Synthesis of feasible heat exchanger networks using attainable regions

Authors:

Avian Yuen

Abstract:

The attainable region (AR) is a region in a performance space in which all physically realizable reactor network designs must exist. ARs have been used since the 1960s for solving reactor network synthesis problems. The benefits of these methods are that the feasibility of a performance target can be assessed prior to running a synthesis routine, the solutions they give are guaranteed to be physically realizable, and a design can be made robust to uncertainties in feed and performance targets by assessing whether a solution and the range of its possible values lie within the AR, just to name a few. In this article, the theory of attainable regions is extended to bring these benefits to the heat exchanger network (HEN) synthesis problem. Basic properties of the HEN-AR are proven and a synthesis method using the AR is presented with examples.

Synthesis of feasible heat exchanger networks using attainable regions

Avian Yuen

Department of Chemical Engineering. McMaster University, 1280 Main Street West, Hamilton, Ontario, L8S 4L7, Canada

Abstract

The attainable region (AR) is a region in a performance space in which all physically realizable reactor network designs must exist. ARs have been used since the 1960s for solving reactor network synthesis problems. The benefits of these methods are that the feasibility of a performance target can be assessed prior to running a synthesis routine, the solutions they give are guaranteed to be physically realizable, and a design can be made robust to uncertainties in feed and performance targets by assessing whether a solution and the range of its possible values lie within the AR, just to name a few. In this article, the theory of attainable regions is extended to bring these benefits to the heat exchanger network (HEN) synthesis problem. Basic properties of the HEN-AR are proven and a synthesis method using the AR is presented with examples.

Keywords: Heat exchanger network synthesis, Attainable region, Process synthesis, Heat integration, Energy recovery

Highlights:

1. The benefits of attainable region based reactor network synthesis, such as being able to assess the feasibility of the problem prior to synthesis, the guarantee of physical realizability, and the ease in which uncertain parameter values are taken into account, is introduced to the heat exchanger network synthesis problem.

2. Treating enthalpy exchange as a kind of chemical reaction allows the notion of an attainable region to arise for heat exchanger networks.

3. The heat exchanger network attainable region is described and its properties are proven.

4. A predictably terminating method is presented which uses these ARs to generate feasible heat exchanger networks within the class of design admitting constant thermophysical properties and no stream splits.

Email address: `yuenah@mcmaster.ca` (Avian Yuen)

1. Introduction and Motivation

The economic and environmental value in reducing external energy requirements in industrial processes needs no defence. One way to do this is through heat integration—using complementary heating and cooling duties in a given process to satisfy both duty requirements via heat exchanger networks (HEN). Emblematic of how clear the value of heat integration is, the famous "pinch analysis" developed by Bonn Linhoff in 1983—one of the earliest techniques developed for heat integration—has become highly cited (Linnhoff and Hindmarsh, 1983).

Since then, with the advent of personal and high-performance computing, many computer-aided algorithms for heat exchanger network synthesis, including those for pinch analysis, have been developed. Mathematical programming approaches like those of the highly-cited mixed-integer nonlinear programming (MINLP) model of Yee and Grossmann are particularly dominant (Yee and Grossmann, 1990). In fact, among the methods covered in an exhaustive re-

view by Furman and Sahinidis, 81 of the 461 articles published on the subject in the last one hundred years take the form of a MILP or MINLP (Furman and Sahinidis, 2002). Pinch technology and MILP or MINLP models are the top two most common solution methods for HENS.

However, among all methods, none have yet been developed using attainable region theory (ART). ART, originally conceived by F.J.M Horn for reactor network synthesis (RNS), has some advantages over other synthesis methods (Horn, 1964). In particular, the knowledge that it provides about the limits of performance among all possible reactor types, guarantee of physical realizability, and ability to handle uncertainties, as well as its amenability to convex optimization, make it stand out (a good discussion of AR compared to other methods can be found in the book on the subject by Ming and colleagues (Ming et al., 2016)). These advantages prompted a flurry of activity at the end of the 1990s, developing ART to the point where it has formed its own class of RNS methods (Glasser et al., 1987; Feinberg and Hildebrandt, 1997; Feinberg, 2000b,a; Tian et al., 2018). ART is slowly making its way into other areas of process engineering. To date, it has been used to tackle problems in solids processing (comminution), economic optimization, and gasification (Khumalo et al., 2006; Muvhiiwa et al., 2018; Bedenik et al., 2007). Hybrid AR-MINLP methods have also been developed to marry some of the benefits of each together (Pahor et al., 2000). Part of the appeal of attainable region analysis is that the geometric interpretation of the design space provides additional information for where an optimal design may lie (Martín and Adams, 2019).

Although attainable region theories have dealt with reactor networks with heat exchangers in the past (see (Nicol et al., 1997) and (Nicol et al., 2001)), none have been developed solely for the HENS problem. Given the strengths of having an AR to inform the synthesis, it would be beneficial if such a theory of ARs could be articulated for heat exchanger network synthesis as well. This article seeks to do just that. Developing a theory of the "HEN-AR" may bring some of the benefits which come with attainable region theory to this significant topic.

Our discussion will proceed first with a review of the "classic" AR theory which applies to reactor networks. It will then be shown that through the definition of appropriate *heat exchange reactions* a similar AR can be generated for HENs. Analogous to reactor network attainable regions (RN-ARs), this HEN-AR is a region in stream-temperature-space wherein the performance of all feasible HENs must lie. Then, much like in RN-ART, the HENs which correspond to these performance states can be derived by identifying an appropriate path through the region. Thus, the theory introduced here will allow for the generation of the complete performance space of heat exchanger networks, as well as a method to synthesize such HENs.

In order to more simply introduce the techniques involved, only a subclass of the HENS problem will be considered here where all thermophysical properties of materials can be presumed constant. Specifically, heat capacity, heat exchange coefficients, and material phases are kept constant. For this subclass of problems, properties of the AR will be deduced and those will be used to devise a method to generate the AR. For the method that will be shown, it turns out that the HEN synthesis using the AR happens in a serial fashion, with non-negative progress towards a feasible solution being made at each step. This contrasts methods which treat the HENS problem as an MILP, such as the transshipment model of Papoulias and Grossmann, wherein some choices of stream matches may need to be "undone" by an algorithm—going back up the binary variable branch-and-bound tree—to reach a feasible solution (Papoulias and Grossmann, 1983). The determination of whether a particular HEN lies within the AR may be aided by some properties of the AR, which will be described and proven.

In sum, this article seeks to accomplish two goals. One,

to introduce an AR-based algorithm which gives robust solutions for HENs in which the constant thermophysical properties assumption is acceptable, and two, to provide proofs of the basic properties of the HEN-AR which can aid explorations in extending the AR techniques to a wider class of HENS problems in the future.

2. Background

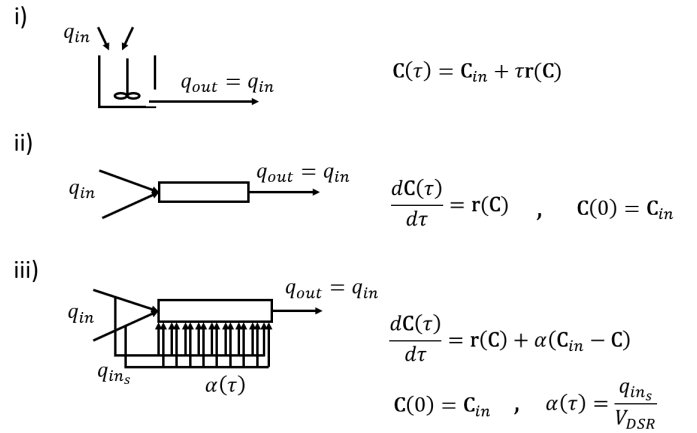The discussion will begin first with a review of RN-ART, followed by heat exchanger models.

2.1. AR theory for RNS

A reactor network (RN) is the class of all processes which involves material streams, stream splitters, stream mixers, and at least one reactor. The attainable region is a region in a performance state-space (typically the concentration space of all involved species) such that it contains the outputs of all RNs for the given RNS problem.

The AR is such that all contained states are associated with at least one RN design. The synthesis of a reactor network is therefore no more complicated than the selection of a particular state in the AR. The task is made more simple if the chosen performance state is on the boundary of the AR since, it turns out, boundary performance states are associated with only one RN design. Thus, the maximization of the performance of RNs—seeking RNs on the boundary of the AR—helpfully coincides with the generation of the unique design with that performance state.

The question now becomes how to generate the RN-AR. Most simply, the AR boundary is composed of mixing tank trajectories and reactor trajectories of the so-called fundamental reactor types: continuous-flow stirred tank reactors (CSTRs), plug-flow reactors (PFRs), and differentially-distributed side-stream reactors (DSRs). These are called the "fundamental reactors" because it has been proven that the output of any reactor can be arbitrarily approximated by an RN containing just these three reactor types (Ming et al., 2016). The common process flow

diagram symbol for each reactor, along with its associated model equation, are given in Figure 1.

i)

$$\mathbf{C}(\tau) = \mathbf{C}_{in} + \tau \mathbf{r}(\mathbf{C})$$

ii)

$$\frac{d\mathbf{C}(\tau)}{d\tau} = \mathbf{r}(\mathbf{C}) \quad , \quad \mathbf{C}(0) = \mathbf{C}_{in}$$

iii)

$$\frac{d\mathbf{C}(\tau)}{d\tau} = \mathbf{r}(\mathbf{C}) + \alpha(\mathbf{C}_{in} - \mathbf{C})$$

$$\mathbf{C}(0) = \mathbf{C}_{in} \quad , \quad \alpha(\tau) = \frac{q_{in_s}}{V_{DSR}}$$

Figure 1: The three fundamental reactors (right) and their equations (left). There are three: i) the CSTR, ii) the PFR, iii) the DSR.

Reactor trajectories are the performance outputs of a reactor given inlet feed information, over increasing residence time τ from zero to the limit of infinite time units (e.g. seconds). For an AR in concentration space, the performance is the output concentration $\mathbf{C}(\tau)$ of chemical species from the reactors. AR theorems have proven that the AR is finite and simply connected (Ming et al., 2016). Thus, the most simple generation of the AR involves starting with an empty AR, adding the feed point, and adding all subsequent states traversed by the trajectories of CSTRs, PFRs, and DSRs from the feedpoint and from their own outputs. Because the region is simply connected, the result is guaranteed to be the complete AR. The RNs corresponding to the AR are derived by determining which type of reactor trajectory a chosen state belongs to.

With these basics of attainable region theory in hand, let us examine the generation of an AR for a simple RNS problem.

2.1.1. RN-AR Example

This example, presented below, is from (Ming et al., 2016).

RNS Problem

3

Reaction:

$$\text{A} \rightarrow \text{B} \rightarrow \text{C}$$

$$2\text{A} \rightarrow \text{D}$$
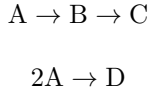
Kinetics:

$$k_1 = 1 \text{ s}^{-1}, \quad k_2 = 2 \text{ s}^{-1}, \quad k_3 = 20 \text{ L mol}^{-1} \text{ s}^{-1}$$

$$\mathbf{r}(\mathbf{C}) = \begin{bmatrix} r_A \\ r_B \\ r_C \\ r_D \end{bmatrix} = \begin{bmatrix} -k_1 c_A - k_3 c_A^2 \\ k_1 c_A - k_2 c_B \\ k_2 c_B \\ k_3 c_A^2 \end{bmatrix} \frac{\text{kmol}}{\text{m}^3 \text{ s}}$$

Feed:

$$\mathbf{C}_{\text{f}} = \begin{bmatrix} c_{A_{\text{f}}} \\ c_{B_{\text{f}}} \\ c_{C_{\text{f}}} \\ c_{D_{\text{f}}} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \frac{\text{kmol}}{\text{m}^3}$$

Objective:

$$q_{in} = 2 \text{ m}^3 \text{ s}^{-1}, \quad \tau = \frac{V}{q_{in}}$$

$$\max_{\tau_{CSTR}, \tau_{PFR}} c_B$$

Additional Constraints:

$$\forall \tau \leq 5 \text{ s}$$

For the above, the molar generation rate $\mathbf{r}$ is only a function of c_A and c_B, and the RNS objective function is only a function of c_B so the AR can be represented by its projection in $\mathbb{R}^2$ ($\{c_A\} \times \{c_B\}$ space) without loss of information.

As a result of some AR theorems regarding two-dimensional constructions, we can conclude that the boundary of this AR is composed of only mixing tanks and trajectories of CSTRs and PFRs. Once generated, two observations can be made of the resulting AR:

- The optimal solution of the RNS problem is a maximum of $c_B = 0.07 \text{ kmol m}^{-3}$ using a design where the feed travels through a CSTR, then a PFR, in series with $\tau_{CSTR} = 0.35$ s and $\tau_{PFR} = 0.25$ s, respectively. With the stated flow rate of $q_{in} = 2 \text{ m}^3 \text{ s}^{-1}$,

these correspond to reactor sizes of $V_{CSTR} = 0.7 \text{ m}^3$ and $V_{PFR} = 0.5 \text{ m}^3$.

- If an additional constraint, such as a hypothetical constraint of $c_B \geq 0.08$, were added to the RNS problem, it can be known to make the RNS problem infeasible since it lies completely above the AR.

Figure 2 shows the resulting AR for the RNS problem.
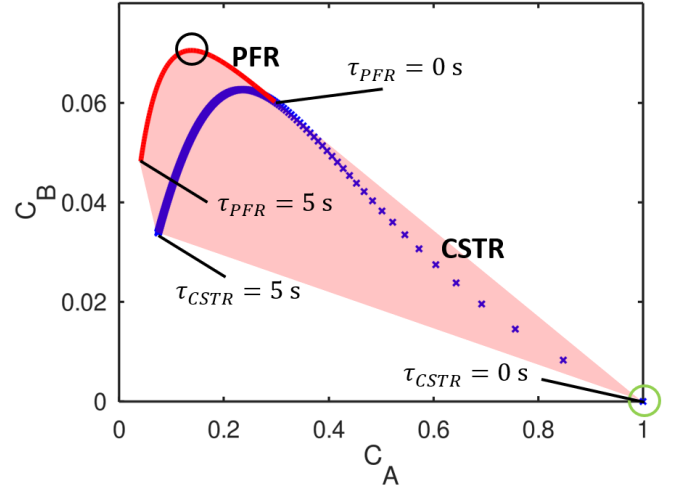
Figure 2: A visualization of the RN-AR (pink), with feed (centre of green ring), and the solution to the problem (centre of black ring). Linear segments on the boundary of the AR are formed by mixing tanks, and the performance of the CSTRs (blue trajectory) and PFRs (red trajectory) are shown for increasing residence times.

The example follows the general procedure of AR-based process synthesis, which proceeds as follows:

1. Generate the AR.
2. Interpret states within the AR to synthesize the corresponding designs.
3. Choose the optimal design.

From this, some remarks can be made about the benefits of AR-based RNS. Firstly, because the first step in AR-RNS is the generation of the AR, if the AR falls outside of the feasible region of the optimization problem, then it is possible to know *a priori* before running any synthesis routine that the RNS problem is infeasible. Secondly, since the AR is generated using reactor model equations, we know that any solution to this RNS problem is

physically realizable (insofar as the models reflect reality). Thirdly, if the solution to the RNS problem corresponds to a RN design lying on the boundary of the AR, then it can be known that it is physically impossible to synthesize a design with better performance. Lastly, were there to be any uncertainty associated with the reaction parameters, these can easily be accounted for by introducing a family of ARs for the possible values of the parameters into the analysis.

