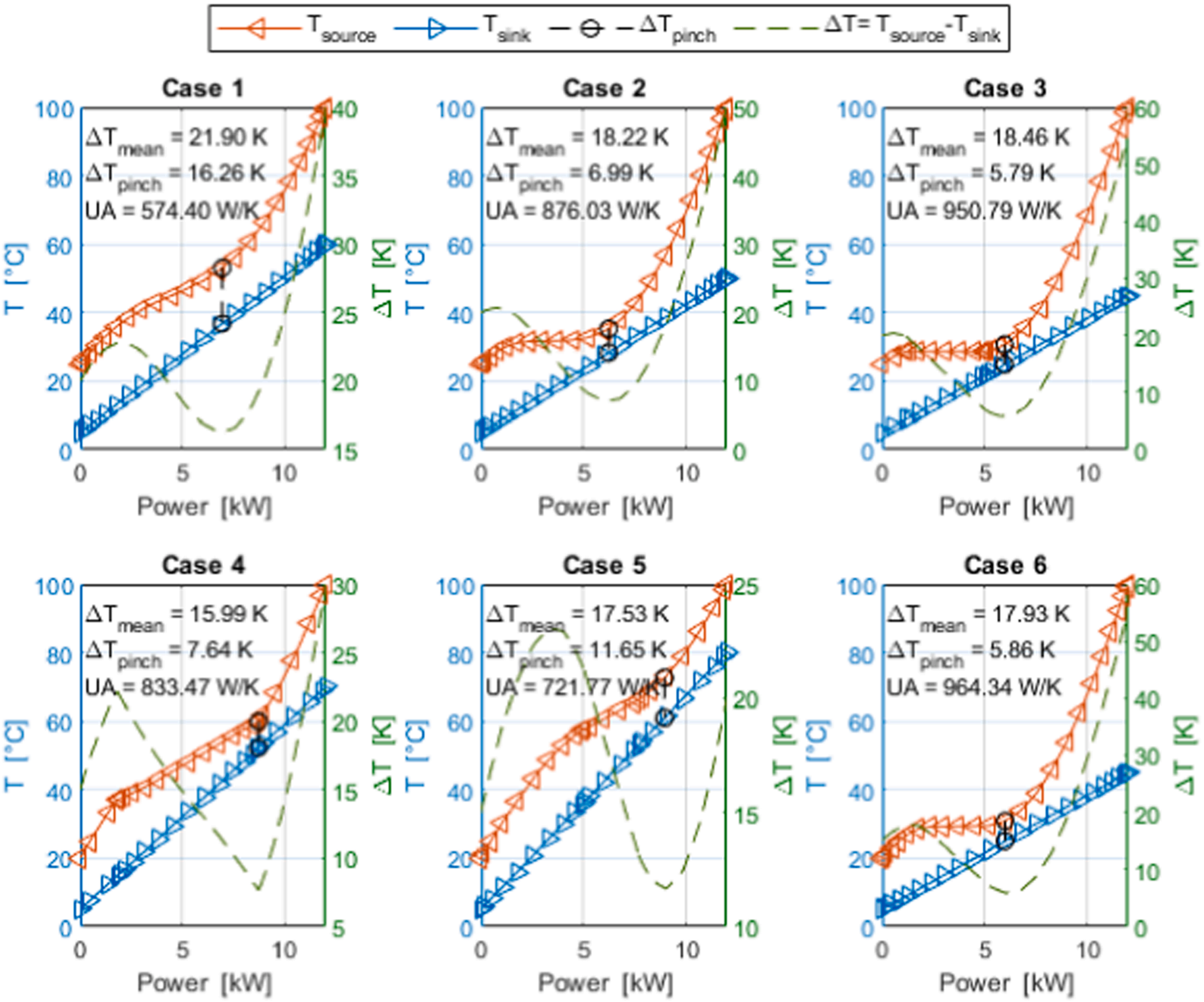

Fig. 2. Temperature profiles and temperature pinches in the heat exchanger for Case 1 – 6.

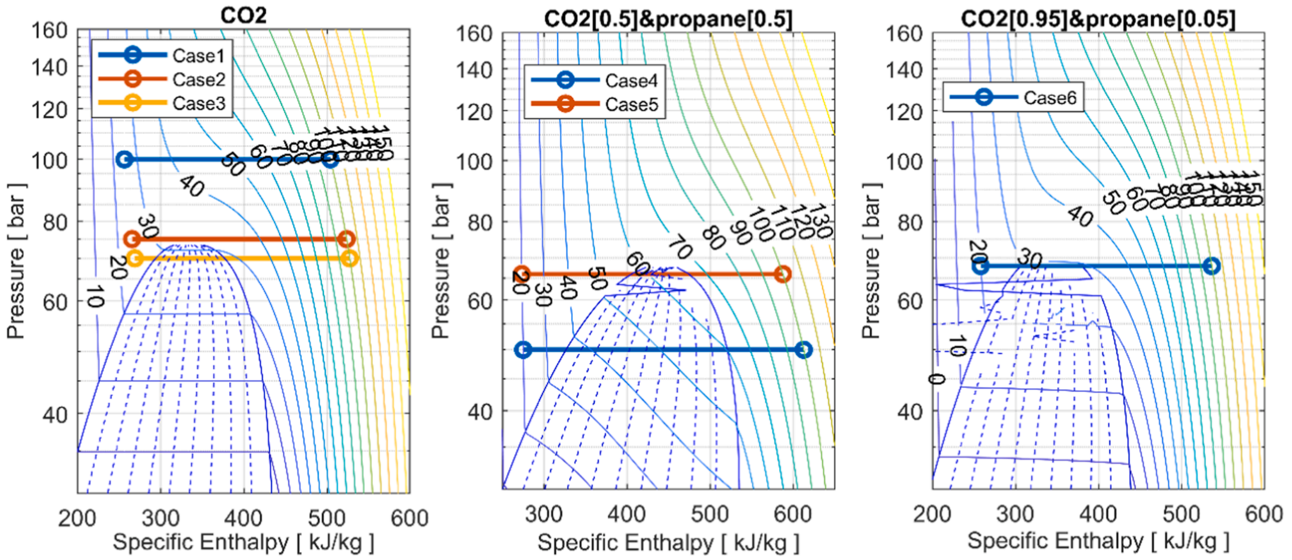

Fig. 3. Pressure-enthalpy (p-h) diagrams showing the working fluid process in the HE for Cases 1 – 6. Pure CO_2 (left), a 50 mass % mixture of CO_2 and propane (middle), and a 95 mass % CO_2 mixture with propane (right). Note that the phase envelope is not always calculated correctly for mixtures.

Eiksund, 2020), and the data presented in the results section show that high-order methods with large grids are affected by such numerical noise. That is, a relatively small random noise in ΔT can cause high-order methods to fail, while low-order methods are less affected. In many problems, high-order methods are avoided because they are less robust for such non-smooth phenomenon. For example, standard MATLAB integration methods designed to solve all kinds of problems are based on moderately high-order schemes (Shampine, 2008). Also, dew and bubble points are always found for pure fluids, but these points are more challenging to calculate for fluid mixtures, as illustrated by the phase envelope errors in Fig. 3.

In this article, pure fluid (one-component) properties are calculated with CoolProp, since CoolProp is a little faster than TREND while having similar accuracy. TREND is used for fluid mixtures since CoolProp does not have pressure (p) and enthalpy (h) as an input variable pair for mixtures, which is necessary in the temperature profile calculations.