The concludes the review of RN-ART. We will now move to a review of heat exchanger models, another key tool on the road towards HEN-ART.

2.2. Heat exchanger models

Heat exchangers can be divided into two large classes based on their operation: co-current and counter-current. In the co-current case, the heat from material in one stream is transferred to material in the other stream such that the residence times of material pair generally shortest-shortest to longest-longest, whereupon the material leaves the heat exchanger. In the counter-current case, the heat from material in one stream is transferred to material in the other stream such that the residence times of the material pair generally shortest-longest to longest-shortest (whereupon the material leaves the heat exchanger).

For this work, it is sufficient that only single-stream idealizations of heat exchangers will be employed. A justification for why this assumption is permissible will be given later in the discussion. For now, general familiarity with the models will suffice.

The models which can be used for both heat exchanger types are the so-called NTU-effectiveness equations for single-stream heat exchangers, which are first-principles models constructed assuming the heat exchanger has a single shell and tube and each stream mixes perfectly with itself (Kays and London, 1984).

Before continuing, a word on notation. Throughout this article, algorithms will be used for the sake of pre-

ciseness. The output of an algorithm in-text will occasionally be represented using the algorithm name. When this occurs, the name of the algorithm will be in boldface with the inputs stated within parentheses. For example, "**function**(x_1, x_2)" stands for the output of the algorithm named "function" with inputs x_1 and x_2.

The heat exchanger models will now be presented in algorithmic form. Starting with the co-current heat exchanger, the archetypical diagram and model equations can be seen in Figure 3 and the algorithm **co-currentHX** below.
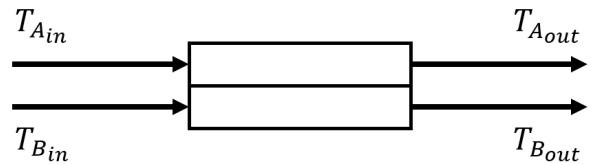

Figure 3: A diagram of a co-current heat exchanger.

In the algorithm, $T_{(\cdot)_{in}}$ and $T_{(\cdot)_{out}}$ are the inlet and outlet temperatures of the streams in °C, $F_{(\cdot)_{in}}$ and $F_{(\cdot)_{out}}$ are the constant mass flow rates of the streams in $\mathrm{kg\,s^{-1}}$, c_{p_1} and c_{p_2} are the heat capacities of the material of the first and second streams in $\mathrm{kJ\,kg^{-1}\,°C^{-1}}$, U is the heat transfer coefficient in $\mathrm{kJ\,s^{-1}\,m^{-2}}$, and A is the heat exchanger area in $\mathrm{m^2}$. Next, the archetypical counter-current heat exchanger and model equations can be seen in Figure 4 and the algorithm **counter-currentHX** below.
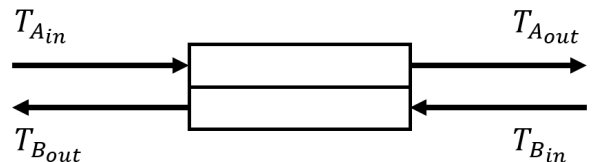

Figure 4: A diagram of a counter-current heat exchanger.

In **counter-currentHX**, the symbols mean the same thing as in the co-current model. These NTU-ε equa-

Algorithm 1: co-currentHX

Input: $T_{1_{in}}, T_{2_{in}}, F_{1_{in}}, F_{2_{in}}, c_{p_1}, c_{p_2}, U, A$

Output: $T_{1_{out}}, T_{2_{out}}, F_{1_{out}}, F_{2_{out}}$

1 **if** $T_{1_{in}} < T_{2_{in}}$ **then**

2 $\quad$ Let $T_{c_{in}} := T_{1_{in}}, T_{h_{in}} := T_{2_{in}}, F_{c_{in}} := F_{1_{in}},$
$\quad\quad F_{h_{in}} := F_{2_{in}}, c_{p_c} := c_{p_1}, c_{p_h} := c_{p_2}$

3 **else**

4 $\quad$ Let $T_{c_{in}} := T_{2_{in}}, T_{h_{in}} := T_{1_{in}}, F_{c_{in}} := F_{2_{in}},$
$\quad\quad F_{h_{in}} := F_{1_{in}}, c_{p_c} := c_{p_2}, c_{p_h} := c_{p_1}$

5 **end**

6 Let $C_c := F_{c_{in}} c_{p_c}$

7 Let $C_h := F_{h_{in}} c_{p_h}$

8 **if** $C_c < C_h$ **then**

9 $\quad$ Let $C_{min} := C_c, C_{max} := C_h$

10 **else**

11 $\quad$ Let $C_{min} := C_h, C_{max} := C_c$

12 **end**

13 Let $C_r := C_{min}/C_{max}$

14 Let NTU $:= UA/C_{min}$

15 Let $\varepsilon := (1 - \exp(-\text{NTU}(1 + C_r)))/(1 + C_r)$

16 Let $q_{max} := C_{min}(T_{h_{in}} - T_{c_{in}})$

17 Let $q := \varepsilon q_{max}$

18 Let $T_{h_{out}} := T_{h_{in}} - q/C_h$

19 Let $T_{c_{out}} := T_{c_{in}} - q/C_c$

20 **if** $T_{1_{in}} < T_{2_{in}}$ **then**

21 $\quad$ Let $T_{1_{out}} := T_{c_{out}}, T_{2_{out}} := T_{h_{out}},$
$\quad\quad F_{1_{out}} := F_{c_{in}}, F_{2_{out}} := F_{h_{in}}$

22 **else**

23 $\quad$ Let $T_{1_{out}} := T_{h_{out}}, T_{2_{out}} := T_{c_{out}},$
$\quad\quad F_{1_{out}} := F_{h_{in}}, F_{2_{out}} := F_{c_{in}}$

24 **end**

Algorithm 2: counter-currentHX

Input: $T_{1_{in}}, T_{2_{in}}, F_{1_{in}}, F_{2_{in}}, c_{p_1}, c_{p_2}, U, A$

Output: $T_{1_{out}}, T_{2_{out}}, F_{1_{out}}, F_{2_{out}}$

1 **if** $T_{1_{in}} < T_{2_{in}}$ **then**

2 $\quad$ Let $T_{c_{in}} := T_{1_{in}}, T_{h_{in}} := T_{2_{in}}, F_{c_{in}} := F_{1_{in}},$
$\quad\quad F_{h_{in}} := F_{2_{in}}, c_{p_c} := c_{p_1}, c_{p_h} := c_{p_2}$

3 **else**

4 $\quad$ Let $T_{c_{in}} := T_{2_{in}}, T_{h_{in}} := T_{1_{in}}, F_{c_{in}} := F_{2_{in}},$
$\quad\quad F_{h_{in}} := F_{1_{in}}, c_{p_c} := c_{p_2}, c_{p_h} := c_{p_1}$

5 **end**

6 Let $C_c := F_{c_{in}} c_{p_c}$

7 Let $C_h := F_{h_{in}} c_{p_h}$

8 **if** $C_c < C_h$ **then**

9 $\quad$ Let $C_{min} := C_c, C_{max} := C_h$

10 **else**

11 $\quad$ Let $C_{min} := C_h, C_{max} := C_c$

12 **end**

13 Let $C_r := C_{min}/C_{max}$

14 Let NTU $:= UA/C_{min}$

15 **if** $C_r \approx 0$ **then**

16 $\quad$ Let $\varepsilon := 1 - \exp(\text{NTU})$

17 **else if** $C_r \approx 1$ **then**

18 $\quad$ Let $\varepsilon := \text{NTU}/(1 + \text{NTU})$

19 **else**

20 $\quad$ Let $\varepsilon := (1 - \exp(-\text{NTU}(1 + C_r)))/(1 - C_r\exp(-\text{NTU}(1 - C_r)))$

21 **end**

22 Let $q_{max} := C_{min}(T_{h_{in}} - T_{c_{in}})$

23 Let $q := \varepsilon q_{max}$

24 Let $T_{h_{out}} := T_{h_{in}} - q/C_h$

25 Let $T_{c_{out}} := T_{c_{in}} - q/C_c$

26 **if** $T_{1_{in}} < T_{2_{in}}$ **then**

27 $\quad$ Let $T_{1_{out}} := T_{c_{out}}, T_{2_{out}} := T_{h_{out}},$
$\quad\quad F_{1_{out}} := F_{c_{in}}, F_{2_{out}} := F_{h_{in}}$

28 **else**

29 $\quad$ Let $T_{1_{out}} := T_{h_{out}}, T_{2_{out}} := T_{c_{out}},$
$\quad\quad F_{1_{out}} := F_{h_{in}}, F_{2_{out}} := F_{c_{in}}$

30 **end**

tions are easily solvable and robust for free choice of inputs (Kays and London, 1984; Bergman et al., 2011).

It is well known that although the initial rate of heat transfer in a co-current heat exchanger is greater than that of a counter-current one of the same heat exchange area, as area increases the counter-current heat exchanger will transfer more heat between the streams (Kays and London, 1984). For example, for very large heat exchangers (approaching the limit of infinite heat exchange area) inputs = $\{T_{1_{in}} = 95, T_{2_{in}} = 10, F_{1_{in}} = 95, F_{2_{in}} = 10, c_{p_1} = 4.186, c_{p_2} = 4.186, U = 10, A = 400\}$, a solution of the co-current heat exchanger model **co-currentHX**(inputs) yields $T_{1_{in}} = 86.9, T_{2_{in}} = 86.9$ ($q = 3219.2$ kW) whereas the a solution of the counter-current heat exchanger model **counter-currentHX**(inputs) yields $T_{1_{in}} = 86.1, T_{2_{in}} = 95.0$ ($q = 3558.1$ kW). This is because the limit of heat exchange in a counter-current exchanger is limited by the feed temperatures, whereas in a co-current exchanger it is limited by a state only intermediate to the feed temperatures. Because of this, and because the heat and temperature of a fluid passing through a heat exchanger is continuous with residence time, any temperature of a stream attained by a co-current heat exchanger is also attainable by the counter-current one with the same feed conditions.

Having finished this review of the necessary heat exchanger models, the background information for the discussion is now complete. The concepts for how attainable region theory may be applied to heat exchanger networks will now be presented.

3. Heat exchanger trajectories

3.1. Motivation for heat exchanger trajectories

The key building block of the AR in RN-ART are trajectories of its main equipment type, the reactor. To adapt ART to HENs, it seems pertinent to examine if similar trajectories can be constructed using the main equipment type of HENS: the heat exchanger. To do this, two observations are made about what a suitable trajectory is in RN-ART. Firstly, trajectories are generated over increasing residence time for all equipment types so that the size and type of equipment can be deduced from a given trajectory point. Secondly, the trajectory is generated in a state-space of interest.

For the first observation, it is noted that for constant thermophysical properties, the design engineer only has control over the heat exchange area and the geometry of the heat exchanger. However, regardless of heat exchanger geometry (e.g. baffles, multiple tube passes, cross-flow, etc.), the maximum attainable heat transfer is always limited by the feed conditions. Since the simplified single-stream counter-current heat exchanger attains this limit, and the goal of AR generation is simply to demarcate these limits, geometry can be ignored—changing geometry only changes the rate at which this limit will be reached. This is why the assumption of single-stream heat exchangers was permissible when choosing the models. Furthermore, we have already established that the counter-current heat exchanger will always be able to attain the range of temperatures possible from a co-current heat exchanger. Then (unlike RN-ART), it can be concluded that trajectories of only one type of equipment need to be considered: that of the single-stream, counter-current heat exchanger, over increasing heat-exchange area.

This leads to the second observation about what "states" are changing within the trajectory. Looking at the counter-current HX model, the outputs are stream temperature and flow rate. Since we have assumed flow rate does not change, the only change is in stream temperature or stream enthalpy content. However, there are some benefits to working in the temperature space. For one, heat exchange is limited by the inlet feed temperature, not by the feed enthalpy content. Since the goal of AR generation is to demarcate the limits of performance, it will be useful to keep the AR in the stream temperature space. A second benefit is that the target performance state of HENs is often the outlet stream temperatures, so the outlet stream temper-

atures are the state of interest. Therefore, for now, the trajectories forming the AR will be represented in stream temperature space.

It is pertinent now to define more precisely what is meant by a "stream". For the purposes of this discussion, a "stream" is the collection of positions within a unique channel through which material can flow. We will associate temperatures with streams. This definition is chosen so that changes in temperature can only be due to heat exchangers, which will greatly simplify the interpretation of designs from the AR later. For example, if we choose instead to say that a stream is "a unique material" like in RN-ART, then the temperature information of streams with the same material composition will be confounded with each other.

The above discussion has given the justification for using trajectories of heat exchangers through stream-temperature space. Much like in RN-ART, we will form trajectories by increasing the residence time, or equivalently, the heat exchange area of a heat exchanger, to construct the AR. To generate these trajectories, one may simply record the output of the counter-current HX model over a mesh of heat exchange areas from 0 to a very large number (an emulation of infinite heat exchange area). In implementations, since the progress of heat exchange decays exponentially with heat exchange area (as seen in the NTU-ε model), the area mesh can be spaced exponentially so as to cover the range of output states using fewer mesh points. The task of "running a heat exchanger trajectory" from a given feed point is given below as algorithm **HXtrajectory**.

Line 2 of **HXtrajectory** indicates that the temperature, flow rate, and heat capacities are those of the two streams indicated by *streampair*. Since streams which did not enter the heat exchanger do not change temperature, the outlet temperature of these streams remains unchanged in the trajectory, hence why the stream temperatures not indicated by *streampair* are simply copied over in Line 8 from the feed to each outlet point. In addition

---

Algorithm 3: HXtrajectory

**Input:** T_{in}, F_{in}, c_p, U, A_{max}, *streampair*, *trajmesh*

Output: *trajectory*

1 Let *trajectory* $:= \varnothing$

2 Let $T_{1_{in}} := T_{in_{streampair_1}}$, $T_{2_{in}} := T_{in_{streampair_2}}$, $F_{1_{in}} := F_{in_{streampair_1}}$, $F_{2_{in}} := F_{in_{streampair_2}}$, $c_{p_1} := c_{p_{streampair_1}}$, $c_{p_2} := c_{p_{streampair_2}}$.

3 Let *areamesh* $:=$ an exponentially-spaced mesh of heat exchange areas having *trajmesh* meshpoints from 0 to A_{max}.

4 **for** *each areamesh$_i$ in areamesh* **do**

5   Let *trajectory$_i$* $:= \{T_{1_{out}}, T_{2_{out}}\} \subset$ **counter-currentHX**$(T_{1_{in}}, T_{2_{in}}, F_{1_{in}}, F_{2_{in}}, c_{p_1}, c_{p_2}, U, areamesh_i)$.

6   **for** *each element T_{in_j} in T_{in}* **do**

7   **if** *neither $j = streampair_1$ nor $j = streampair_2$* **then**

8   Add T_{in_j} to *trajectory$_i$*.

9   **end**

10   **end**

11   Add *areamesh$_i$* to *trajectory$_i$*.

12   Add *trajectory$_i$* to *trajectory*.

13 **end**

14 Add *streampair* to *trajectory*.

---

to stream temperature changes, the output of **HXtrajectory** also contains the trajectory meta-data: associated heat-exchanger area and the stream pairing, which will be useful later in the discussion.