2.2.2. Temperature grid point calculations

Specific enthalpies h are calculated using fluid property packages, with temperature and pressure as input parameters, e.g., h_1 is calculated from T_1 and p_1. For a HE with duty Q, the mass flows of the heat source and the heat sink are:

$$\dot{m}_{\text{source}} = Q/(h_1 - h_2)\mathbf{and}\dot{m}_{\text{sink}} = Q/(h_4 - h_3) \tag{1}$$

In Cases 3 – 6, the working fluid process crosses both the dew and bubble point curves during the heat exchange process, as illustrated in Fig. 3. The data presented for Case 4 in Fig. 2 also clearly illustrates that the temperature approaches (ΔT) are non-smooth functions across these points. To obtain an accurate model of ΔT in the heat exchanger, suitable for high-order methods, the exchanger must be divided into different sections (i), separated by internal bubble and dew points. The bubble and dew point enthalpies are found using the fluid property packages, hence, the specific enthalpies at the first node in each HE section, $h_{\text{source},i,1}$ and $h_{\text{sink},i,1}$, are known for the heat source and heat sink respectively, where $h_{\text{source},1,1} = h_2$ and $h_{\text{sink},1,1} = h_3$ in the first section. Each HE section has at least two grid points, and the heating duty of section i is: $\Delta Q_i = \dot{m}_{\text{source}} \cdot \Delta h_{\text{source},i}$, where $\Delta h_{\text{source},i}$ is the difference between the inlet and outlet enthalpies of section i. In the HE schemes developed here, the ratio $\Delta Q_i / Q$ is used to distribute N grid points evenly over the different sections, and the number of grid points in section i is named $N_{\text{grid,i}}$. Two different grid distributions are considered: equidistant distribution and Chebyshev distribution (Trefethen, 2000); both are defined on the interval $x_i \in$ [-1,1]. The Chebyshev grid ('Cheb') with $N_{\text{grid},i}$ nodes is defined by Eq. (2):

$$x_i(j) = -\cos\left(\frac{\pi \cdot (j-1)}{N_{\text{grid},i} - 1}\right), \mathbf{forj} = 1, \ \ldots, \ N_{\text{grid},i}. \tag{2}$$

Chebyshev points cluster at the ends, which is necessary in order to create spectral (high-order) interpolations suitable for the numerical integration and numerical derivation of smooth functions, as explained by Hesthaven et al. (2007). The working fluid process in Case 1 does not cross the dew and bubble point curves during the heat exchange process, as illustrated in Fig. 3. Hence, Case 1 has only one HE section ($i = 1$). Fig. 2 shows Case 1 with $N = N_{\text{grid},1} = 25$ Chebyshev grid points, while the HE in Case 3 is divided into three sections since the working fluid is cooled from superheated gas to subcooled liquid, see Fig. 3, but the total number of Chebyshev grid points in Case 3 is still 25: $N = N_{\text{grid},1} + (N_{\text{grid},2} - 2) + N_{\text{grid},3} = 25$, since the middle section ($N_{\text{grid},2}$) shares two nodes, one from each neighboring section. The equidistant grid points ('Equi') are defined on the interval $x_i \in$ [-1,1] as:

$$x_i(j) = 2 \cdot \left(\frac{j-1}{N_{\text{grid},i} - 1}\right) - 1, \mathbf{for}j = 1, \ \ldots, \ N_{\text{grid},i}. \tag{3}$$

The specific enthalpies in HE section (i) are not limited to interval [-1,1], but obtain the same relative grid point distribution as x_i by using the linear transformations presented in Eq. (4):

$$h_{\text{source},i}(j) = \left(\frac{x_i(j)+1}{2}\right) \cdot \left(\frac{\Delta Q_i}{\dot{m}_{\text{source}}}\right) + h_{\text{source},i,1}\mathbf{and}h_{\text{sink},i}(j) = \left(\frac{x_i(j)+1}{2}\right) \cdot \left(\frac{\Delta Q_i}{\dot{m}_{\text{sink}}}\right) + h_{\text{sink},i,1} \tag{4}$$

The temperatures $T_{\text{source},i}(j)$ and $T_{\text{sink},i}(j)$ are calculated by the fluid property package, using the source and sink pressures defined in Section 2.1 and the enthalpies calculated in Eq. (4) as input parameters. If the property package fails in the evaluation of the temperature at one of the grid points, the heat exchanger calculation is terminated, returning 'not a number'. Otherwise, the temperature difference between the heat source and heat sink is calculated as:

$$\Delta T_i(j) = T_{\text{source},i}(j) - T_{\text{sink},i}(j) \tag{5}$$

and the heating duty related to each grid point is calculated from:

$$Q_i(j) = \dot{m}_{\text{source}} \cdot \left(h_{\text{source},i}(j) - h_2\right) \tag{6}$$

A vector notation of node values in the HE section i is used later in this work, e.g., $\overrightarrow{\Delta T}_i$ and $\vec{Q}_i$.

2.2.3. Temperature pinch calculation methods

The simplest estimate of the minimum temperature difference in the heat exchanger can be made by finding the smallest value calculated in Eq. (5) on the entire HE grid:

$$\Delta T_{\text{pinch}} = \min\left(\overrightarrow{\Delta T_i}\right), \mathbf{forallHEseksjons}i \tag{7}$$

When this calculation is done from equidistant grids the method is named 'Simpl:Equi'. An alternative method is to represent the temperature approach as a set of polynomial functions with order n_i using the MATLAB-function 'polyfit':

$$q_i = \text{polyfit}\left(\overrightarrow{Q_i}, \overrightarrow{\Delta T_i}, n_i\right) \tag{8}$$

The polynomial order is chosen based on $N_{\text{grid},i}$: $n_i = \min(25, N_{\text{grid},i})$, where $n_i \leq 25$ is specified because large errors were sometimes observed in the polynomial derivation above this limit, due to the noise level introduced by the numerical accuracy of the fluid property packages. The extremal points of this polynomial, which are candidates for being the temperature pinch, are calculated from the roots of the derivative of q_i, using the MATLAB-functions 'roots' and 'polyder':

$$\overrightarrow{Q_{i,ex}} = roots(polyder(q_i)). \tag{9}$$

MATLAB could also be used directly to find the temperature differences $\Delta T_{i,\text{ex}} = \text{polyval}(q_i, Q_{i,\text{ex}})$ from the $Q_{i,\text{ex}}$ vector, but this approach was found to generate large errors in some cases. Instead, a slower, but more stable method was applied, based on the calculation of the enthalpies $h_{i,\text{ex}}$ corresponding to the different heating duty values in $Q_{i,\text{ex}}$. That is, adding more specific enthalpy points to those generated by Eq. (4), which are then used to calculate the temperature differences: $\Delta T_{i,\text{ex}} = T_{\text{source},i,\text{ex}} - T_{\text{sink},i,\text{ex}}$. Finding these temperatures requires additional calls to fluid properties packages and are therefore more time-consuming than the polyval approach. Finally, the temperature pinch was evaluated as the minimum temperature difference for all the different points:

$$\Delta T_{\text{pinch}} = \min\left(\overrightarrow{\Delta T_i}, \overrightarrow{\Delta T_{i,ex}}\right), \mathbf{for}i = 1, \ \ldots, \ N_{\text{sections}}. \tag{10}$$

Both the equidistant and Chebyshev grid methods, named 'Poly: Equi' and 'Poly:Cheb' respectively, were modeled in this way. In addition, a similar approach was developed based on 2nd-order polynomial approximation for sets of three neighboring nodes values ($j_{\min} - 1$, $j_{\min}$, $j_{\min} + 1$), where $j_{\min}$ is the node with minimum ΔT in each HE section (i). Calculations of the 2nd-order polynomial approximation were made with equidistant grid and Chebyshev grid and are named ‘Quad:Equi’ and ‘Quad:Cheb’, respectively.

High-order (spectral) interpolation methods can also be used based on analytical expressions derived from Chebyshev polynomials, and values at the Chebyshev nodes $x_i(j)$. In this article, the $(N_{\mathrm{grid},i} \times N_{\mathrm{grid},i})$ spectral differentiation matrix $\mathbf{D}_i$ was calculated in the MATLAB function named ‘gallery’. The construction of the $\mathbf{D}_i$ matrix is explained by Trefethen (2000), and is here used to calculate the derivate of ΔT with respect to the heat transfer duty Q:

$$\vec{a}_i = \left(\frac{d\ \Delta T}{dQ}\right)_{i,j} = \left(\frac{dx}{dQ}\cdot\frac{d\Delta T}{dx}\right)_{i,j} \approx \frac{Q_i\left(N_{\mathrm{grid},i}\right) - Q_i(1)}{2}\cdot\mathbf{D}_i\overrightarrow{\Delta T_i} \tag{11}$$

for HE section i, since $x_i(N_{\mathrm{grid},i}) - x_i(1) = 2$. The temperature pinch must be located at $\overrightarrow{\Delta T}_i$, or at one of the extremal points, i.e., where the derivate of ΔT is zero, which are found by solving $a_i(Q_{\mathrm{i,ex}}) = 0$. Standard MATLAB routines are used to solve this problem. In this work, the cubic function ‘spline’ is used to create a continuous polynomial function of the spectral derivative, i.e., $s_i = \mathrm{spline}(\vec{Q_i}, \vec{a}_i)$, while the extremal values are found using the function ‘roots’. To reduce run time, only regions separated by node points where a_i has different signs are examined:

$$\begin{aligned}\overrightarrow{Q_{i,ex}} &= \mathrm{roots}(s_i.\mathrm{coefs}(\mathrm{j}-1,:)) + s_i.\mathrm{breaks}(\mathrm{j}-1), \mathbf{for} j \\ &= 2,\ \ldots,\ N_{\mathrm{grid},i}\mathbf{and} a_i(j)\cdot a_i(j-1) < 0.\end{aligned} \tag{12}$$

The temperature pinch is calculated as $\Delta T_{\mathrm{pinch}}$=min $(\overrightarrow{\Delta T}_i,\ \overrightarrow{\Delta T}_{i,\mathrm{ex}})$, where $\overrightarrow{\Delta T}_{i,\mathrm{ex}}$ is found from $\vec{Q}_{i,\mathrm{ex}}$ using the same CoolProp or TREND approach as described for ’Poly:Cheb‘. This modeling approach is named ’Spec:Cheb’.