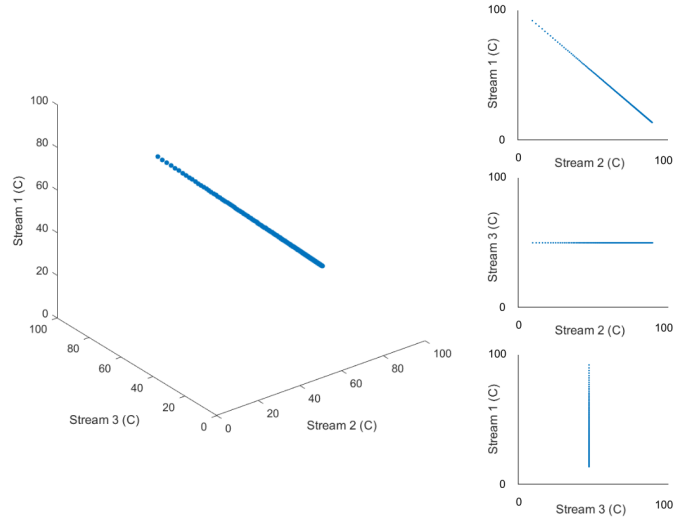### 3.2. Example of a heat exchanger trajectory

To see what a heat exchanger trajectory looks like, consider a process involving three streams of material S1, S2, and S3 with temperatures 95, 10, and 50 °C and flow rates 95, 10, and 50 kg s^{-1}, respectively. Their heat capacities are each 4.186 kJ kg^{-1} °C^{-1}. The trajectory formed from putting S1 with S2 together through a heat exchanger (U=10 kJ s^{-1} m^{-2}, maximum size 400 m^2) with a mesh of 100 points is **HXtrajectory**({95,10,50}, {95,10,50}, {4.186,4.186,4.186}, 400, {1,2}, 100), visualized in Figure 5 for a counter-current heat exchanger and a co-current heat exchanger, for comparison.
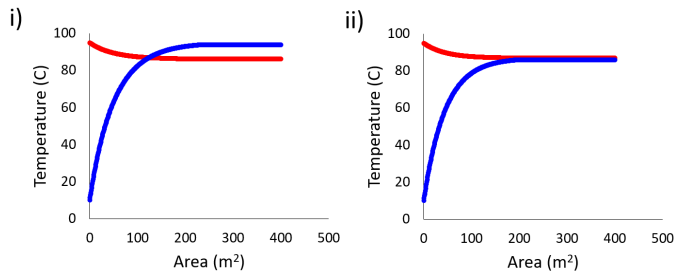
Figure 5: Outlet stream temperatures versus increasing heat exchanger area for the above example in i) a counter-current heat exchanger, and ii) a co-current heat exchanger.

This figure shows that, as stated earlier, the counter-current heat exchanger trajectory attains a wider range of temperatures, including the co-current trajectory range. Below in Figure 6 is the counter-current heat-exchanger trajectory in the stream temperature space.

As in the case above, for a trajectory starting from a state corresponding to unequal stream temperatures, the trajectory will be of some positive length (i.e. some heat gets exchanged). For a trajectory starting from a state corresponding to equal stream temperatures, the trajectory will be of length zero (i.e. no heat is exchanged since

Figure 6: Clockwise from left: The described heat exchanger trajectory in 3-space, and the projection of the trajectory on the YX plane, the YZ plane, and the ZX plane.

there is no temperature difference). In all cases, a trajectory can start from any point in the state space. When a trajectory has reached sufficiently large heat exchange area such that it approaches the limit of outlet stream temperature change, we will say such a trajectory is "maximal". Maximal trajectories represent the heat exchange at infinite heat exchange area, but in implementations a very large heat exchange area can serve as an emulation instead.

4. Properties of the HEN subspaces

Having defined the basic building block of HEN-ARs—the heat exchanger trajectory—some properties of the AR will now be described which can be used to inform a method to generate the AR and to help in the interpretation of AR states as HEN designs.

For the remainder of the article, the notation $_nC_k$ signifies "n choose k"= $\frac{n!}{k!(n-k)!}$ and $_nP_k$ signifies "n permute k"= $\frac{n!}{(n-k)!}$

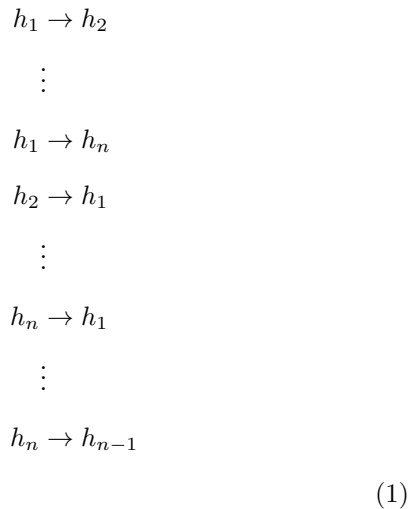### 4.1. The heat exchange and temperature subspaces

The notion of an AR subspace will be introduced by way of a theorem and a proof rather than starting with

definitions so that the context for how the notions are used can become immediately apparent.

Theorem 1. *Let the n-stream temperature space be $\mathbb{R}^n$. Then the AR resides in a stream temperature subspace $\mathcal{T} = B_{\mathcal{H}}^{-1}\mathcal{H}$. Here, $B_{\mathcal{H}}^{-1}$ is a diagonal matrix of heat capacity rates and $\mathcal{H}$ is the corresponding stream enthalpy subspace $\mathcal{H} = \boldsymbol{h}_f + R_H\xi$, where $\boldsymbol{h}_f$ is the n-dimensional coordinate vector of the feed enthalpies, R_H is a matrix of heat exchange reactions, and $\xi \subset \mathbb{R}_{\geq 0}^{nP_2}$ are nP_2-vectors of heat exchange reaction extents.*

Proof. This theorem introduces the notion of the stream enthalpy subspace. This important object is constructed in a similar manner to that of a stoichiometric subspace in reacting systems.

First, recall that the operating principle of heat exchangers necessitates a transfer of heat between streams. Under the constant thermophysical properties assumption, this can be equated to a transfer of enthalpy between streams. By the first law of thermodynamics, a unit of enthalpy is gained by a stream only by the loss of a unit in another. Drawing a connection to the stoichiometric subspace, this is analogous to how a molecular species is only generated by the loss of its reactants. Drawing out the analogy further prompts us to write the *heat exchange reactions* (1).

$$h_1 \rightarrow h_2$$
$$\vdots$$
$$h_1 \rightarrow h_n$$
$$h_2 \rightarrow h_1$$
$$\vdots$$
$$h_n \rightarrow h_1$$
$$\vdots$$
$$h_n \rightarrow h_{n-1}$$

$$\tag{1}$$

For n streams, there are nP_2 ways that they can be paired. So, there are nP_2 ways that heat can be transferred between them. The equations (1) therefore represent nP_2 reactions. Every possible direction in which heat may be transferred is accounted for in (1), whether stream 1 is hotter than stream 2 or vice-versa, stream 1 than stream 3 or vice-versa, and so on. These heat exchange reactions can be represented in the $n \times (nP_2)$ heat exchange matrix R_H as

$$R_H = \begin{bmatrix} -1 & \ldots & -1 & 1 & \ldots & 1 & \ldots & 0 \\ 1 & \ldots & 0 & -1 & \ldots & 0 & \ldots & \vdots \\ 0 & \ldots & \vdots & 0 & \ldots & \vdots & \ldots & 0 \\ \vdots & \ldots & 0 & \vdots & \ldots & 0 & \ldots & 1 \\ 0 & \ldots & 1 & 0 & \ldots & -1 & \ldots & -1 \end{bmatrix} \tag{2}$$

where the rows of R_H are the streams and the columns are the heat exchange reactions. The number of enthalpy units transferred per unit time (called the enthalpy rate) in reaction i can be represented by the extent of reaction ξ_i. The term $R_H\xi$ represents the change in enthalpy rates given a vector of such extents ξ. So

$$R_H\xi = \begin{bmatrix} -1 & \ldots & -1 & 1 & \ldots & 1 & \ldots & 0 \\ 1 & \ldots & 0 & -1 & \ldots & 0 & \ldots & \vdots \\ 0 & \ldots & \vdots & 0 & \ldots & \vdots & \ldots & 0 \\ \vdots & \ldots & 0 & \vdots & \ldots & 0 & \ldots & 1 \\ 0 & \ldots & 1 & 0 & \ldots & -1 & \ldots & -1 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_{nP_2} \end{bmatrix} \tag{3}$$

for all ξ in $\mathbb{R}_{\geq 0}^{nP_2}$. Adding $R_H\xi$ to the vector of feed-stream enthalpies $\mathbf{h}_f$ therefore gives all possible changes in stream enthalpy rates from the feed; the span of these changes is the stream enthalpy subspace $\mathcal{H} = \mathbf{h}_f + R_H\xi$. By the thermodynamic relationship between temperature and enthalpy rate

$$H_1 = F_1 c_{p_1} T_1$$
$$H_2 = F_2 c_{p_2} T_2$$
$$\vdots$$
$$H_n = F_n c_{p_n} T_n$$

$$\tag{4}$$

and the constant thermophysical properties assumption, the temperature-to-enthalpy transform takes the form of a change of basis matrix $B_{\mathcal{H}}$ written as

$$B_{\mathcal{H}} = \begin{bmatrix} F_1 c_{p_1} & 0 & \dots & 0 \\ 0 & F_2 c_{p_2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & F_n c_{p_n} \end{bmatrix} \tag{5}$$

such that $H = B_{\mathcal{H}}T$ for any $T, H \in \mathbb{R}^n_{\geq 0}$. Then $\mathcal{T} = B_{\mathcal{H}}^{-1}\mathcal{H}$, which is the desired result. $\square$

4.2. Temperature subspace dimension

Theorem 2. *Let the n-stream temperature subspace be* $\mathcal{T} \subset \mathbb{R}^n$. *Then* $\dim(\mathcal{T}) = n - 1$.

Proof. By Theorem 1, the following set of column vectors b_H are columns of the heat exchange reactions matrix R_H.

$$b_H = \left\{ \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \dots, \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \right\} \tag{6}$$

Create a matrix whose columns are the elements of b_H. This matrix will have all elements of the first row equal to -1 and the remaining $n - 1$ rows being those of the $(n-1) \times (n-1)$ identity matrix. There are $n-1$ elements in b_H and it is easily seen they are all linearly independent.

Since the matrix formed from b_H has the identity matrix from the second row onwards, those elements from b_H form a basis for all columns of R_H which have 0 in the first row. For example, the column $[01 - 10\dots0]^{\mathrm{T}}$ is created from the first two elements given in (6).

For the rest of the columns of R_H which involve nonzero elements in the first row, those are simply an element of b_H multiplied by -1. Therefore, all columns of R_H are linear combinations of the elements of b_H. So, b_H forms a basis for the column space of R_H. There are $n-1$ elements of b_H, so $\mathrm{rank}(R_H) = n - 1$.

From Theorem 1, the transformation of the extent vector ξ to $\mathcal{T}$ involves only three operations: addition of $n \times 1$ vectors, multiplication by R_H and multiplication by $B_{\mathcal{H}}$. It can be easily seen that $B_{\mathcal{H}}$ is full rank. Addition of $n \times 1$ vectors also preserves dimensions. Therefore, multiplication by R_H is the only transformation which $\dim(\mathcal{T})$ is dependent on. Then, $\dim(\mathcal{T}) = \mathrm{rank}(R_H)$ which was just shown to be equal to $n - 1$. So $\dim(\mathcal{T}) = n - 1$ $\square$

This result of Theorem 2 is consistent with the findings of Hohmann, who was the first to show that the minimum number of heat exchangers to reach a target stream is $n-1$ process stream-process stream heat exchangers (plus any additional utility-stream heat exchangers) (Hohmann, Edward Charles, 1971). This $n - 1$ number is explained by Theorem 2 as the number of heat exchanger trajectory directions needed to span the stream enthalpy subspace $\mathcal{H}$.

Let us now look at some examples of AR subspaces.

4.3. AR subspace example 1

Consider a HENS problem with feed temperatures $T_{fS} = \{95, 10\}$ °C, corresponding flow rates $F = \{95, 10\}$ kg s^{-1}, and corresponding heat capacities $c_p = \{4.186, 4.186\}$ kJ kg^{-1} °C^{-1} Then by Theorems 1 and 2 the subspace is equivalent to

$$\mathcal{T} = T_f + \mathrm{span}\left(\begin{bmatrix} \frac{-1}{(95)(4.186)} \\ \frac{1}{(10)(4.186)} \end{bmatrix}\right) \tag{7}$$

which will take the form of Figure 7.

4.4. AR subspace example 2

Consider a HENS problem with feed temperatures $T_{fS} = \{95, 10, 50\}$ °C, corresponding flow rates $F = \{95, 10, 50\}$ kg s^{-1}, and corresponding heat capacities $cp = \{4.186, 4.186, 4.186\}$ kJ kg^{-1} °C^{-1}. Then by Theorems 1 and 2 the subspace is equivalent to

$$\mathcal{T} = T_f + \mathrm{span}\left(\begin{bmatrix} \frac{-1}{(95)(4.186)} \\ \frac{1}{(10)(4.186)} \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{-1}{(95)(4.186)} \\ 0 \\ \frac{1}{(50)(4.186)} \end{bmatrix}\right) \tag{8}$$

which will take the form of Figure 8.

11
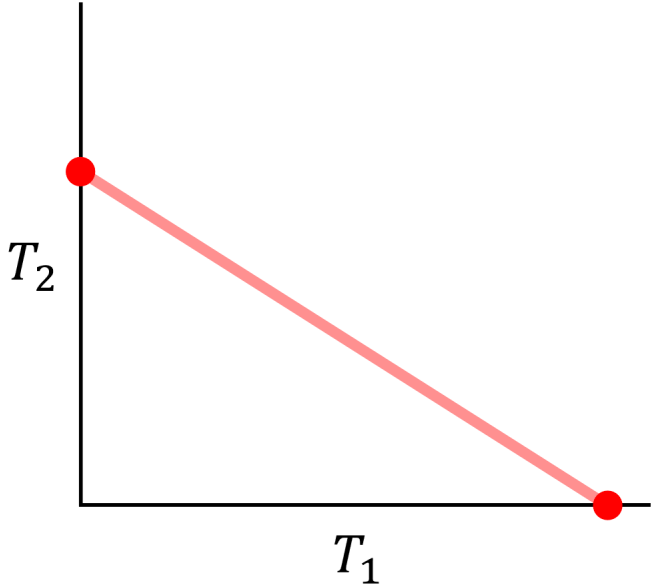
Figure 7: Archetype of the temperature subspace for a two-stream HENS problem. Boundary is in red and inside is in pink.

Figure 8: Archetype of the temperature subspace for a three-stream HENS problem. Boundary is in red and inside is in pink.
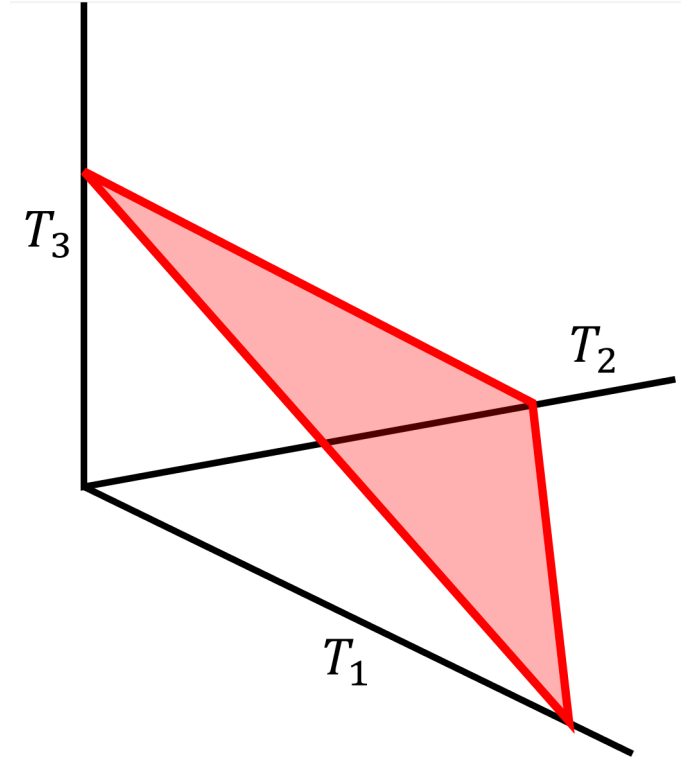
Note that the temperature subspace in absolute temperature units must lie only in the positive orthant of $\mathbb{R}^n$.

An interesting property of the stream enthalpy/heat exchange subspace $\mathcal{H}$ follows from the presented theorems:

Theorem 3. *Let the n-stream stream enthalpy subspace be $\mathcal{H} \subset \mathbb{R}^n$. Let v be an n-dimensional vector whose coordinate in each dimension is 1. Then the orthogonal complement of $\mathcal{H}$ relative to $\mathbb{R}^n$ is $(\mathbb{R}^n \setminus \mathcal{H}) = \mathrm{span}\,(v)$.*

Proof. By Theorem 1, finding the complement of $\mathcal{H}$ relative to $\mathbb{R}^n$ is equivalent to finding the orthogonal complement of the column space of R_H in $\mathbb{R}^n$ (consider the first n columns of R_H). By Theorem 2, this is equivalent to finding the orthogonal complement of the column space of b_H (also in $\mathbb{R}^n$; consider b_H having an additional column of zeros). It is known that the orthogonal complement of the column space of any matrix is equivalent to the null space of its transpose (Anton and Rorres, 2010). Therefore, the orthogonal complement of $\mathcal{H}$ relative to $\mathbb{R}^n$ is $\mathrm{null}(b_H^{\mathrm{T}})$.

The null space of b_H^{T} is equivalent to the solution of $b_H^{\mathrm{T}} x = 0$, which are the equations

$$
\begin{aligned}
-x_1 + x_2 &= 0 \\
-x_1 + x_3 &= 0 \\
&\vdots \\
-x_1 + x_n &= 0
\end{aligned}
$$

$$(9)$$

By Theorem 2, there are $n - 1$ linearly independent equations of n unknowns. Therefore, there is one degree of freedom. Let this degree of freedom be expressed as $x_1 = s$ where s is a scalar constant. Then by (9) $x_1 = s$, $x_2 = s, \ldots,\ x_n = s$ is the null space of A, which is equivalent to $\mathrm{span}\,(sv) = \mathrm{span}\,(v)$. $\square$

5. Properties of the HEN-AR

The existence of the heat exchange and stream temperature subspaces have now been proven, along with some of their properties. These spaces form the "home" of the AR. But they are, so far, an empty home. In this section, we

will prove the existence of the AR and further properties which will be useful later in the discussion.

For all the results in this section, assume we are working in the n-stream temperature subspace $\mathcal{T} \subset \mathbb{R}^n$.

5.1. AR existence, boundedness, and connectedness

Theorem 4. *For any HENS problem, there is an attainable region $A \subset \mathcal{T}$ such that A is a non-empty region bounded in all directions whose comprising points are connected to at least one other point.*

Proof. Existence of the AR: All HENS problems assume the feed conditions are attainable. These include the set of feed stream temperatures $T_{Sf} = \{T_{1f}, T_{2f}, ..., T_{nf}\}$. Any T_{Sf} can always be uniquely associated with the coordinate vector $T_f = (T_{1f}, T_{2f}, ..., T_{nf}) \in \mathbb{R}^n$. That is, $T_f \subseteq A$. Therefore, the attainable region is always non-empty.

Boundedness of the AR: Consider any AR point $a = (a_1, a_2, ..., a_n) \in A$. For real feed stream temperatures $T_{Sf} = \{T_{1f}, T_{2f}, ..., T_{nf}\}$ in Kelvin, $0 < T_1, T_2, ..., T_n < \infty$, the operating principle of heat exchangers allows only $\min(T_{Sf}) \leq a_1, ..., a_n \leq \max(T_{Sf})$. So any AR point has all its coordinates bounded. Therefore, A is bounded.

Connectedness of the AR: Only the feed point is assumed to be attainable without any heat exchanger trajectories, so all other points in the AR must be in heat exchanger trajectories that originate either from the feed, or from others connected by trajectories to the feed. Since temperatures change continuously in a heat exchanger, all heat exchanger trajectories are continuous lines, and therefore every point of the AR is connected to at least one other point. $\square$

Having proved the basic properties of existence, boundedness, and connectedness, some finer details about the AR are presented next.

5.2. Trajectory growth directions in $\mathcal{T}$

Lemma 1. *Let "direction" mean a vector in $\mathbb{R}^n$ or any of its scalar multiples. Then for a HENS problem with con-*

stant thermophysical properties, the heat exchanger trajectories comprising the AR will be straight lines in $\mathbb{R}^n$ parallel to some set of $_nC_2$ directions.

Proof. Heat exchanger trajectories will be straight lines: By Theorem 1, the heat exchanger trajectories only grow according to changes in one $\xi_i \in \xi$ (since only two streams are put in a heat exchanger to generate a heat exchanger trajectory). These changes in ξ_i are then subjected to solely linear mappings to $\mathbb{R}^n$. Therefore, each heat exchanger trajectory will be a straight line.

Heat exchanger trajectories will be lines parallel to at most $_nC_2$ directions: By Theorem 1, for each column vector c_i of R_H, some $c_j = -c_i$ is also a column vector of R_H. Therefore, changes in ξ_i and ξ_j will cause a trajectory to grow in the same direction, by the definition of direction. Therefore, only half of the columns of R_H are associated with unique trajectory growth directions. Since there are $_nP_2$ column vectors of R_H, there are $_nP_2/2 = {}_nC_2$ unique directions a trajectory can grow in. $\square$

Corollary 1. *If the constant thermophysical properties assumption holds, then a trajectory between two streams, represented in their temperature space, will always have negative slope.*

Proof. From the heat exchange reactions (1), we can see that an increase of enthalpy in one stream is always accompanied by a decrease in enthalpy of another stream. Since heat capacity and mass flow rate are positive quantities, then by (4), an increase in outlet temperature in one stream is always accompanied by a decrease in outlet temperature in another stream. This relation describes a negative slope in the stream temperature space. Since a trajectory is a representation of this phenomena, it must be of negative slope. $\square$

Note that if the constant thermophysical properties assumption does not hold, then phase change may occur, and so heat exchange may occur in a heat exchanger without

a corresponding change in outlet temperature. However, this case falls outside of the scope of this article.

5.3. Existence of AR edges and vertices

Lemma 2. *For a HENS problem of constant thermophysical properties, the AR is a polytope in $\mathcal{T}$ whose edges are heat exchanger trajectories. The boundary of the AR is defined by the end points of its edges, which are its vertices.*

Proof. By Theorem 4, all points in the AR must be connected by paths composed of heat exchanger trajectories. Therefore, the boundary of the AR must also be composed of heat exchanger trajectories. By Lemma 1, all paths in the AR are composed of straight lines oriented in a finite number of directions, so segments of the AR boundary which do not lie completely on a finite number of planes are impossible (i.e. smoothly curved boundary surfaces are impossible). Therefore the boundary of the AR is composed solely of straight line segments; edges. Since the AR is bounded in all directions by Theorem 4 and can only have planar boundary surfaces, this implies the AR is a polytope.

Straight edges can be defined by their end points which are the vertices of a polytope, so the AR has vertices which also define its boundary. □

5.4. Inclusivity of the AR boundary

Lemma 3. *Let p be a vertex of the AR with associated stream temperature set p_S. Then a trajectory Q from p which does not change the value of $\min(p_S)$ or does not change the value of $\max(p_S)$ (that is, does not involve the streams associated with $\min(p_S)$ or $\max(p_S)$) for every point in the trajectory) lies completely on the boundary of the AR.*

Proof. There will be a proof for the case where Q is a trajectory which does not change $\min(p_S)$ and a proof for where Q is a trajectory which does not change $\max(p_S)$.

For the case where $\min(p_S)$ does not change in Q: Suppose that Q does not lie completely on the boundary of

the AR. Then all $q \in Q$ where $q \neq p$ are on the interior of the AR. Let B be another trajectory on the boundary of the AR from p. Then all $b \in B$ are on the boundary of the AR and B lies below Q in the dimension associated with $\min(p_S)$. Let q_S and b_S be the associated set of stream temperatures for q and b respectively. Then $\min(b_S) < \min(q_S)$. But $\min(q_S) = \min(p_S)$, and by the operating principle of heat exchangers, no trajectory from p can attain temperatures less than $\min(p_S)$. Therefore, Q is on the boundary of the AR.

For the case where $\max(p_S)$ does not change in Q: Suppose Q does not lie completely on the boundary of the AR. Then all $q \in Q$ where $q \neq p$ are on the interior of the AR. Let B be another trajectory on the boundary of the AR from p. Then all $b \in B$ are on the boundary of the AR and B lies above Q in the dimension associated with $\max(p_S)$. Let q_S and b_S be the associated set of stream temperatures for q and b respectively. Then $\max(b_S) > \max(q_S)$. But $\max(q_S) = \max(p_S)$, and by the operating principle of heat exchangers, no trajectory from p can attain temperatures greater than $\max(p_S)$. Therefore, Q is on the boundary of the AR. □

Note that this lemma does not differentiate between boundary trajectories which are AR edges and those which are on the boundary but lie strictly within higher-dimensional faces.

A related property of edges follows from Lemma 3

Lemma 4. *If Q is an AR edge from some vertex p to some vertex q, the trajectory from p to q is one which does not involve a stream associated with $\min(p_S)$ or $\max(p_S)$ (where p_S is the set of stream temperatures associated with p).*

Proof. By Lemma 2, Q is a trajectory lying completely on the boundary of the AR. By Lemma 1, the direction of a trajectory is given by the choice of the two streams involved. So, starting at vertex p, choose two streams which create a new trajectory P in the same direction as

14

Q with the property of Lemma 3 which keeps it completely on the AR boundary. By Lemma 2, P will end at q since q is adjacent to p. Since an edge is uniquely defined by its starting vertex and ending vertex, Q is identical to P. P has the property that it did not involve a stream associated with $\min(p_S)$ or $\max(p_S)$, so that is also the case for Q. $\qquad\square$

5.5. Location of the feed

Lemma 5. *Let the n-stream temperature subspace be $\mathcal{T} \subset \mathbb{R}^n$. Then the point in the temperature space corresponding to the feed is an extreme point of the AR with respect to $\mathcal{T}$.*

Proof. By Theorem 4, the feed point T_f associated with stream temperatures $T_{Sf} = \{T_{1f}, T_{2f}, ..., T_{nf}\}$ is always part of the AR. By the operating principle of heat exchangers, no trajectory can contain a point p with associated stream temperatures p_S such that $\min(p) < \min(T_{Sf})$ or $\max(p) > \max(T_{Sf})$. That is, the feed point lies on the intersection of two $(n-1)$-dimensional planes representing temperature inequality constraints in $\mathcal{T}$. Therefore, the feed is an extreme point of the AR. $\qquad\square$

5.6. Convexity of the AR boundary in $\mathcal{T}$

By "convex", we mean a plane can be drawn through any vertex of the AR boundary and the rest of the boundary will lie only on one side of the plane.

Theorem 5. *Let the n-stream temperature subspace be $\mathcal{T} \subset \mathbb{R}^n$. Then the boundary of the AR is convex in $\mathcal{T}$.*

Proof. Let a "concavity" be segment of the AR boundary wherein a vertex lies within the convex hull of the AR. By this definition then, a figure with no concavities will be convex.

By Lemma 2, all vertices are connected by at least two edges, and all edges are between two streams. So all concavities can be represented by its projection in the temperature space $T_1 \times T_2$ of any two streams, call them S1 and

S2. Furthermore, if a vertex is not in a concavity in any two-stream projection, then it is not part of a concavity. The proof will show the boundary of the AR is convex by showing that all concavities of the AR are impossible.

All trajectories represented in the two-stream space will be either S1 exchanging heat with S2 (diagonal line) or S1 or S2 exchanging heat with any third stream (horizontal and vertical line, respectively). There are thus only these three families of trajectory directions which form the shape of the AR around each vertex.

A trajectory can move in the diagonal directions only if it does not lie on the line $T_1 = T_2$. Call the region above the line region A (that is, where $T_2 > T_1$). Call the region below the line region B (that is, where $T_1 > T_2$). If a diagonal trajectory starts in region A, it can only proceed downwards. If a diagonal trajectory starts in region B, it can only proceed upwards.

The AR with $n = 2$ streams is convex, since the AR is only a diagonal line.

Now for ARs with $n > 2$ streams. Consider a trajectory lying completely on the AR boundary from vertices p to q (although not necessarily lying in the convex hull). This trajectory pq may be in the positive or negative horizontal direction, positive or negative vertical direction, downwards diagonal direction if beginning in A, or upwards diagonal direction if beginning in B. It may also be completely in A, completely in B, or have parts in both A and B. Therefore, there are exhaustively sixteen possible cases that this trajectory can be in, shown in Figure 9.

The terminology we will use is that if xyz is a concavity, then the AR is in a region which is on a particular side of trajectories xy and yz such that y is in a concavity. For example, in Figure 9i), if pqs is a concavity then the AR is above pq and to the right of qs. Then, q is in a concavity. That xyz is a concavity also implies that for the vertical trajectories r and r', only r is possible from q.

Note that for all cases, both a positive and a negative horizontal trajectory, or both a positive and negative
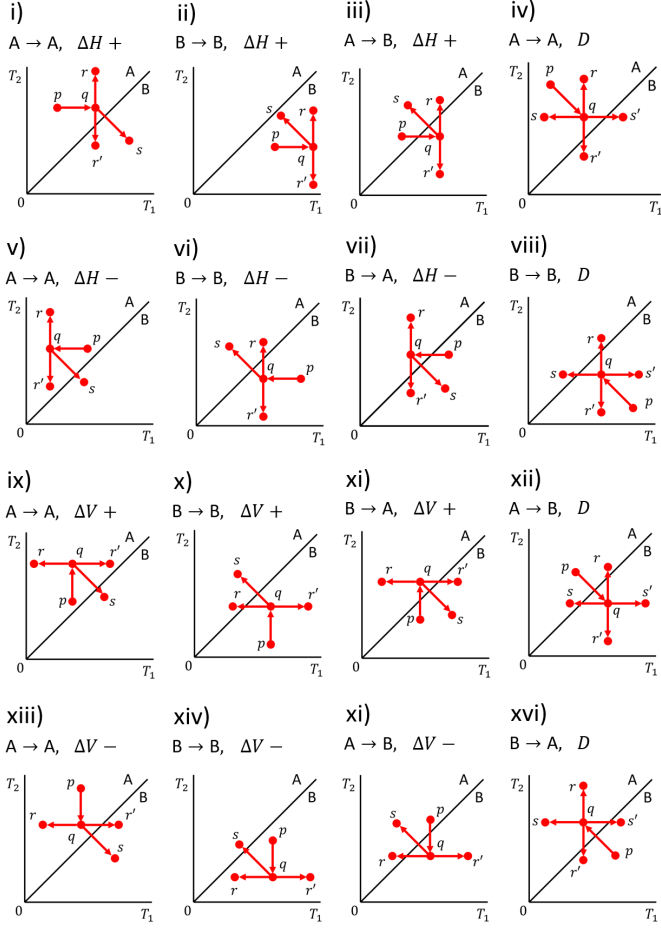
Figure 9: All possible instances of trajectory pq in the temperature space and the possible trajectories from vertex q afterwards. The statement $X \to Y$ means that the trajectory starts at p in region X and ends at q in region Y. The symbols $\Delta H + / -$ mean trajectory pq is in the positive/negative horizontal direction respectively, $\Delta V + / -$ mean trajectory pq is in the positive/negative vertical direction respectively, and D means the trajectory is in the diagonal direction.

vertical trajectory, cannot come from q in all two-stream projections since then q would only strictly be an edge point and not a vertex. Also note that a trajectory from q cannot be in the same direction as pq in all two-stream projections since then that would also imply q is only an edge point. We therefore reject these cases in the analysis. With all the above considerations in mind, we will show that q cannot be in a concavity in all of these sixteen cases.