Since all the methods are based on direct calculation of the temperature approach at the extremal points ($\Delta T_{i,\mathrm{ex}}$) from the pinch point estimates of the duty, $Q_{i,\mathrm{ex}}$, i.e., not interpolated $\Delta T_{i,\mathrm{ex}}$ values, adding more $\Delta T_{i,\mathrm{ex}}$ points can only improve the accuracy of the temperature pinch model within the limits of the fluid property package. The methods described above are therefore also all combined in a hybrid modeling approach, named ‘All:Cheb’, where ‘Quad:Cheb’, ‘Poly:Cheb’ and ‘Spec:Cheb’ are computed in a sequence, and an additional 2nd-order method ‘Quad’ is performed in the end, around the smallest temperature pinch point found by all the different methods and the original Chebyshev grid.

2.2.4. Methods for UA calculation

A small amount of heat (dQ) transmitted through an infinitesimally small heat transferee area (dA), where the heat transfer coefficient (U) and the temperature difference between the fluids exchanging heat (ΔT) are both approximately constant, is given directly from the definition of heat transfer through a wall:

$$dQ = \Delta T\cdot U\cdot dA, \tag{13}$$

which can be used to calculate the UA-value (Chen, 2016) for the overall exchanger:

$$UA = \int_0^A U\cdot dA = \int_0^Q \frac{1}{\Delta T}\ dQ. \tag{14}$$

The last integral can be solved directly, e.g., using the MATLAB function ‘integral’ (Shampine, 2008), which uses a global adaptive quadrature and an input error tolerance. A similar approach has earlier been applied by Chen (2016). However, it is difficult to create a general function for integrating all kinds of integrals, e.g., with both smooth or unsmooth integrands, and at the same time keep the number of grid points in an adaptive algorithm to a minimum relative to a tolerance input. Therefore, instead of using adaptive algorithms, this work focuses on finding an optimal scheme to solve HE integrals based on general trends. The main idea is that the temperature profile in different HEs has much in common, i.e., all of the integrands in each HE section are relatively smooth, they do not contain singularities and they do contain a similar level of noise generated by the fluid property package.

The last integral in Eq. (14) can also be calculated numerically using the standard numerical trapezoidal integration method, which is a first-order (linear) polynomial scheme:

$$UA \approx \sum_i \sum_{j=2}^{N_{\mathrm{grid},i}} \frac{1}{2}\cdot\left(\frac{1}{\Delta T_i(j)} + \frac{1}{\Delta T_i(j-1)}\right)\cdot(Q_i(j) - Q_i(j-1)). \tag{15}$$

When this approach is used with an equidistant grid, it is named ‘Trap:Equi’. It is also possible to obtain high-order (spectral) accuracy by evaluating the integral numerically based on values of ΔT at Chebyshev grid nodes x_i and Chebyshev integral weights w_i, as explained by Trefethen (2000):

$$UA = \int_0^Q \frac{1}{\Delta T}\ dQ = \int_{-1}^1 \frac{1}{\Delta T}\ \frac{dQ}{dx}dx \approx \sum_i \frac{1}{\overrightarrow{\Delta T}_i}\cdot\left(\frac{Q_i\left(N_{\mathrm{grid},i}\right) - Q_i(1)}{x_i\left(N_{\mathrm{grid},i}\right) - x_i(1)}\right)\cdot\vec{w}_i, \tag{16}$$

where $x_i(N_{\mathrm{grid},i}) - x_i(1) = 2$. This approach is referred to as ‘Spec:Cheb’. The weights w_i can be calculated analytically using the properties of Chebyshev polynomials and Chebyshev nodes, and in this work the ‘clencurt’ function developed by Trefethen (2000) is used to do this. Note that the bracket in Eq. (16) is just a scaling due to the linear relation, see Eq. (4), between the heating duty Q_i and the grid points x_i. Another approach is to find a polynomial function, r_i, of order n_i of the temperature approach data using the MATLAB-function ‘polyfit’:

$$r_i = \mathrm{polyfit}\left(\vec{Q}_i,\ \frac{1}{\overrightarrow{\Delta T}_i}, n_i\right), \tag{17}$$

where the temperature differences $\overrightarrow{\Delta T_i}$ are given by Eq. (5). The polynomial order is set to $n_i = \min(20, N_{\mathrm{grid},i})$ because polynomials with $n_i >$ 20 sometimes generated large errors in MATLAB routines. Hence, the total integral is the sum of UA for each section i, and is calculated using MATLAB functions ‘diff’, ‘polyval’ and ‘polyint’:

$$UA \approx \sum_i \mathrm{diff}\left(\mathrm{polyval}\left(\mathrm{polyint}(r_i), \left[\vec{Q}_i(1),\ \vec{Q}_i\left(N_{\mathrm{grid},i}\right)\right]\right)\right). \tag{18}$$

Two different cases are studied in this work, one with equidistant grid and another one using Chebyshev grid, which are named ‘Poly: Equi’ and ‘Poly:Cheb’, respectively.

Note that, if the fluids have constant specific heat c_p during heat transfer, the equation can be further simplified using the expression ≈ $\dot{m}\cdot c_p\cdot dT$. This is known as the logarithmic mean temperature difference (LMTD) approach, where the simplest approach ‘LMTD:EndP’ is only to include temperature differences from the heat exchanger end-points (Watson et al., 2015):

$$UA \approx \frac{Q}{\mathrm{LMTD}} \approx Q\cdot\frac{\ln\left(\frac{\Delta T_{\mathrm{A}}}{\Delta T_{\mathrm{B}}}\right)}{\Delta T_{\mathrm{A}} - \Delta T_{\mathrm{B}}}, \mathbf{where}\Delta T_{\mathrm{A}} = T_1 - T_4\mathbf{and}\Delta T_{\mathrm{A}} = T_1 - T_4 \tag{19}$$

A more accurate version of this approach is to divide the exchanger into smaller elements, where the constant c_p assumption is more valid. In this article, the approach ‘LMTD:Equi’ calculates all the $(UA)_{i,j}$ values for all the neighboring nodes (j and $j-1$) for each element i, and the UA-value is given as the sum of the UA for each part of the heat exchanger:

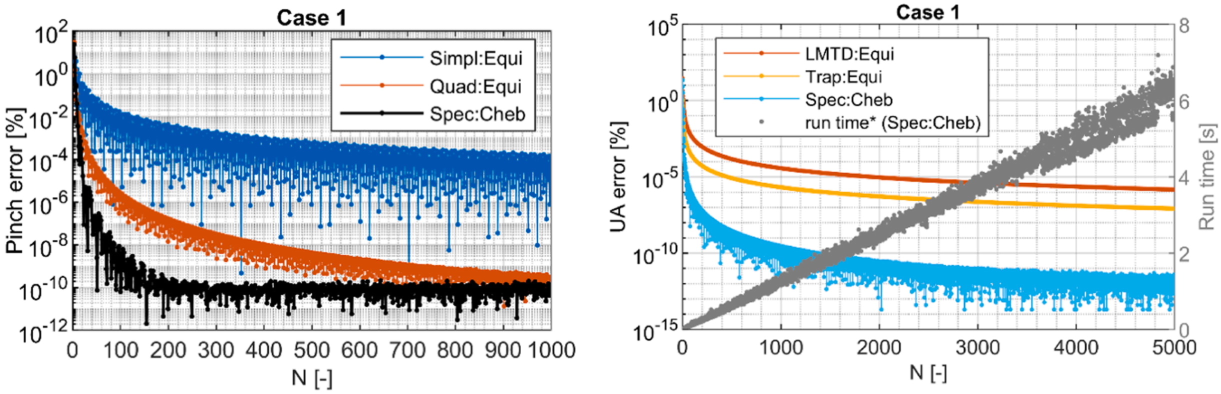

Fig. 4. Convergence test of the numerical accuracy of the temperature pinch (left) and the UA-value (right), using different methods and schemes with *N* grid points. Actual run time is illustrated in the right figure.