For case i), the concavity pqr is impossible since qr implies a third stream is warmer than S2 at all points of pq, so an upwards vertical trajectory could be made on any point in pq, making it not an AR edge. The concavities

pqs and pqr' are impossible since a downwards diagonal trajectory can be made on any point in pq since pq lies entirely within region A, making it not an AR edge. Since these are all the possible concavities in the case, q is not in a concavity in case i).

For case ii), the concavities pqs and pqr are impossible since an upwards diagonal trajectory could be drawn on any point in pq, making it not an AR edge. The concavity pqr' is impossible since qr' implies there is some third stream warmer than S2 at all points of pq so a downwards vertical trajectory can be made on any point in pq, making it not an AR edge. Since these are all the possible concavities in the case, q is not in a concavity in case ii).

For case iv), the concavity pqr is impossible since qr implies there is a third stream warmer than S2 at some points of pq so an upwards vertical trajectory can be made on some parts of pq, making it not an AR edge. The concavity pqs' is impossible since qs' implies there is some third stream warmer than S1 at all points, so a rightwards horizontal trajectory can be made at all points of pq making it not an AR edge. The concavity pqr' is impossible since qr' implies there is some third stream warmer than S2 at all points of pq so a downwards vertical trajectory can be made on any point in pq, making it not an AR edge. The concavity pqs is impossible since qs implies there is a third stream cooler than S1 at some points of pq so a leftwards horizontal trajectory can be made on some parts of pq, making it not an AR edge. Since these are all the possible concavities in the case, q is not in a concavity in case iv).

For the remaining cases, their possible concavities can be identified with the concavities already discussed above, so similar arguments for cases with the same directions of possible trajectories from q can be made. For example, by the directions of trajectories from q, identify case vi) with case i). Then, identify the concavity pqr' of case vi) with concavity pqr in case i) and argue that it is impossible by an analogous argument as used for pqr in case i). Here,

it would be that since qr' implies a third stream is cooler (instead of warmer) than S2 at all points of pq, a downwards vertical trajectory could be made on any point in pq, making it not an AR edge. So concavity pqr' of case vi) is impossible.

Therefore, for cases vi), x), and xiii), a similar argument as for case i) can be made to conclude q is not a concavity in these cases either; for cases iii), v), vii), ix), xi), xiv), and xi), a similar argument as for case ii) can be made to conclude q is not a concavity in these cases either; and, for cases viii), xii), and xvi), a similar argument as for case ii) can be made to conclude q is not a concavity in these cases either.

This covers all sixteen cases. Since these sixteen cases are exhaustive, there is no case where q can be in a concavity. Since q can be any vertex of the AR, we must conclude no vertex of the AR can be in a concavity, so the AR has no concavities. Therefore, the AR is convex. $\qquad\square$

The following result follows from this convexity theorem, and shows that every point in the AR is attainable.

Corollary 2. *The AR has no handles nor voids.*

Proof. If the AR has handles or voids, then the surface of the handle or void lying strictly in the convex hull of the AR would contain a concavity of the AR. But by Theorem 5, it is impossible for the AR to have concavities. Therefore, the AR has no handles nor voids. $\qquad\square$

6. HENARG: An AR-generating Algorithm

These AR properties can be used to generate the set of boundary points of the AR by the following reasoning. By Lemma 5 (feed location), the feed is an extreme point. Then, by Lemma 3 (boundary inclusivity), all the trajectories incident on the feed which have the necessary condition to be completely on the boundary can be generated. Thus, the edges incident on the feed will also be generated, by Lemma 4. The terminal points of the edges

will be other extreme points, vertices, on which this process can again be repeated. Since all edges are adjacent to other edges on the AR by Lemma 2, the entire AR will be generated after a finite number of iterations. The generation of the AR by such a process resembles the growth of a tree starting from the feed point, whose branches are the emanating edge trajectories.

The conclusion of this reasoning is stated as the following theorem.

Theorem 6. *Any method which generates the trajectory tree of boundary trajectories from the feed point will also generate all AR edges.*

It is this theorem which will serve as the basis for the method of generating the AR, which we will now discuss in more detail.

A method which generates a list of all the possible trajectories from a point given the number of streams is the algorithm **completegraph**, which takes as input the number of streams involved in the HENS problem (the "n" of "$\mathbb{R}^n$") and returns as output a set of pairs of streams, each stream being denoted by a number between 1 and n. The number of elements of this set will be $_nC_2$ unique pairs, which is also known to be the number of edges of a complete graph on n vertices. Algorithm **completegraph** is given below.

As an example, consider $n = 4$. Then, **completegraph**$(4) = \{\{2,1\}, \{3,1\}, \{3,2\}, \{4,1\}, \{4,2\}, \{4,3\}\}$.

In order to generate the trajectory tree, we go through the list of stream pairs specified by **completegraph**(n) and generate the "branches"—the trajectories which lie completely on the boundary—from known extreme points. The points from which these trajectories were run from are then moved from the "branching" set to the "branched" set and the procedure repeats itself on the ends of the new branches. Let the *attained point (branched) set* be called $\mathcal{A}$, the *explorable point (branching) set* be called $\mathcal{E}$ (these points are "explored" by seeing whether new boundary

trajectories can be generated from them), and the *completed explorable point (branch) set* be called $\mathcal{C}$ (because these points are created by completing the generation of all boundary trajectories on an explorable point). By these definitions, $\mathcal{A} \cup \mathcal{C}$ is the attainable region.

At the end of each iteration, the end point of each trajectory will be put into $\mathcal{C}$, and then the point from which the trajectory started will be removed from $\mathcal{E}$ and added to $\mathcal{A}$. All elements of $\mathcal{C}$ will then be moved to $\mathcal{E}$ and the next iteration can begin. In this way, the "branched" points $a \in \mathcal{A}$ which have been explored are inactivated, and the "branch" points $c \in \mathcal{C}$ are points from which new trajectories will emanate from.

Because the process always adds more AR edges if it is possible to, if the AR does not change between successive iterations then no new points are attainable by heat exchanger trajectories. By definition, the AR is therefore complete. It may seem difficult to check whether the AR really has not changed between iterations, since the AR generated by this procedure may involve many hundreds or thousands of trajectory end points (remember, not every trajectory generated will be an AR edge). However, by Theorem 5, we can check whether the AR has changed using the equivalent condition of checking whether its convex hull has changed. A convex hull algorithm such as Quickhull can be used to return the convex hull (vertices) of the AR in $\mathcal{T}$, which is stated in simplified form as algorithm **CHull** below.

Algorithm 4: completegraph

Input: n

Output: *pairs*

1 Let the set of pairs *pairs* $:= \varnothing$.

2 Let the set of numbers corresponding to each stream *stream* be the integers from 1 to n.

3 Let *pairnumber* $:= 0$.

4 **for** *each element $stream_i$ in stream* **do**

5   Let $s_1 := stream_i$.

6   **for** *each element $stream_j$ in stream* **do**

7   Let $s_2 := stream_j$.

8   **if** $s_2 := s_1$ **then**

9   **break**. (Go to next iteration of outermost for-loop.)

10   **else**

11   Let *pairnumber* $:= pairnumber + 1$.

12   Let $pair_{pairnumber} := \{s_1, s_2\}$.

13   Move $pair_{pairnumber}$ into *pairs*.

14   **end**

15   **end**

16 **end**

Algorithm 5: CHull

Input: *points,dimension*

Output: *convexhull*

1 Let *convexhull* $:= convexhull \subset points$, the extreme points of the convex hull in a space of dimension *dimension* of the set *points* given by some convex hull algorithm such as Quickhull.

Note that the convex hull will be in the $(n-1)$-dimensional temperature subspace $\mathcal{T}$, so the use of **CHull** cannot be

done with *points* in $\mathbb{R}^n$. This problem can be solved using the well-known fact that edges and convexity of polytopes are preserved in linear transformations from $\mathbb{R}^k$ to $\mathbb{R}^k$ (Sanyal and Ziegler, 2010). Since the AR is $(n-1)$-dimensional, simply truncate *points* to any $n-1$ coordinates.

When **CHull** gives the same output between successive iterations, the AR has been generated and we stop the process. Now we will discuss implementation. First, the mesh for each trajectory can be reduced to just two mesh points (the beginning and end of the trajectory) since the vertices completely determine the AR. Second, the test for whether a trajectory will lie on the AR boundary can be carried out as in algorithm **isbtraj** below, whose inputs are the stream temperature set corresponding to a point in temperature space and a pair of two numbers denoting a pair of streams. The output is the truth value of whether the resulting trajectory lies on the AR boundary. One can see the test for the boundary edge condition of 4 in the algorithm.

Algorithm 6: isbtraj

Input: p_S, *pair*

Output: *truthval*

1 Let $p_{max} := \max(p_S)$ and $p_{min} := \min(p_S)$.
2 **if** $\{p_{S_{pair_1}}, p_{S_{pair_2}}\} = \{p_{max}, p_{min}\}$ **then**
3   *truthval* = **false**.
4 **else**
5   *truthval* = **true**.
6 **end**

Note when the HENS problem involves only $n = 2$ streams, the minimum and maximum stream temperatures are the only ones available so **isbtraj** cannot be used. However, the AR in this case is simply a single trajectory between the two streams.

With these considerations in mind, the AR-generation method for $n > 2$ streams is now presented in more precise form as algorithm **HENARG** (Heat Exchanger Network Attainable Region Generator), found below.

Note that line eleven in **HENARG** is the reason why the streampair meta-data is included in each stream trajectory created by **HXtrajectory**—it prevents the algorithm from re-tracing the trajectory it just ran (pairing the same streams together at the end of their trajectory). This is necessary for at least two reasons. First, if the trajectories are maximal (i.e. *maxarea* is very large) then the same stream pair cannot exchange more heat along the same trajectory by definition, so it would not help in finding new extreme points. Second, if the heat exchange area is limited by the design engineer to some non-maximal *maxarea* (for example, due to capital cost constraints), then it would probably be unhelpful to generate the AR where **HENARG** is allowed to pair up the same two streams sequentially through many heat exchangers. Of course, for the cases where one does want to consider these HEN designs, the algorithm can be modified as needed.

Algorithm **HENARG** generates the trajectory tree by the reasoning given in this section. So by Theorem 6, it generates the AR. Two examples will now be presented to demonstrate how the algorithm generates the AR.

6.1. HENARG example 1

Consider the three-stream HENS problem in Section 4.4 with maximum heat exchanger areas of 400 m^2 and two mesh points per trajectory (only two mesh points—the start and end point—are necessary since the AR can be equivalently defined by its vertices). Then **HENARG**(T_f, F, cp, U, 400, 2) is visualized as Figure 10 and Figure 11 below. Figure 10 depicts the trajectories generated during the execution of **HENARG**, and Figure 11 is the output of **HENARG**.

6.2. HENARG example 2

Consider a four-stream HENS problem with feed stream temperature point $T_f = (95, 10, 50, 10$ °C, corresponding flow rates $F = \{95, 10, 50, 50\}$ kg s^{-1}, corresponding heat
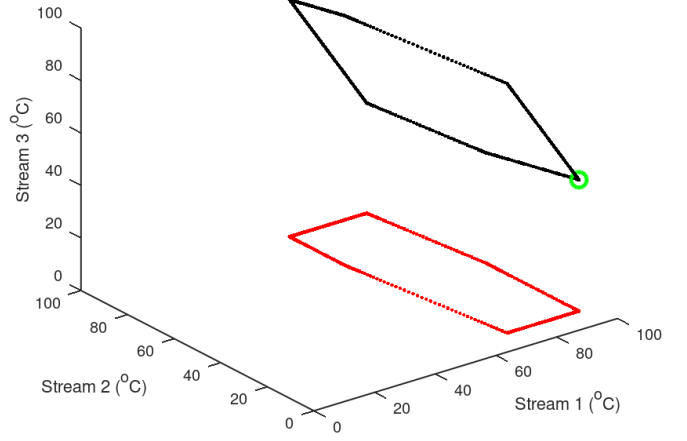
Algorithm 7: HENARG

Input: $feedpoint$, $flowrates$, $heatcaps$, U, $maxarea$, $trajmesh$

Output: AR

1 Let n be the number of elements in $feedpoint$.

2 Let $pairs =$ **completegraph**(n).

3 Let $AR := feedpoint$.

4 Let $ARprev := \varnothing$.

5 Let the set of completed explorable points $\mathcal{C} := \varnothing$.

6 Let the set of attained points $\mathcal{A} := feedpoint$.

7 Let a set of explorable attained points
$\quad \mathcal{E} := feedpoint$.

8 **while** $AR \neq ARprev$ **do**

9   **for** *each* p_i *in* $\mathcal{E}$ **do**

10   **for** *each* $pairs_j$ *in* $pairs$. **do**

11   **if** $pairs_j$ *corresponds to the stream-pair used to generate* $point_i$ **then**

12   **continue** (start next iteration of inner for-loop)).

13   **end**

14   **if** ***isbtraj***$(p_{iS},pairs_j)=$***false*** **then**

15   **continue**.

16   **end**

17   Let $trajectory :=$**HXtrajectory**$(p_i,$ $flowrates$, $heatcaps$, U, $maxarea$, $pair_j$, $trajmesh)$.

18   Copy the final point of $trajectory$ to $\mathcal{C}$.

19   **end**

20   Move p_i from $\mathcal{E}$ to $\mathcal{A}$.

21   **end**

22   Let $A := \mathcal{A} \cup \mathcal{E}$.

23   Let $ARprev := AR$.

24   Let $AR :=$**Chull**$(A, n-1)$.

25   Move all points from $\mathcal{C}$ to $\mathcal{E}$.

26 **end**

Figure 10: The trajectories generated by HENARG for the three-stream HENS problem example from the specified feed point (centre of green ring).

capacities $cp = \{4.186, 4.186, 4.186, 4.186\}$ kJ kg^{-1} °C^{-1}, maximum heat exchanger areas of 400 m^2 and two mesh points per trajectory. Then HENARG(T_f, F, cp, U, 400, 2) is 3-dimensional, and the AR can be visualized as a projection onto the first three stream temperature dimensions, seen in Figures 12 and 13 below.

From these examples, one may guess that the number of required iterations to generate the AR is $_nC_2$ iterations. Our examples are consistent for this: for two streams it takes only $_2C_2 = 1$ iteration to generate the AR, for three streams it took $_3C_2 = 3$ iterations, and for four streams it took $_4C_2 = 6$ iterations. However, this is only a conjecture.

This concludes the discussion on how to generate the AR and some of its relevant properties. The next section will discuss how to use the AR to synthesize feasible HENs.

7. Synthesis of feasible HENs

The goal of this section is to present a method aided by the AR to synthesize a feasible HEN. It is important to remember that although process synthesis typically involves the selection of an optimal design among alternatives, optimization is not within the scope of this article. Instead, the goal is simply to synthesize a feasible HEN, one which 1) transforms inlet feed conditions to the target outlet feed
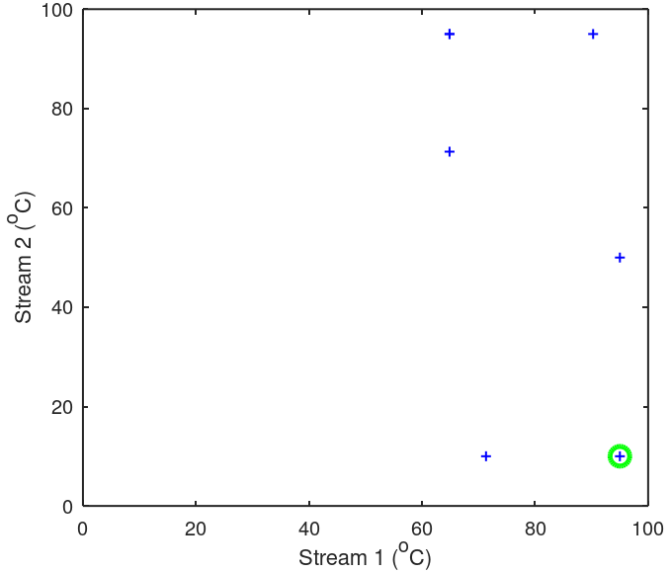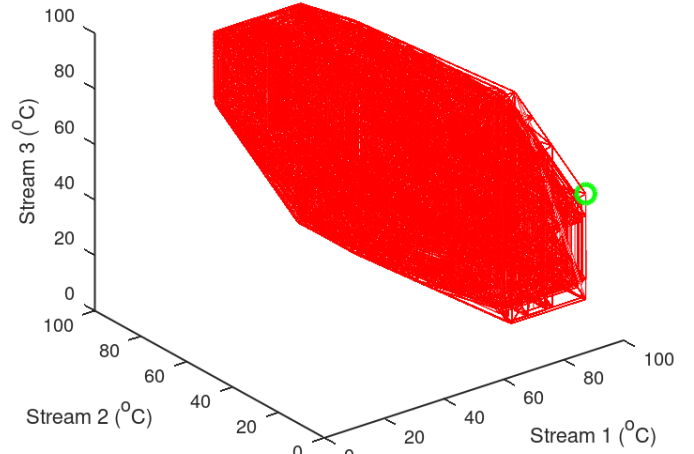
Figure 11: The output of HENARG—the vertices of the AR from the $n = 3$-stream HENS problem example with the specified feed point (centre of green ring). HENARG terminated on iteration 4.

conditions, while 2) satisfying any additional constraints. The former task will be discussed first.

The transformation of the feed stream conditions to the outlet stream conditions corresponds to finding the set of heat exchanger trajectories which go from the inlet state to the target temperature state in the AR. However, the $(n-1)$-dimensional AR will almost never contain an arbitrary target point in $\mathbb{R}^n$. Therefore, a HEN composed of solely process-process stream heat exchangers will almost never be able to transform the feed to the target. This is why utilities—extra-process sources of heat exchange material (e.g. steam and cooling water)—will be required when synthesizing a HEN regardless of what method is used. Therefore, all solutions will involve a combination of process-process stream heat exchangers and process-utility heat exchangers.

Although solely process-utility heat exchanger trajectories can bring the streams from the feed to the target stream conditions, this is typically not the only feasible solution considered in a synthesis method since using utilities for all of the heat transfer is typically expensive. Therefore, let us consider the more interesting feasible solution

**Figure 12:** The trajectories generated by HENARG for the four-stream HENS problem example.
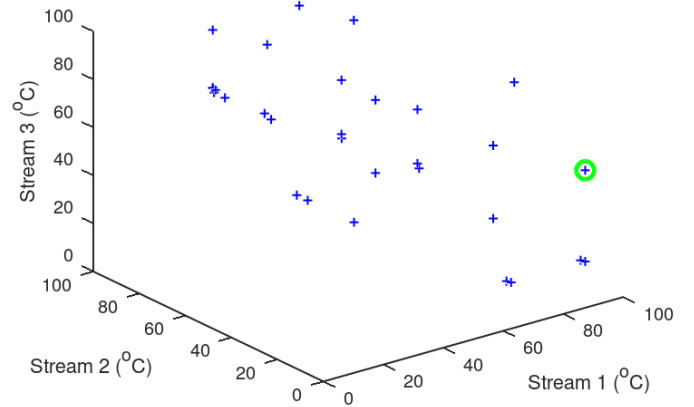
Figure 13: The output of HENARG—the vertices of the AR for the four-stream HENS problem example. HENARG terminated on iteration 7. Also visible is the feedpoint for reference.

where a combination process-process and process-utility heat exchangers are used. A method to do this will be described below.

Consider the example of the family of solutions using process-process heat exchanger trajectories to bring some streams to their target state, and then process-utility heat exchanger trajectories to make up the difference on the remaining ones. To do this, first generate the AR using **HENARG**. Second, project the target stream state onto the temperature subspace along each one of the n dimensions. Among these images, check if one falls within the

AR. If an image falls within the AR, navigate to it from the feed using process-process heat exchanger trajectories, and then make up the difference using a process-utility trajectory for the stream along whose dimension the projection was carried out. If no image falls within the AR, navigate as close as possible towards the closest point to the boundary of the AR and then make up the difference using as many process-utility heat exchanger trajectories as necessary. Figures 14 and 15 below show typical solution trajectories for both these cases.
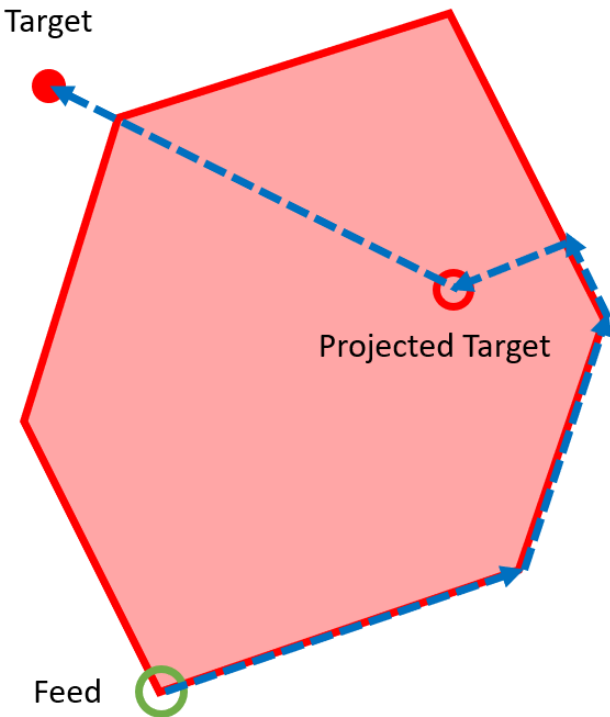
Figure 14: Typical solution trajectory for a target point whose projection along a dimension lies within the AR.

The procedure always consists of projecting the target point onto the temperature subspace along one dimension, navigating to a point within the AR, followed by navigating off to reach the target point. These steps will now be described in detail.

7.1. Projection of the target onto $\mathcal{T}$

Since the temperature subspace $\mathcal{T}$ is $(n-1)$ dimensional, it can be represented in $\mathbb{R}^n$ as a $(n-1)$-dimensional
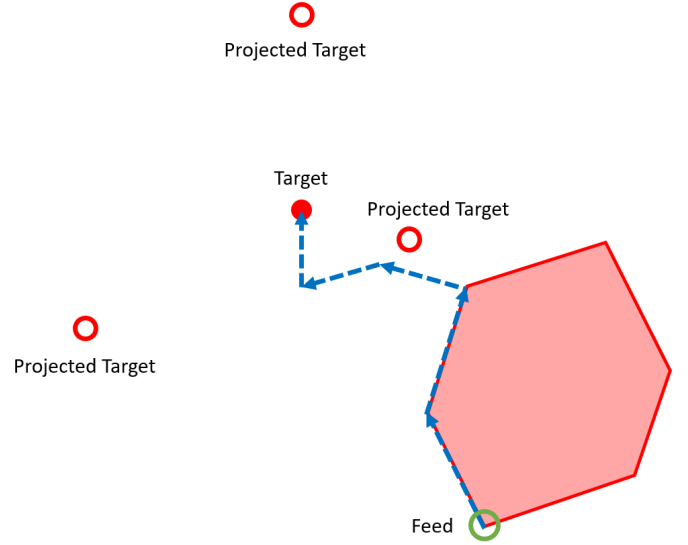
Figure 15: Typical solution trajectory for a target point whose projections along the dimensions onto the stream temperature subspace do not lie within the AR.

plane

$$a_1 T_1 + a_2 T_2 + \ldots + a_n T_n = b \qquad (10)$$

where $a_1, \ldots, a_n$ are constants equivalent to a vector orthogonal to $\mathcal{T}$. In Equation (10), b is solved by substituting in a point $T = (T_1, \ldots, T_n)$ which is known to lie within $\mathcal{T}$. Once the equation of the plane is determined, the projection of the target point onto the plane along each dimension can be done by substituting the appropriate coordinates of the target point into (10) and solving for the remaining one.

To find $a_1, \ldots, a_n$, a vector $u \in \mathbb{R}^n$ orthogonal to $\mathcal{T}$ must be found. Let v be an n-vector of ones. Then by Theorem 3 (complement), the orthogonal vector is simply $u = B_H^{-1} v$. Then $a_1 = (F_1 cp_1)^{-1}$, $a_2 = (F_2 cp_2)^{-1}$, $\ldots$, $a_n = (F_n cp_n)^{-1}$. Now, by Lemma 5 (feed location), the feed $T_f = (T_{1f}, \ldots, T_{nf})$ is a known point in $\mathcal{T}$. Plugging these into (10) gives the value of b as

$$b = \sum_{i=1}^{n} (F_i cp_i)^{-1} T_{if} \qquad (11)$$

Then, the kth coordinate $k \in [1, n]$ of the projection of a

point x onto $\mathcal{T}$ along the kth dimension is given by

$$C_k(x) = \frac{b - \sum_{i=1}^{k-1} a_i x_i - \sum_{i=k+1}^{n} a_i x_i}{a_k} \qquad (12)$$

which gives the projection $P_k(x)$ of point x onto $\mathcal{T}$ as

$$P_k(x) = (x_1, \ x_2, \ \ldots, \ x_k = C_k(x), \ \ldots, \ x_n) \qquad (13)$$

From these calculations we can find the projection of the target point onto $\mathcal{T}$ along each kth dimension.

7.2. Navigating to a point within the AR

Since heat exchanger trajectories in temperature space are straight lines in unique directions, it might seem promising at first to find the trajectories which go from the feed T_f to the projected target point $P_k(x)$ using basis vector decompositions of the displacement vector $d = T_f - P_k(x)$, treating the $_nC_2$ unique trajectories as possible basis vectors. However, this approach does not reliably work for at least two reasons. First, some vectors which d are decomposed into may leave the AR. Second, the vectors which d are decomposed into may have a length which is physically impossible. For example, if a trajectory happens to start very close to the $T_1 = T_2$ line for some two streams, then it may not be able to be as long as what is required to reach the target point. The basis vector decomposition cannot take these limitations into account. We therefore desire a method which can avoid these problems and reliably synthesize feasible heat exchanger networks.

In the following part of the discussion, we will see that a method using optimization techniques can be used to carry out the synthesis step. Readers familiar with linear optimization may have already remarked on how similar the AR looks to the feasible regions of linear programming problems. Indeed, we will use some concepts from the famous "simplex search" optimization method of Dantzig to solve this problem (Dantzig, 1990). Simplex search is a method for solving linear optimization problems in which the optimal state is found by starting at a vertex of a convex polytope of inequality constraints, and traversing the edges of the polytope until the optimal point is found. More information on simplex search can be readily found in many optimization textbooks (Rardin, 2016).

First, recall that the output from algorithm **HENARG** is the set of vertices which denote the AR. The first step required to use simplex search to solve our problem is to convert this vertex representation of the AR, called its V-representation, into a representation of it as an intersection of inequality constraints within the temperature space, called its H-representation. This H-representation takes the form of a linear system of inequalities

$$Ax \leq b \qquad (14)$$

where A is a matrix of real coefficients, x are the state variables, and b are real constants.

The second step is to introduce a variable for each inequality constraint whose value is the displacement of the projected target point from the inequality constraint. Let each variable be called a_j for the jth row of (above) such that

$$Ax + a = b \qquad (15)$$

This is the same idea as the variables of the simplex search method from which one can derive the active constraint set, so the same name—"slack variables"—will be used here. These variables become 0 when the corresponding inequality constraint is "active". That is, when a state x is on the $(n-1)$-dimensional plane denoting the jth inequality constraint, then $a_j = 0$.

The third step is to draw a line through the desired point in the AR in one of the $_nC_2$ directions corresponding to a heat exchanger trajectory, and find the two $(n-2)$-faces of the AR which it intersects with. This can be done by traversing along the line in small steps until one of the a_j slack variables becomes 0. The corresponding inequality constraint is the intersecting face. Then, solve for the point of intersection on the face and run the corresponding heat exchanger trajectory from that point to see if it can reach the target point. If it can, proceed to the next step. If it

cannot for either of the two faces which the line intersects with, repeat with another line of another one of the $_nC_2$ directions until it does. Figure 16 below shows this process for $n = 4$.
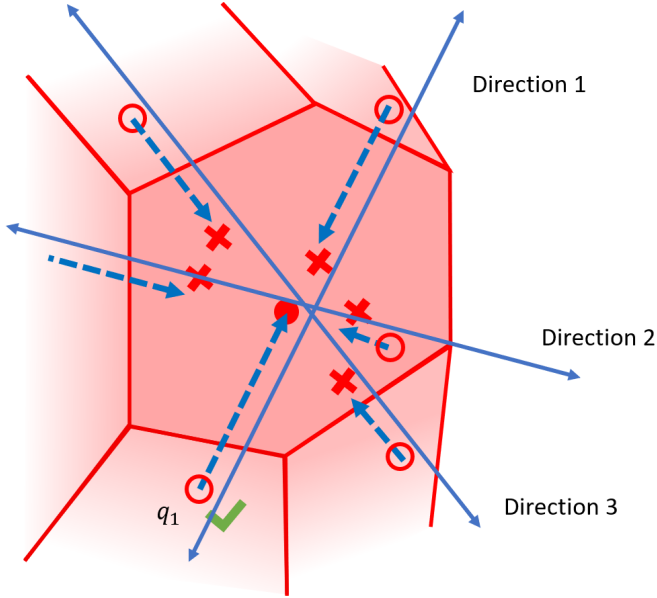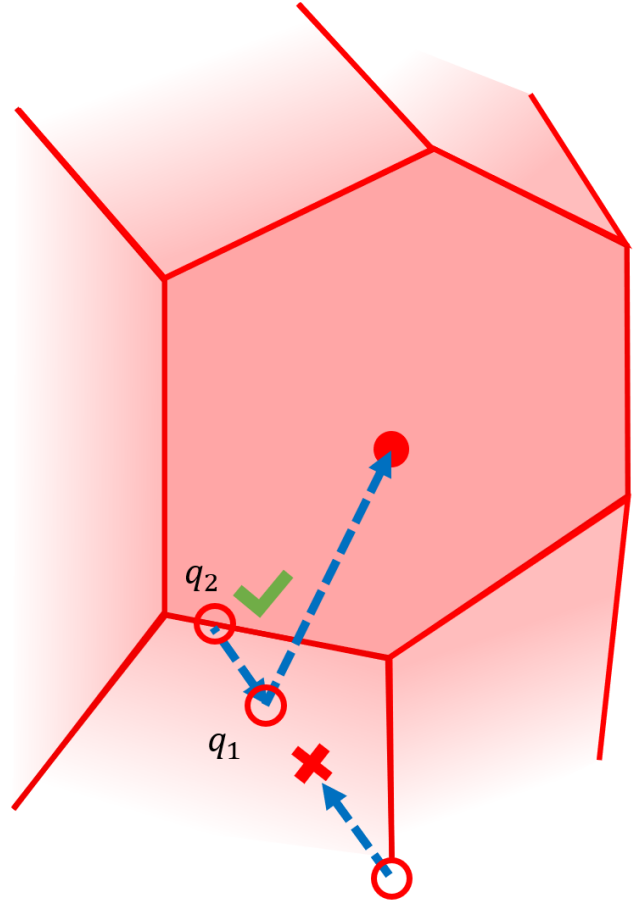

Figure 16: A visualization of the step in the synthesis method where the projected target in the AR is projected onto an AR facet. The check mark indicates a trajectory was able to succeed in reaching the projected target point from its projection on the lower dimensional face, whereas a cross indicates a failure.