$$UA \approx \sum_{i=1}^{N_{\text{sections}}} \sum_{j=2}^{N_{\text{grid},i}} \left(Q_i(j) - Q_i(j-1)\right) \cdot \left(\frac{\ln\left(\frac{\Delta T_i(j)}{\Delta T_i(j-1)}\right)}{\Delta T_i(j) - \Delta T_i(j-1)} \right) \tag{20}$$

2.3. Error estimates and model performance

The accuracy of the HE models ultimately depends on the accuracy of the fluid property package predictions. These programs are designed for solving chemical engineering problems, where often a physical error less than 1 % is sufficient. However, the equations in these property packages are typically calculated at a much higher precision than 1 %, so that they can be used in more complex calculations, e.g., derivative based optimization problems. Although an analytical solution is not available, the numerical algorithm should converge towards a single solution as the number of nodes is increased. In this study, the error estimate of the numerical heat exchanger models is created using more nodes than the data presented in the figures. The ΔT_{pinch} error calculations are based on the 'All:Cheb' scheme, defined as:

$$\text{Pinch error} = \frac{\left|\Delta T_{\text{pinch}} - \left(\Delta T^{\text{All:Cheb}}_{\text{pinch},\ N=149} + \Delta T^{\text{All:Cheb}}_{\text{pinch},\ N=150}\right)\Big/2\right|}{\left(\Delta T^{\text{All:Cheb}}_{\text{pinch},\ N=149} + \Delta T^{\text{All:Cheb}}_{\text{pinch},\ N=150}\right)\Big/2} \tag{21}$$

while the *UA* error estimate is based on the 'Spec:Cheb' scheme using the definition:

$$UA \text{ error} = \frac{\left|UA - \left(UA^{\text{Spec:Cheb}}_{N=149} + UA^{\text{Spec:Cheb}}_{N=150}\right)/2\right|}{\left(UA^{\text{Spec:Cheb}}_{N=149} + UA^{\text{Spec:Cheb}}_{N=150}\right)/2} 100\% \tag{22}$$

Note that, in the convergence studies with larger grids presented in Fig. 4, errors were defined similarly with 1049/1050 and 5049/5050 grid points (*N*) for pinch and *UA*, respectively. A similar approach has also been used earlier to study the accuracy of the LMTD method (Chen, 2016).

Pinch and *UA* errors typically decrease as *N* increases, but not necessarily monotonic since these approximation errors also have random aspects, e.g., if one grid point happens to be very close to the pinch point. Hence, to identify general trends it is more interesting to study maximum ΔT_{pinch} and *UA* errors obtained for schemes with *N* grid points or more. Because of this, identifying a specific value for the required accuracy of the numerical approximation method is not appropriate in this work and the approach, as set out above, uses relative error as the basis.

2.4. Run time

The run time, i.e., the time it takes a computer to calculate the heat exchanger property of interest, depends on the method used. This is not a perfect measurement of the performance, since run time also depends on programs running in the background and the hardware of the computer. However, the relative run time between different schemes is less computer dependent. To be able to study and compare the run time for different schemes in more detail, the run time plots in Figs. A.1 and A.2 were smoothed using the MATLAB function polyfit to create 5th-order polynomials with respect to the number of grid points *N*. The mean run time does not include calculations that failed, which only occurred for fluid mixtures. The results in this study were obtained using a computer with Intel(R) Xeon(R) W-2123 CPU @ 3.60GHz processor and 32 GB RAM.

2.5. Model validation

A convergence test is used to validate the different HE schemes; the idea is that standard low-order method converges to the same values as the novel high-order HE methods presented in this article. The results from the convergence study are presented in Section 3.1, where model validation is discussed based on the convergence behavior and quantified precision levels.

3. Results and discussion

It is often important that numerical *UA* and ΔT_{pinch} errors are small, e.g., such errors will appear as random noise in optimization studies which can cause optimization algorithms to fail or be inefficient. Since an analytical solution to compare the results does not exist, this article explores error estimates defined in Eqs. (21) and (22). Hence, the error of interest in this article is the errors generated when computing the *UA* and ΔT_{pinch} numerically, and not the actual error which is also affected by the accuracy of the EOS used by the fluid property package.

It is well known that high-order methods need less grid point-calculations to model smooth functions, but also that they can be unreliable if noise is added to the function, or the function is not smooth. The functions being modeled here are based on the temperature approach in the HE (ΔT), and the smoothness of ΔT in each HE section, separated by the bubble and dew points, depends on noise and bubble/dew points errors introduced by the fluid property packages when solving the EOS. This article focuses on investigating the accuracy obtained in heat exchangers with pure and mixed fluids. Since the mixed fluids are more challenging for fluid property packages to solve, the methods developed will be tested for different situations that can create different challenges for the high-order methods. The six different cases described in Table 1 are modeled with the different *UA* and ΔT_{pinch} heat exchanger schemes. Error estimates and computer run times are found for schemes with different number of grid points, and modeling accuracy

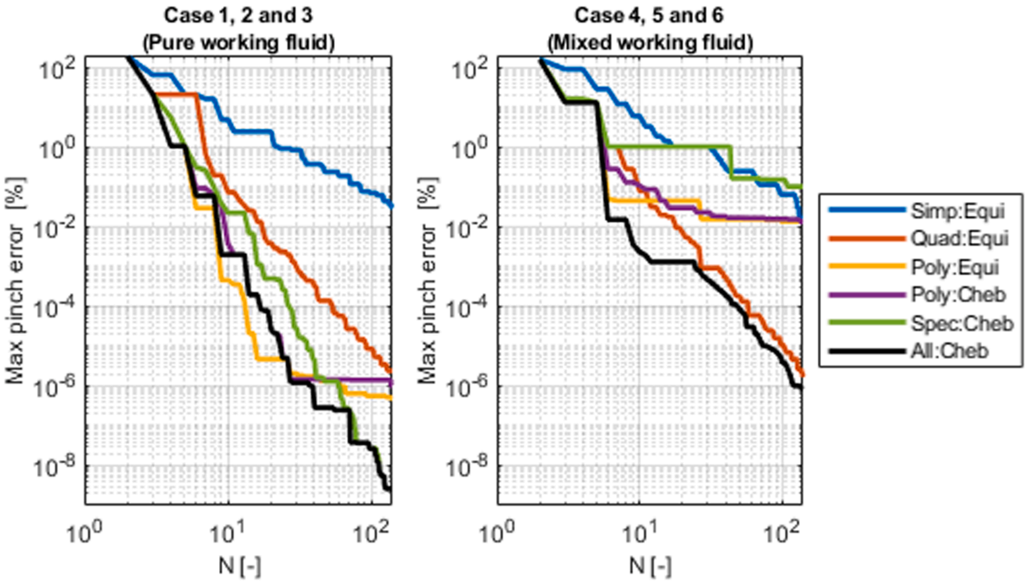

Fig. 5. Maximum temperature pinch errors found in schemes having at least N grid points. Pure working fluid left (Case 1 – 3), and mixed working fluid right (Case 4 – 6).

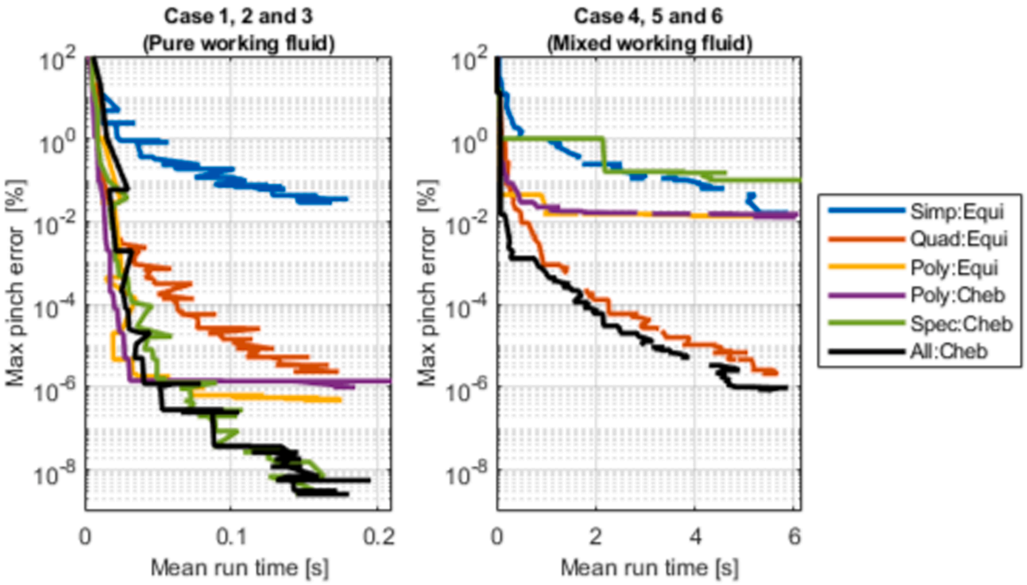

Fig. 6. Maximum temperature pinch errors versus mean run time.

versus run time is investigated by studying cases with pure and mixed working fluids separately.

3.1. Model validation and convergence

Fig. 4 shows Case 1 modeling errors for the different methods developed and includes schemes with large grids (N) to compare and to verify convergence between different low and high-order methods. Fig. 4 illustrates that the different methods have been implemented correctly, since the low-order methods converges to the same values as the high-order methods for large grids. For example, the 'Spec:Cheb' method converges quickly for the $\Delta T_{\mathrm{pinch}}$ calculations, and the second-order method 'Quad:Equi' converge to the same value (pinch error < 1.0E-9 %) if 1000 grid points are used. Fig. 4 also shows that the UA error converges quickly for the high-order method 'Spec:Cheb'. For 5000 grid point schemes, 'Spec:Cheb' converge to UA error < 1.0E-11 %, while the low-order methods converge toward the same value, i.e. 'LMTD:Equi' has UA error < 1.0E-6 % and 'Trap:Equi' has UA error < 1.0E-7 %. That is, the novel approaches presented in this study are validated based on the almost identical precision levels obtained for both the conventional and unconventional methods in the large convergence studies.

3.2. Performance of the ΔT_{pinch} models

Cases with pure and mixed working fluids are studied separately since such schemes operate with different accuracies. The pinch error obtained with a given HE model typically decreases if more grid points are used in the modelling work (see Figs. 4 and A.1). The 'maximum pinch error' for a scheme with grid size N is defined as the largest pinch error found in all schemes having at least N grid points, i.e., all schemes with grid sizes in the interval $[N,100]$. The maximum pinch error is used

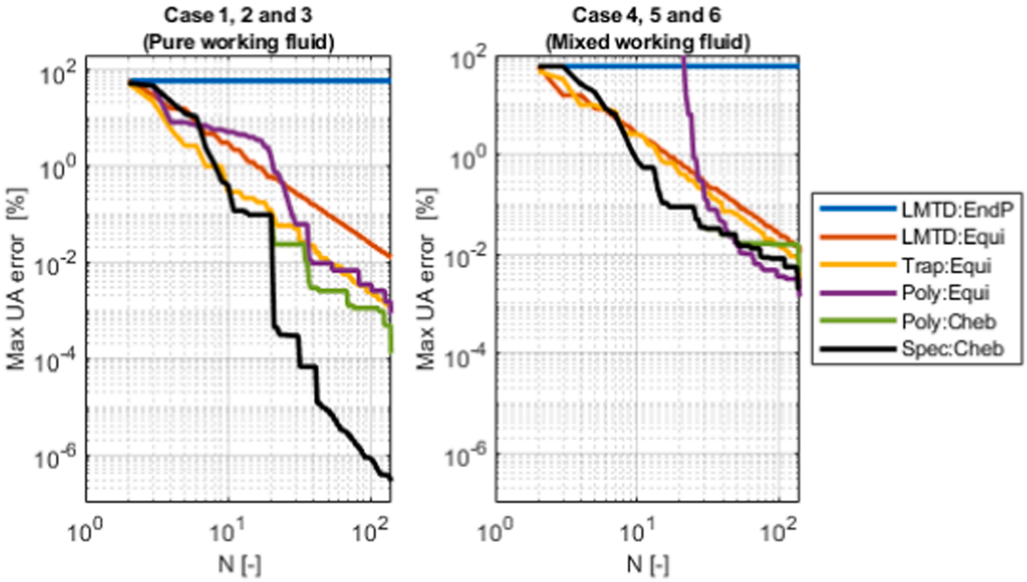

Fig. 7. Maximum UA errors in schemes having at least N grid points. Pure working fluid left (Case 1 – 3), and mixed working fluid right (Case 4 – 6).

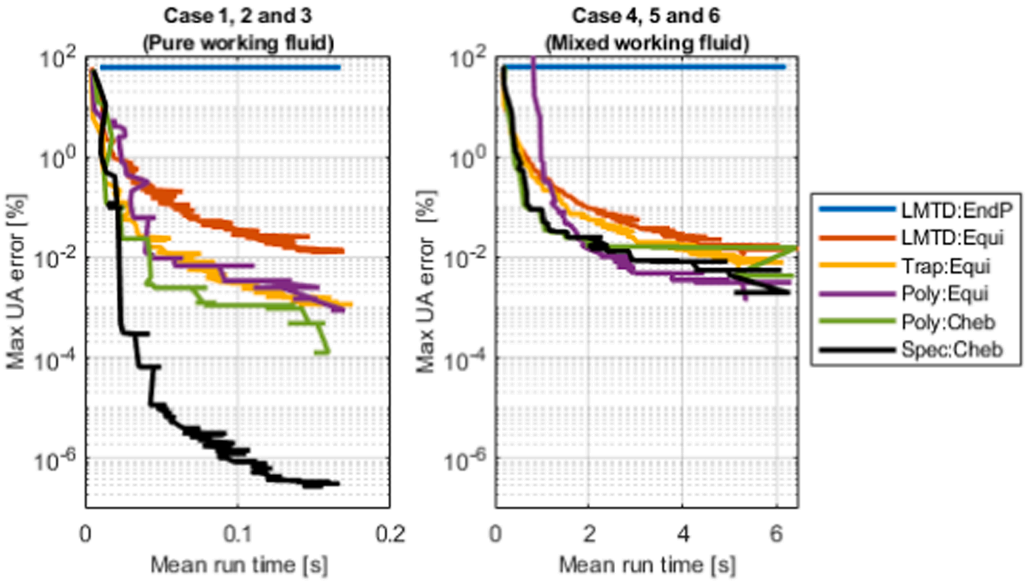

Fig. 8. Maximum UA errors versus the mean run time.

to study the accuracy more generally and decreases monotonically as the grid size increases. Figs. 5 and 6 present results illustrating how the maximum pinch error varies with grid size and run time, and are calculated from data presented in Fig. A.1 in the appendix, which shows heat exchanger modeling error and run time for each of the six cases for different grids. Figs. 5 and 6 show that the method 'Simp:Equi', which is probably the most commonly used method in the literature (Kemp, 2007; Dimian et al., 2014), is inefficient. This method needs large grids to obtain accurate results, and only introduces pinch errors less than 0.1% for $N \geq 90$. $N = 140$ is not even sufficient to obtain a 0.01% accuracy. Fig. 5 shows that high-order methods are successful when modeling HEs with pure working fluids, and Fig. A.1 shows that high-order methods obtain similar errors 10 times faster than 'Simp: Equi' in all cases except Case 6, where the dew-point and bubble-point for mixtures could not be calculated by the fluid property package. Fig. 6 shows that the ΔT_{pinch} results obtained by combining high and low-order methods ('All:Cheb') is among the best, and a small grid $N \geq 10$ is sufficient for obtaining a 0.01 % accuracy for all the cases studied, i.e., including the difficult Case 6. The 'All:Cheb' method requires more calculations and is therefore more time-consuming than the other methods with similar N, however, the extra calculations often improves the accuracy.