Figure 17: A visualization of the above step in the synthesis method where the projected target on the facet is projected onto the next lowest dimensional AR face, which happens to be an AR edge here. The check mark indicates a success; a cross, a failure. For simplicity, only one tested trajectory direction is pictured here.

The fourth step begins by realizing that since the AR is convex, each of it's facets is convex since they lie completely on the boundary of the AR (Ziegler, 2012). Then, for the face in which a heat exchanger trajectory can begin at (call this point q_1) and reach the projected target point, find the vertices of the AR which lie on that face (i.e. the ones which satisfy the plane equation of the face) and find the convex hull of those points in $\mathbb{R}^{n-2}$. Repeat step 3, except this time for q_1 within the face instead of the projected target point within the AR. This will form q_2, as seen in Figure 17 below.

Steps 3 and 4 will occur k times until q_k lies on a 0-face (i.e. it is an AR vertex). Figure 18 below builds on the previous figure.

Now we note that, by Theorem (edges), a set edges of the AR connects q_k to the feed point. We want to begin

to find this set in the fifth step, and it is here where the simplex search becomes useful.
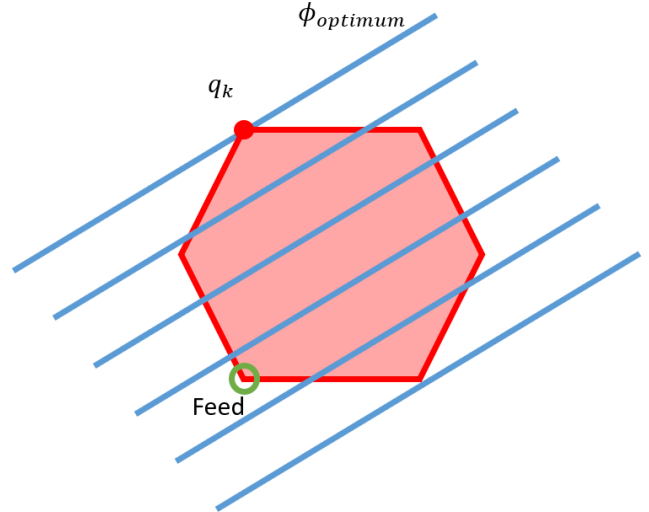
In the fifth step, define an objective function $\phi(x)$ such that ϕ and the H-representation of the AR form a linear optimization problem whose solution is q_k. Such a $\phi(x)$ can be formed, for example, by using a particle swarm optimization to find the equation of a $((n-2))$-dimensional plane which intersects the AR only at q_k, and creating a family of planes parallel to this which denote level sets of ϕ. Since the AR is convex, such a plane which intersects only q_k will be the furthest such level set one can reach while staying in the AR, and hence will be the solution to the optimization problem. Note that for all ϕ, finding the
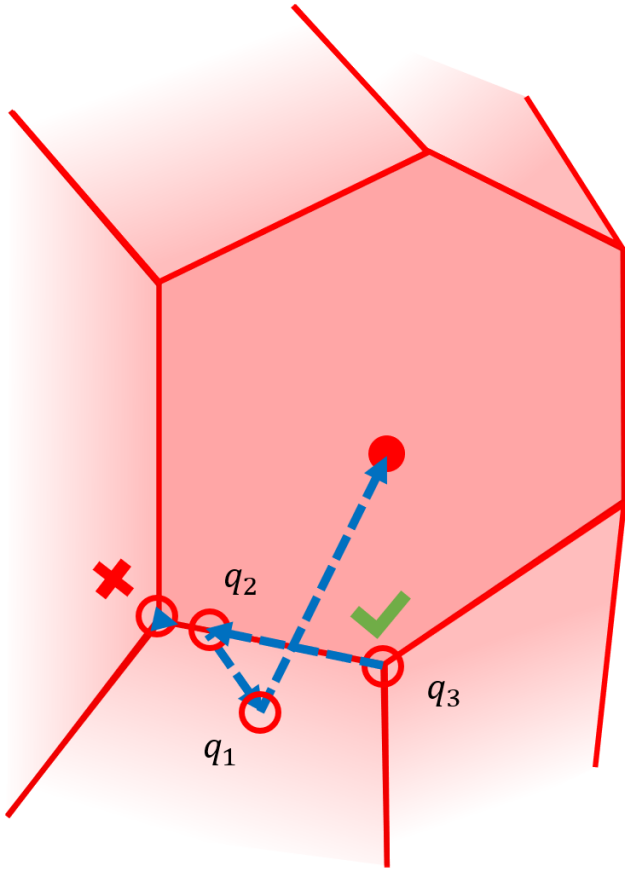
Figure 18: A visualization of the above step in the synthesis method where the projected target on the edge is projected onto an AR vertex. The check mark indicates a success; a cross, a failure.

Figure 19: The level sets (blue) of ϕ on the AR, with the feed and q_k.

H-representation and creating the optimization problem can be done in a $(n-1)$-dimensional projection of the AR without loss of information, since the AR is convex and $(n-1)$-dimensional. Figure 19 below shows what such a ϕ may look like for the $n=3$ case.

For the sixth step, run simplex search to solve this linear optimization problem starting at the feed point, making sure to log which trajectories were used during the run, to find the ordered set of trajectories which connects the feed to q_k. Since we found a set of trajectories which brings the feed state to q_k in steps five to six, and from q_k to the target point projection in steps one to four, a series of trajectories exists which brings the feed to the target point projection. The process-stream to process-stream part of the heat exchanger network is thus synthe-

sized. Since simplex search is a "continuously improving" optimization method, this part of the AR-based synthesis method has the benefit of always making non-negative progress towards synthesizing the heat exchanger network.

Note that if the AR is composed of maximal heat exchanger trajectories, then physically realizable heat exchangers will probably not be able to reach q_k since reaching q_k would require near-infinite area heat exchangers. This problem will be addressed in a later subsection. In the next subsection, we will discuss how to bring the target point projection within the AR to the target point outside, which will complete the HEN synthesis.

7.3. Navigating off the AR

To navigate from the projected target point in the AR to the target point is a fairly straightforward task. For a dimension which q_k and the target point have an unequal coordinate, use a utility whose temperature makes the target point intermediate between the utility temperature and the projected target point. In other words, if a stream needs to be heated, use a utility which is a higher temperature than the stream needs to be heated to. If a stream needs to be cooled, use a utility which is a lower temperature than the stream needs to be cooled to.

Finally, solve for the heat exchanger area required to bring the stream in question to the corresponding coordinate of the target point. This can be done using an iterative process on **counter-currentHX** (e.g. linear search) or by using the equivalent log-mean temperature difference equation below.

$$A = \frac{Q}{U\Delta T_{LM}} \qquad (16)$$

$$\Delta T_{LM} \equiv \frac{\Delta T_2 - \Delta T_1}{\ln\left(\frac{\Delta T_2}{\Delta T_1}\right)} \qquad (17)$$

$$\Delta T_2 \equiv T_{b_{out}} - T_{a_{out}}, \quad \Delta T_1 \equiv T_{b_{in}} - T_{a_{in}} \qquad (18)$$

where a and b are streams, *out* and *in* are the outlet and inlet states from/to the heat exchanger, respectively, and the rest of the symbols have the meaning already mentioned.

Repeating this process for every unequal coordinate in q_k will complete the process-utility stream part of the HEN. Often, it is only required for one coordinate since the AR is $(n-1)$-dimensional. Figure 20 below is a visualization of this process.
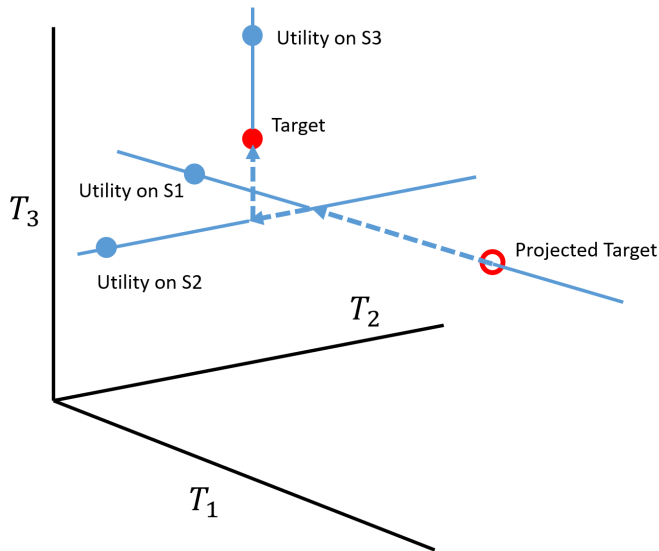

Figure 20: A visualization for the process of choosing appropriate utilities to navigate off the AR and onto the target point.

The HEN synthesis will then be complete, as a set of heat exchangers will have been found whose trajectories bring the feed point to the target point.

7.4. Synthesis with Additional Constraints

As discussed before, points on the boundary of the maximal-trajectory AR are impossible to reach with realistically sized heat exchangers. There is, however, a simple solution afforded by the AR method: simply generate the AR with those constraints in mind. For example, if there is a limit to heat exchanger size of 40 m^2, then generate the AR not with maximal heat exchanger trajectories, but with trajectories of maximum size 40 m^2. The same synthesis method can then be used on this AR. Figure 21 below illustrates this concept of a "sub-AR".
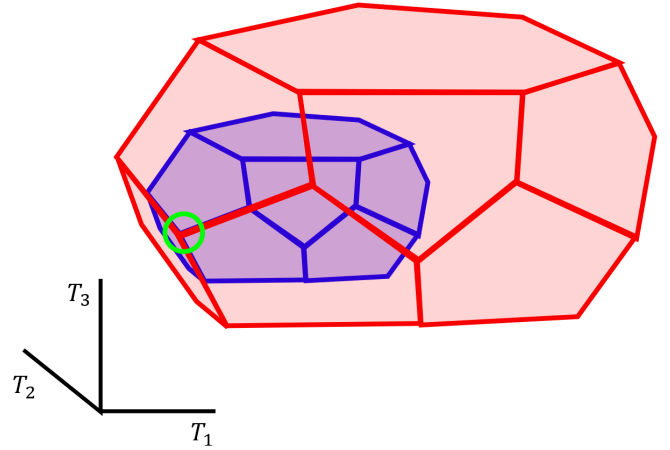
Figure 21: A visualization of the concept of a sub-AR (blue) within an AR (red) for a particular feed (centre of green ring).

It is assured that heat exchanger trajectories which connect vertices in this sub-AR will not exceed the maximum size constraint nor will they be physically impossible since it must lie within the maximal-trajectory AR. It must lie in the maximal-trajectory AR since one cannot exchange more heat with less heat exchange area, all else being equal. However, a check will have to be done on trajectories between any other points to ensure that they correspond to a feasible heat exchanger. For example, connecting two points which are close to the $T_1 = T_2$ line may require a larger heat exchanger than the constraint allows.

It is important to note that all the properties of the AR described in Sections 4 and 5 apply to the sub-AR as

well, since the AR properties only assumed that the heat exchanger trajectories are limited in some way. For the AR with maximal trajectories, they are limited by the second law of thermodynamics. For the sub-ARs, they are limited by the second law of thermodynamics, and something else (e.g. heat exchanger area).

Sub-ARs are also useful for allowing some steps to be skipped in the synthesis process. This is the case if the projected target point lies strictly within the AR. Then, a sub-AR can be found which has the projected target point on one of its faces, which allows the synthesis method to skip some steps where trajectories must be found which connect the target point projection to an AR vertex (i.e. we can skip the part of the method described by Figure 16).

This concludes the discussion about the AR-based heat exchanger network synthesis method. The upcoming section presents an example HENS problem and its solution using the techniques discussed in this article.

7.5. Synthesis example with four process streams

In this section, a complete worked example of using the AR method to synthesize a heat exchanger network for four process streams will be presented, starting from the HENS problem statement.

Consider the HENS problem where a process involving four streams of material labelled S1, S2, S3, and S4 have temperatures 95, 10, 50, and 10 °C and flow rates 95, 10, 50, and 50 kg s^{-1}, respectively. Their heat capacities are each a constant 4.186 kJ kg^{-1} °C^{-1}. The temperature target is for S1, S2, and S3 to have temperatures 60, 25, and 60 °C, respectively, and S4 can be at any temperature. The tolerance for the final temperature of the streams is ± 2°C. Available to us are heat exchangers with a constant heat transfer coefficient of 10 kJ s^{-1} m^{-2} and a maximum heat exchange area of 40 m^2. This concludes the HENS problem statement. In reality, such a problem may come about, for example, if S4 is a process fluid which is safer to

handle at elevated temperatures than the others, or if S4 is a utility fluid which is being reused from an upstream part of the process.
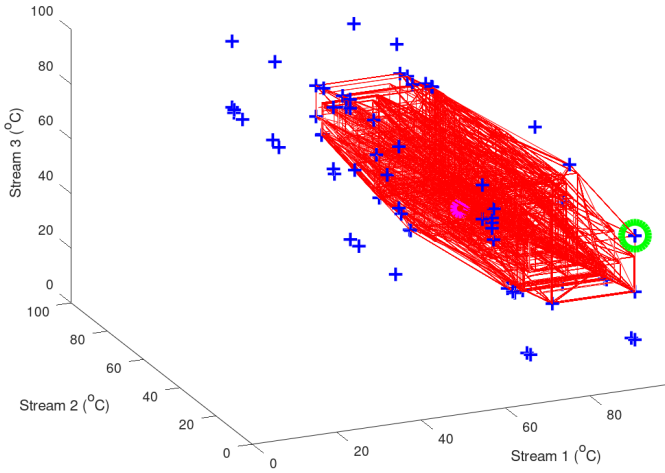
To start solving the HENS problem, we first use algorithm **HENARG** to generate the set of vertices composing the AR of the problem, noting that the AR will be a sub-AR since there is a constraint on the maximum heat exchanger size. Since only the vertices are needed to define the AR, we use two heat exchanger trajectory meshpoints to get the beginning and end of each trajectory. In this case, the inputs into **HENARG** are $feedpoint = (95, 10, 50, 50)$, $flowrates = (95, 10, 50, 50)$, $heatcaps = (4.186, 4.186, 4.186, 4.186)$, $U = 10$, $maxarea = 40$, and $trajmesh = 2$. Since the temperature of S4 is irrelevant, the AR can be represented by its projection onto the temperature space of S1, S2, and S3 without loss of pertinent information. The output of the HENARG onto this space, along with the feed and target points, is shown below in Figure 22.
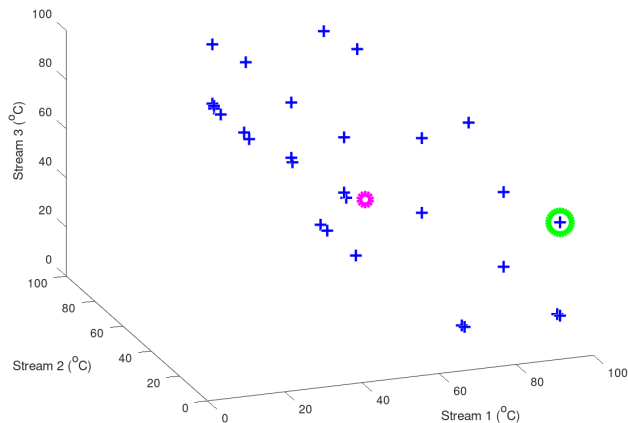
Figure 22: The output of HENARG with the feed point (centre of green ring) and target point (centre of magenta ring).