3.2.1. *Pure fluids*

Fig. 5 shows that the hybrid 'All:Cheb' scheme, which include both high and low-order methods, typically is the most accurate with respect to a given N, with pinch error less than 1.0E-7 % for $N \geq 100$, and requires less than 10 grid points to obtain a 0.01 % accuracy. For pure working fluids, Fig. 6 clearly shows that high-order based methods are the best approach, with respect to runtime, if high accuracy is needed. All the high-order methods can be used to obtain accuracy around 1.0E-6 %. The 'Spec: Cheb' and 'Spec: All' both managed to obtain errors less than 1.0E-8 %. Fig. 6 shows that a second-order approach on an equidistant grid ('Quad:Equi') becomes inefficient for obtaining an accuracy

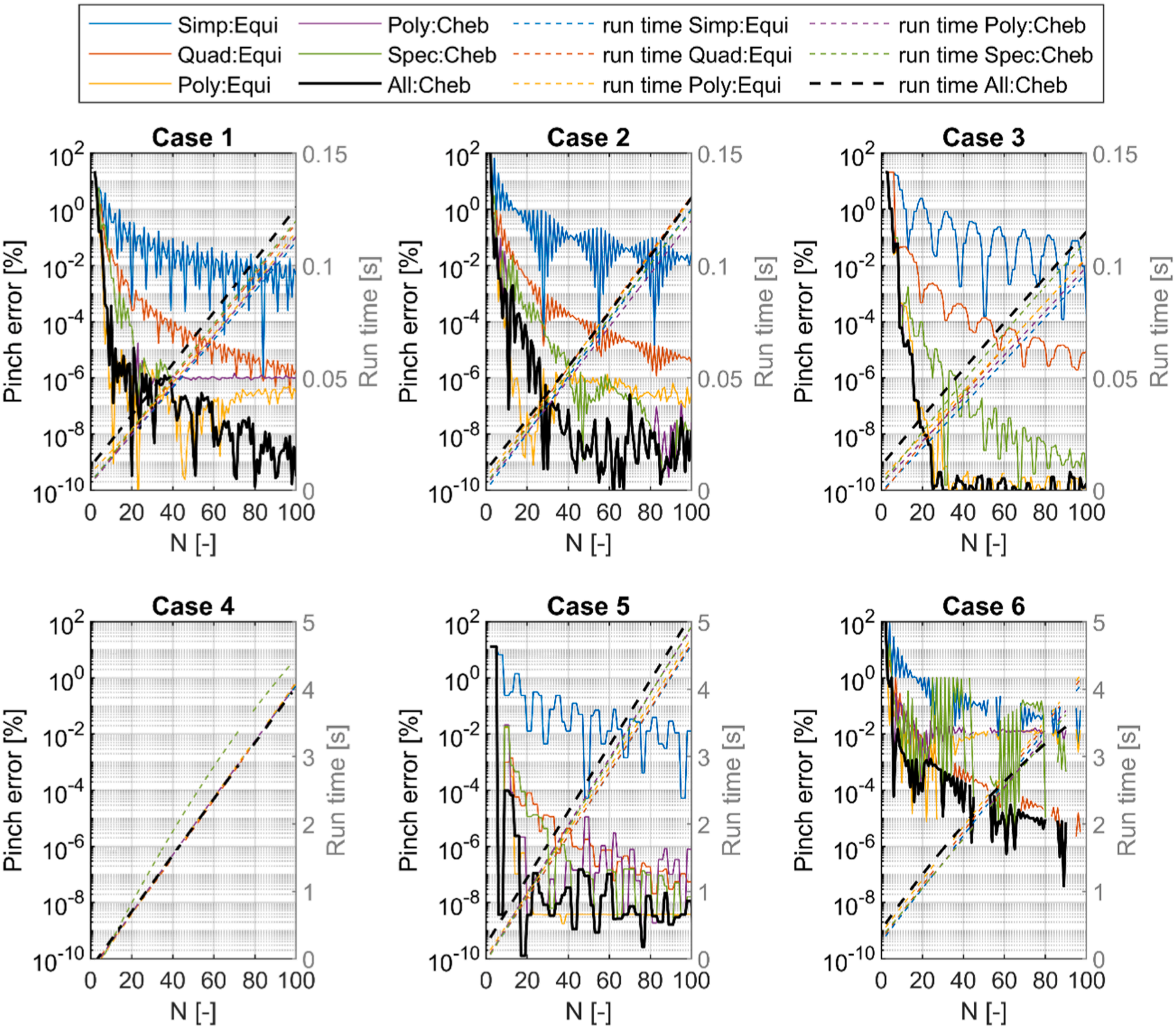

Fig. A.1. Temperature pinch errors and run times for Case 1 – 6, using six different methods. Note that, the temperature pinch in Case 4 is located at the dew point, as illustrated in Fig. 2. Hence, all the internal grid point configurations and interpolation methods obtain the same result, and the ΔT_{pinch} error is therefore exactly zero for all the schemes modeling Case 4.

better than 0.01 %, since a much larger grid is needed, as illustrated in Fig. 5.

3.2.2. Mixed fluids

Figs. 5 and 6 show that the 'All:Cheb' method is the most accurate mixture-scheme for a given *N*, but also with respect to run time, and benefits from using two 2$^{\text{nd}}$-order methods ('Quad') in a sequence when the high-order methods fail. Fig. 6 also shows that the 'Quad:Equi' method is an alternative, since high-order methods become less accurate in Case 6, as illustrated by in Fig. A.1. The discontinuities visible in Fig. 6 indicate that the corresponding *N*-scheme failed, and that the fluid package was unable to calculate all the mixed fluid temperatures. As illustrated in Figs. 6 and A.1 this occurred more often when using large grids, i.e., in schemes with many calls to the fluid package.

3.3. Performance of the UA-value models

As mentioned earlier, general numerical adaptive quadrature based integration methods to compute *UA* by defining a tolerance, instead of a grid size *N* (Shampine, 2008), have been applied in earlier studies (Chen, 2016). However, temperature profiles in HEs with different fluids have much in common, as illustrated by Fig. 2, and it should therefore be possible to find a fixed (non-adaptive) optimal numerical integration scheme, with respect to accuracy and run time, that can be applied to all HEs. This is also confirmed by the results discussed below.

The *UA* errors shown in Figs. 4 and A.2 typically decrease with the number of grid points *N*. The 'maximum *UA* error' for a scheme with grid size *N* is defined as the largest *UA* error obtained in the modelling work with *N* grid points or more, i.e., all schemes with grid size in the interval [*N*, 100]. How the maximum *UA* errors depend on the grid size and run time are illustrated in Figs. 7 and 8, respectively. These figures are based on data illustrated in Fig. A.2, which shows heat exchanger modeling errors and run times. The LMTD method is based on an analytical integration approach to find the *UA*-value and is probably the most common approach used by engineers today. However, Fig. 8 shows that such methods are unreliable and slow compared to numerical integration methods. For example, the *UA* errors with the 'LMTD:EndP' method is between 30 – 60 % for Case 1 and 2 where CO_2 is cooled in a gas cooler. The poor quality of similar 'LMTD:EndP' methods has also been studied (Chen, 2016), however, the LMTD endpoint approach is still used by many. Others, such as (Watson et al., 2015; Elias et al., 2019; Vikse et al., 2020), use a more accurate method by calculating the

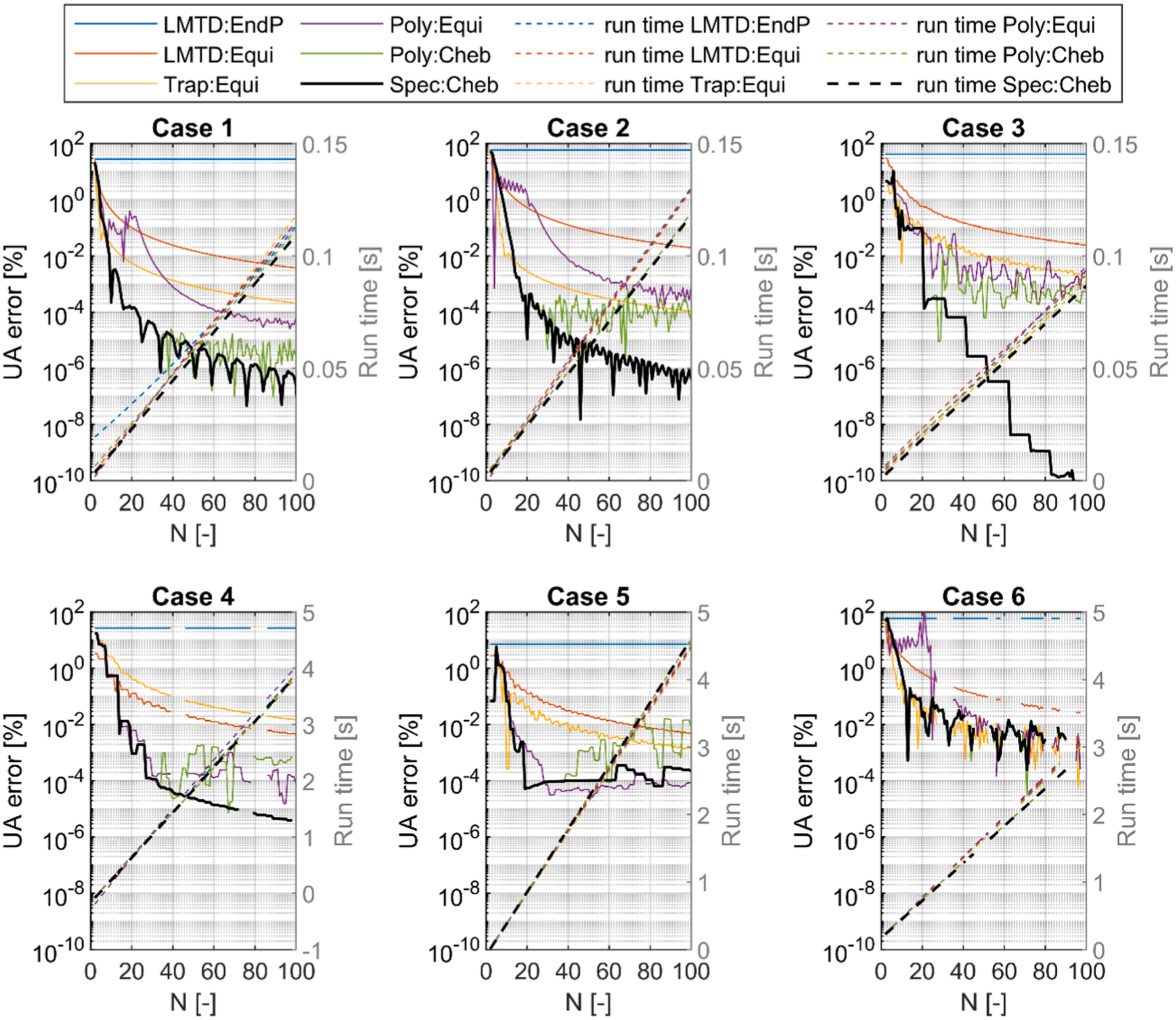

Fig. A.2. UA errors and run times for Case 1 – 6, using six different methods.

LMTD between all neighboring grid points, here named ‘LMTD:Equi’, but also this method can be inaccurate as illustrated by Fig. 8, i.e., generating *UA* errors that are typically 10 times greater than a simple numerical Trapezoidal integration approach (‘Trap:Equi’) for a given *N*. The high-order ‘Spec:Cheb’ approach obtains solution with *UA* errors less than 0.1 % around 3 times faster than the perhaps most commonly used method ‘LMTD:Equi’. A better accuracy than 0.1 % is almost impossible to obtain with ‘LMTD:Equi’.

3.3.1. *Pure fluids*

The ‘Spec:Cheb’ approach stands out as the fastest method to obtain an accurate *UA*-value for one component fluids, which is clearly illustrated in Fig. 8, where *UA* errors less than 0.01 % are obtained for $N > 20$. For $N \geq 100$, the error is less than 1.0E-6 %. The high-order ’Spec: Cheb’ approach obtains solution with *UA* errors less than 0.01 % about 8 times faster than ’LMTD:Equi’, and a better accuracy than this is almost impossible to obtain with “LMTD:Equi”. The other high-order methods (’Poly:Cheb’ and ’Poly:Equi’) are less reliable, and struggle to obtain *UA* errors less than 0.01 % for Case 3.

3.3.2. *Mixed fluids*

As for the pure working fluid cases, Case 4 and 5 with mixtures obtain *UA* errors less than 0.01 % for $N > 20$ with the ‘Spec:Cheb’ method (see Fig. A.2 in the appendix). However, the *UA* error is not reduced below 5.0E-4 % if *N* is increased beyond 30. Case 6 is more difficult to solve accurately because the fluid package is not able to find the bubble and dew points (see Fig. 2). Because of this the calculations do not obtain *UA* errors less than 0.1 % for $N \approx 20$, and the error is about 0.01 % if $N \geq 80$. For Case 6, which is a special case very close to the critical point, high-order methods such as ‘Spec:Cheb’ obtain about the same accuracy as the low-order ‘Trap:Equi’. In the end, ‘Spec:Cheb’ is recommended since this method is much better than ‘Trap:Equi’ in the two other cases (Case 4 and 5) where dew and bubble points were identified correctly. Fig. 3 also shows that Case 6 is relatively special, since the phase envelope calculations have only failed in a few points that lie near to the critical point.

3.4. *Run time*

As mentioned in Section 2.4, the computer run time is not a perfect measurement of the performance of different heat exchanger models. However, the results presented illustrate larger run time trends which are interesting and that seems logical. For example, Fig. 4 shows that the run time has 25 % or less numerical fluctuations, relative to a linear approximation, as *N* increases. The grid points generated are different when *N* is changed, and some fluid states require less work to be solved by the property packages. Figs. A.1 and A.2 also illustrate that the run time (approximated as 5th-order polynomials) is almost a linear function with the number of grid nodes N, and that the run time is highly influenced by the fluid property packages applied in the modeling work. The most time-consuming calculations in the heat exchanger modeling are related to the fluid property package calculations, especially when calculating fluid properties of mixtures. Figs. 5 and 6 illustrate that the heat exchanger schemes with mixtures, Case 4 – 6, are about 30 times slower than the pure fluid schemes (Case 1 – 3). The run time for HEs with fluid mixtures is therefore totally dominated by the number of calls to the fluid package.

Note that the run time includes the calculation of the spectral differentiation matrix $\mathbf{D}$ and the integral weights w, but these computations only have to be done once in optimization studies where the heat changer models are used multiple times. In optimization studies, it is therefore possible to reduce the run time of the high-order spectral schemes ("Spec: Cheb" and "All:Cheb") relative to the low order schemes ('Simp:Equi' and 'Quad:Equi'). For mixed fluids the number of calls to the fluid package dominates the run time, and the time it takes to calculate $\mathbf{D}$ and w is therefore less important.

3.5. Modeling issues related to fluid property packages

The accuracy of the heat exchanger models depends on the fluid property package, which solves EOSs numerically using algorithms that can introduce both errors and inaccuracies that affect high and low-order schemes differently. High-order schemes are superior if the temperature profile in the HE sections, separated by the bubble and dew points, is smooth. However, such schemes can be unreliable if the property package cannot find dew or bubble points, or calculate internal temperatures, or accurately solve the EOS numerically. Low-order methods, on the other hand, are more reliable, but require a much larger grid to obtain a given accuracy if ΔT is smooth. If high-order schemes fail, moderately high-order HE methods should be considered in UA modeling (Shampine, 2008), and hybrid low and high-order methods should be used for ΔT_{pinch} modeling (see Fig. 8). That is, future studies could investigate how N grid points can be optimally distributed between interpolation order ($N_{\text{grid},i} - 1$) and number of HE sections (e.g., $i \approx N/N_{\text{grid},i}$) for different fluid property packages, using the Chebyshev grid nodes x_i and weights w_i as explained in Eq. (16).

Three different aspects of fluid property packages are identified in the modeling work to be important for the UA and ΔT_{pinch} schemes:

1 Errors finding dew and bubble points.
2 Accuracy of the T_{source} and T_{sink} calculations.
3 Errors finding T_{source} or T_{sink} node values.

For pure fluids, Fig. 4 shows that UA and ΔT_{pinch} errors converge to less than 1E-10. Figs. 5 and 7 show that errors are less than 1E-6 for 'Spec:Cheb' schemes with $N \geq 100$. That is, EOS is solved numerically with high precision for pure fluids, and the correct dew and bubble points are always found. Even though the fluid property packages return properties with very high precision for pure fluids, it is still sometimes too much numerical noise for the high-order methods 'Poly:Cheb' and 'Poly:Equi'. In particular for Case 3, where results with UA errors less than 0.01 % were difficult to obtain. However, none of the pure fluid cases lost the high-order accuracy in the 'Spec:Cheb' schemes, and the main problems in the UA and ΔT_{pinch} calculations appeared when modeling mixtures.

The fluid mixture schemes (Case 4, 5 and 6) are very interesting when discussing HE methods, since they introduce the fluid property package problems relevant for the HE models, as summarized below:

1 Loss of high-order accuracy occurred when dew and bubble points are not found. High-order interpolation properties are lost when the HE is not separated into section where the temperature profile is smooth. Case 6 is the only case studied where the fluid package is not able to find bubble and dew points, and is also the only case studied where the best high-order methods ('Spec:Cheb') sometimes were less accurate than the low-order methods for large grids. That is, 'Spec:Cheb' failed to obtain a result with UA and ΔT_{pinch} errors less than 0.01 % and 0.1 %, respectively, for $N \geq 100$. Fig. 3 illustrates that the phase-envelope of the mixtures is not always calculated correctly by the fluid property package, and that this typically occurs near the critical point. There are also more errors in the mixture schemes with a high concentration of CO_2 (Case 6), which perhaps explains why HENs with this concentration were found to be more difficult to optimize (Brodal and Eiksund 2020).
2 Loss of high-order accuracy due to inaccuracies in the T_{source} and T_{sink} calculations using the fluid property package. Fig. A.2 shows that it is difficult to obtain UA errors less than 3.0E-4 % when modeling the other mixed fluid cases (Case 4 and 5). The ΔT_{pinch} errors are less affected and obtain a similar accuracy as for pure fluids. However, the errors might have been similar if the interpolated values $\Delta T_{i,\text{ex}} = \text{polyval}(q_i, Q_{i,\text{ex}})$ had been used directly, instead of calculating temperature and enthalpies from $Q_{i,\text{ex}}$ using additional calls to the fluid property package. Note that, the noise introduced in the T_{source} and T_{sink} calculations affects the high order 'Spec:Cheb' results much less than the dew and bubble point errors in Case 6, where UA and ΔT_{pinch} errors typically are at least 10 times greater.
3 Fluid property package errors in finding T_{source} or T_{sink} node values. Unlike pure fluids, the temperatures on the internal grid points sometimes failed with mixtures. This is a bigger problem in schemes with many grid points, since it is statistically more likely to occur with increased calls to the fluid property package. This is also exactly what Figs. A.1 and A.2 show, where the discontinuities in Case 4 and Case 6 indicate that the fluid property package failed to evaluate the temperature profile in one of the grid points.

Note that only two fluid property packages were tested here, and other programs can solve different EOS or use different numerical algorithms to evaluate fluid properties. However, the cases studied here include best case fluid property package scenarios (pure fluid), and more difficult scenarios (mixtures) where the three different problems described above create problems for the UA and ΔT_{pinch} models.

4. Conclusions

UA and ΔT_{pinch} methods have gained much attention in HEN modeling due to their simplicity, which is an important advantage when discussing and comparing systems on a general basis. Obtaining these values without adding too much numerical noise is particularly important in multivariable optimization problems, however, this computational aspect has typically been neglected in existing literature, where the mainstream approaches have been applying very simple, and often inaccurate, HE models. This article explores and compares both low and high-order methods, as well as a hybrid method.

The results in Figs. 6 and 8 show that the mainstream methods ('Simp:Equi' and 'LMTD:Equi') are 2–5 times slower than high-order based methods if errors less than 1 % are required. Optimization studies can require even more accurate schemes to be successful, however, finding the required HE accuracy for different optimization algorithms is outside the scope of this article. The ΔT_{pinch} calculations show that the hybrid 'All:Cheb' method, which combines high and low-order methods, is relatively accurate, fast and reliable for all cases investigated. Based on Figs. 5 and 6, it is therefore recommended that the ΔT_{pinch} is modeled with the hybrid 'All:Cheb' method using at least 10 grid points if a 0.01 % accuracy is required. However, for pure fluids, high-order methods obtain similar performance as the hybrid method. The simplest method 'Simp:Equi' needs more than 140 grid points to reach 0.01 % accuracy and operates with a computer run time which is 10 – 20 times slower than for 'All:Cheb' (see Fig. 6). For *UA*-value calculations, the high-order method 'Spec: Cheb' method is recommended (see Fig. 8). The worst-case scenario, Case 6 where the dew and bubble points calculation failed, the high-order methods perform with similar accuracy as the low-order methods. In all the other cases also including working fluid mixtures, the "Spec: Cheb" method were outperforming the low-order methods (see Fig. A.2). As Fig. 7 shows, 20 grid points, or more, are needed for pure fluid systems to obtain results with an *UA* error less than 0.01 %. For fluid mixtures, a 20 grid points scheme only generates results with errors less than 0.1 %. However, increasing the number of grid points to 70, the accuracy for working fluid mixtures improves to 0.01 % with 'Spec: Cheb' schemes. The 'LMTD:Equi' method, where LMTD values are calculated between each neighboring nodes, but this approximation is not doing very well in the comparison tests. Compared with 'Spec: Cheb', for example, 'LMTD:Equi' increases the run time 2 – 5 times if results with 0.01 % accuracy are required, and the 'LMTD:Equi' scheme is also extremely inefficient for obtaining results with higher precision than 0.01 %, even for schemes with pure fluids.

For pure fluids, *UA* and ΔT_{pinch} errors obtained with high-order (spectral) interpolation methods converge quickly to zero with increased number of grid points, e.g., HE models with more than 30 grid points generated results with an accuracy better than 1.0E-4 %. For systems with mixtures, the smoothness required by high-order methods is sometimes lost due to inaccuracies and errors in the fluid property software. In particular, if the fluid property package fails to find dew and bubble points, as in Case 6. In general, high-order methods converge quickly if the temperature profile in the HE sections, separated by the bubble and dew points, is smooth, but can also be unreliable if something is slightly wrong, e.g., if the property package is unable to find dew point, bubble point or calculate internal temperatures accurately. Low-order methods, on the other hand, are more reliable, but require a much larger grid to obtain a given accuracy if the EOS can be solved with high presession as for pure fluid. Finding the necessary level of accuracy needed in optimization studies of HENs, i.e., an optimal number of HE grid points, with respect to optimization run time, should be further investigated for the numerical heat exchanger models that have been recommended in this study.

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. **Getu Hailu:** 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.

Appendix

Fig. A1, Fig. A2

References

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 Internationale 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 CO2 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.

Chen, Y.-G., 2016. Pinch point analysis and design considerations of CO2 gas cooler for heat pump water heaters. Int. J. Refrig. 69, 136–146.

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., C. S. Bildea and A. A. Kiss (2014). Chapter 13 - Pinch Point Analysis. Computer Aided Chemical Engineering. A. C. Dimian, C. S. Bildea and A. A. Kiss, Elsevier. 35: 525-564.

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.-Revue Internationale Du Froid 65, 12–26.

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.

Forooghi, P., Hooman, K., 2014. Experimental analysis of heat transfer of supercritical fluids in plate heat exchangers. Int. J. Heat Mass Transfer 74, 448–459.

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. In: Kemp, I.C. (Ed.), Pinch Analysis and Process Integration, Second Edition. Oxford, Butterworth-Heinemann. xiv-xv.

Kunz, O., Wagner, W., 2012. The GERG-2008 Wide-Range Equation of State for Natural Gases and Other Mixtures: An Expansion of GERG-2004. J. Chem. Eng. Data 57 (11), 3032–3091.

Rao, R.V., Saroj, A., Oclon, P., Taler, J., 2020. Design Optimization of Heat Exchangers with Advanced Optimization Techniques: A Review. Arch. Comput. Meth. Eng. 27 (2), 517–548.

Sarkar, J., Bhattacharyya, S., 2009. Assessment of blends of CO2 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.

Span, R., Eckermann, T., Herrig, S., Hielscher, S., Jäger, A., Thol, M., 2016. TREND. Thermodynamic Reference and Engineering Data 3.0. Lehrstuhl für Thermodynamik, Ruhr-Universität Bochum.

Span, R., Wagner, W., 1996. A New Equation of State for Carbon Dioxide Covering the Fluid Region from the Triple-Point Temperature to 1100 K at Pressures up to 800 MPa. J. Phys. Chem. Ref. Data 25 (6), 1509–1596.

Trefethen, L.N., 2000. Spectral methods in MATLAB. SIAM, Philadelphia, Pa.

Vikse, M., Watson, H.A.J., Kim, D., Barton, P.I., Gundersen, T., 2020. Optimization of a dual mixed refrigerant process using a nonsmooth approach. Energy 196, 116999.

Watson, H.A.J., Khan, K.A., Barton, P.I., 2015. Multistream heat exchanger modeling and design. AlChE J. 61 (10), 3390–3403.