As can be seen, the target point lies within the AR, so no utilities will be necessary. The next part of the synthesis process is to find a path of heat exchanger trajectories which connects the feed to the target point. The first step of this is to connect a point on one of the faces of the

AR to the target point using a trajectory. Two methods have been presented to do this. The first is to use a heat exchanger trajectory to go from a face of the AR to the target point. The second method is to find a "sub-sub-AR" which lies within this (sub)AR and has the target point intersecting one of its faces. The second method will be used here. Using binary search on the parameter *maxarea* in **HENARG** with $0 < maxarea < 40$, the sub-sub-AR which has the target point intersecting one of its faces has edges composed of heat exchanger areas of 22 m^2, as seen in Figure 23.

Figure 24: The sub-sub-AR vertices, after filtering. Edge trajectories have also been added to clarify the shape of the AR.
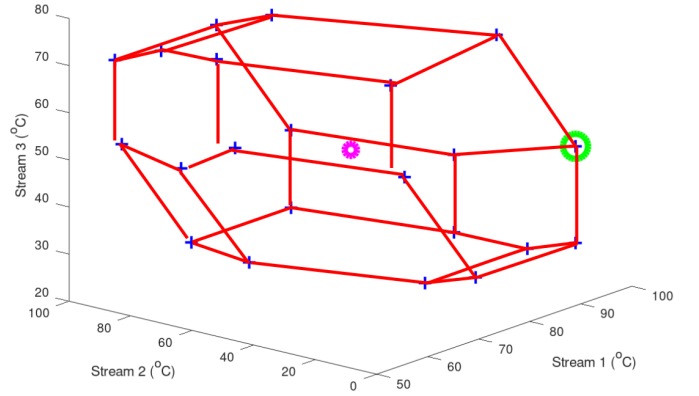

Figure 23: The sub-sub-AR inside the previously generated sub-AR. Notice the target point is visible on the surface of the smaller AR.

We will clean up the HENARG output of the sub-sub-AR for the sake of clarity in this example by picking random points and eliminating others which are a very small distance away (so we know they are not distinct vertices). This gives the result in Figure 24.

Putting this sub-sub-AR into its H-representation allows us to see that the target lies within tolerance to the plane corresponding to the inequality constraint $48/7x + y \leq 2950/7$, a 2-face. We now seek to find trajectories which connect the target point to lower dimensional faces until we reach a 0-face (an AR vertex).

We will start with finding a heat exchanger trajectory which connects a point on a 1-face to the target point on

the 2-face. Deducing from the directions of the edges of this face, we can see the current face corresponds to the family of trajectories from pairing S1 and S2 in a heat exchanger, or S3 and S4. Trying the direction corresponding to S3 and S4, we see that a trajectory starting from an edge on this face must start at either (60,25,76,52) or (60,25,56,67.7). A heat exchanger trajectory using S3 and S4 from (60,25,76,52) reaches the target point with a heat exchanger of size 42 m^2. However, since this is beyond the maximum heat exchanger size, we back off to size 40 m^2 which leaves S3 at 59 °C, which is within tolerance. Therefore, the next point is (60,25,76,52).

The next step is to connect the point on the 1-face to the point on the 0-face of the AR. Since we know a heat exchanger network between S3 and S4 was used previously, and the face is a 2-face, we know that in order to move in a different direction in the face we need the only other heat exchanger pair possible in this face, which is the one between S1 and S2. Trying the direction corresponding to S1 and S2, we see that a trajectory starting from a vertex on this face must start at either (60,10,76,52) or (54,58.7,76.3,52). Only a heat exchanger trajectory using S1 and S2 from (60,10,76,52) reaches the target point (actually, it reaches (58.5,25,76,52), within tolerance) with a heat exchanger of size 1.5 m^2. Therefore, the next point is (60,10,76,52). The point (60,10,76,52) is a 0-face of the

AR, so this part of the synthesis is complete; we have connected the target point in the AR to a vertex of the AR using heat exchanger trajectories.

The next step is to find an objective function in the temperature space to create an optimization problem whose constraints are the AR and whose solution is this vertex. To do this, we will use a particle swarm optimization to find a vector of fixed length whose tail is at this vertex, and whose tip points as far as possible from all vertices of the AR. This vertex denotes the normal of a 3-plane (a flat sheet in $\mathbb{R}^3$ which touches the AR only at this vertex). We will set this plane to denote a level set where only this vertex is the solution of an optimization problem, and define a family of parallel planes making up the other level sets of the objective function. The results of the particle swarm optimization to create this plane are seen below in Figure 25.
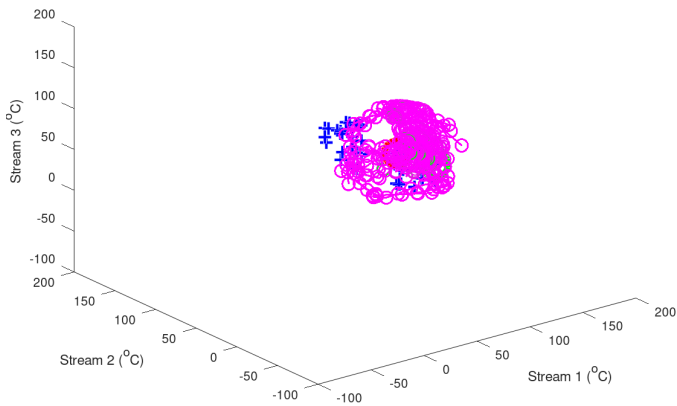
Figure 25: The paths of particles (magenta) throughout the particle swarm optimization. The particles roam the surface of a sphere of radius fifty whose centre is the desired AR vertex.

The solution of the particle swarm optimization is (47.6,-31.0,101.72) so the normal vector to the plane is (47.6,-31.0,101.72)-(60,10,76)= (-12.4,-41,25.72). The equation of the solution plane who has this normal vector and touches this vertex is therefore

$$-12.4T_1 - 41T_2 + 25.72T_3 = 800.72 \qquad (19)$$

To find out whether this function is maximized or minimized at the vertex, we substitute in another point, say the

feed, and look at the value on the right hand side. Since we know this vertex is the solution, if the right hand side is larger, then it is a minimization problem; if smaller, it is a maximization problem. Substituting in the feed point gives -302. We see $-302 < 800.7$. So, the objective function is

$$\max\left(\phi = -12.4T_1 - 41T_2 + 25.72T_3\right) \qquad (20)$$

The next step is to solve the linear optimization problem using simplex search. Since the AR has thirteen facets, there are thirteen slack variables and three state variables. All iterations of the simplex search can be seen below in Figure 26.
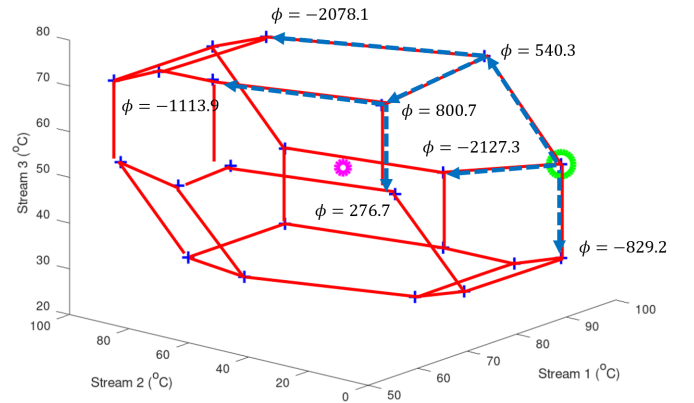

Figure 26: The results of the simplex search. On the first iteration, there will be three evaluations of the objective function: S3-S4=-829.2, S1-S3=540.3, S2-S3=-2127.3. S1-S3 will be the solution of this iteration. On the second iteration, there will be two evaluations of the objective function: S1-S4=800.7, S2-S4=-2078.1. S1-S4 will be the solution of this iteration. On the third iteration, there will be two evaluations of the objective function: S1-S2=-1113.9, S3-S4=276.7. Since there is no improving direction, the algorithm will stop.

Since a series of heat exchanger trajectories was found which brings the feed state to the (sub-sub-)AR vertex, and the vertex to the target point, this concludes the HEN synthesis.

Interpreting these trajectories appropriately gives the following heat exchanger network. First, S1 and S3 are put through a counter-current heat exchanger of area 22 m^2. Second, S1 and S4 are put through a counter-current heat exchanger of area 22 m^2. Third, S1 and S2 are put

through a counter-current heat exchanger of area 1.5 m^2. Lastly, S3 and S4 are put through a counter-current heat exchanger of area 40 m^2. The process flow diagram of this heat exchanger network is seen below in Figure 27.
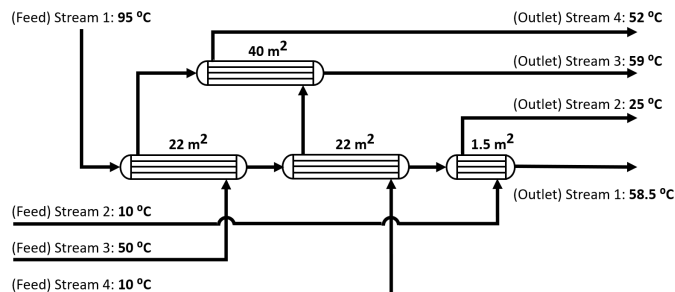
Figure 27: The final result of the AR synthesis method—a feasible solution to the HENS problem.

Note how the solution found by the synthesis method matches with intuition: because stream 1 has such a high temperature and heat capacity rate, all cooler streams are first brought in to lower its temperature, starting with the second warmest stream and ending with the coldest stream with the lowest flow rate. Then, the smaller amounts of heat transfer required among them are done.

7.5.1. Handling Uncertainties

One can also design heat exchanger networks taking uncertain system parameters into account using the AR method.

For example, suppose there is some uncertainty with the feed flow rates. Then, generating an AR for each confidence level of the flow rate can tell us whether any of them still contain the target point, and therefore whether the problem is still feasible or not for some confidence level. As another example, if there is some uncertainty in the target point temperature, then it can be checked whether an AR contains the range of the possible values that the target point can take.

Thus, synthesis with uncertain parameters is done in a similar fashion as for known parameter values, but for a family of ARs representing the uncertainty instead of just

a single AR. The check can be done especially quickly if the ARs have been converted into their H-representation, since containment can be checked by going through the finite list of inequality constraints. Again, we have the benefit that all this information can be obtained from the AR prior to carrying out any part of the synthesis step.

7.6. Improvements to the Method

There are opportunities for future theoretical and applied work which can improve the AR synthesis method. For example, finding a sufficient condition for a trajectory to be an edge of the AR would greatly reduce the computation time required to generate the AR from edges. Using Lemma 3 generates the AR edges, but also other trajectories, and trajectories starting at non-vertex points have no guarantee of landing on a vertex point or even staying on the boundary by this Lemma.

Opportunities for future applied work will be to extend the AR synthesis results to cover a larger class of HENS problems (for example, non-constant thermophysical stream properties), or incorporate them into a method to find cost-optimal HENs.

8. Conclusion

We have introduced a method for synthesizing feasible heat exchanger networks under uncertainty using attainable regions. The discussion has shown how an AR theory of heat exchanger networks can be used to synthesize feasible HENs within the class of designs which uses a combination of process-process and process-utility heat exchangers, under constant thermophysical properties, and no stream-splits. A region in the stream temperature space is created in which all HEN designs in this class must lie—the HEN-AR. From the properties of the HEN-AR proven in this article, a path can be constructed through the AR which connects the stream feed state to the target state. Due to a correspondence between the path and particular sizes and connections of heat exchangers, a feasible HEN is

derived. Particular strengths of this AR method lie in its ability to determine whether a HENS problem is feasible prior to carrying out any synthesis, its guarantee that any solution that it does give is physically realizable, and its ability to give solutions taking into account uncertainties in stream parameters.

References

Anton, H., Rorres, C., 2010. Elementary Linear Algebra: Applications Version. John Wiley & Sons. URL: `https://books.google.ca/books?id=1PJ-WHepeBsC`.

Bedenik, N.I., Ropotar, M., Kravanja, Z., 2007. Minlp synthesis of reactor networks in overall process schemes based on a concept of time-dependent economic regions. Computers & Chemical Engineering 31, 657 – 676. URL: `http://www.sciencedirect.com/science/article/pii/S0098135406002675`, doi:https://doi.org/10.1016/j.compchemeng.2006.10.007. eSCAPE-15.

Bergman, T., Lavine, A., Incropera, F., 2011. Fundamentals of Heat and Mass Transfer, 7th Edition. John Wiley & Sons, Incorporated. URL: `https://books.google.ca/books?id=5cgbAAAAQBAJ`.

Dantzig, G.B., 1990. A history of scientific computing, ACM, New York, NY, USA. chapter Origins of the Simplex Method, pp. 141–151. URL: `http://doi.acm.org/10.1145/87252.88081`, doi:10.1145/87252.88081.

Feinberg, M., 2000a. Optimal reactor design from a geometric viewpoint — iii. critical cfstrs. Chemical Engineering Science 55, 3553 – 3565. URL: `http://www.sciencedirect.com/science/article/pii/S0009250900000075`, doi:https://doi.org/10.1016/S0009-2509(00)00007-5.

Feinberg, M., 2000b. Optimal reactor design from a geometric viewpoint. part ii. critical sidestream reactors. Chemical Engineering Science 55, 2455 – 2479. URL: `http://www.sciencedirect.com/science/article/pii/S0009250999005114`, doi:https://doi.org/10.1016/S0009-2509(99)00511-4.

Feinberg, M., Hildebrandt, D., 1997. Optimal reactor design from a geometric viewpoint—i. universal properties of the attainable region. Chemical Engineering Science 52, 1637 – 1665. URL: `http://www.sciencedirect.com/science/article/pii/S000925099600471X`, doi:https://doi.org/10.1016/S0009-2509(96)00471-X.

Furman, K.C., Sahinidis, N.V., 2002. A critical review and annotated bibliography for heat exchanger network synthesis in the 20th century. Industrial & Engineering Chemistry Research 41, 2335–2370. URL: `https://doi.org/10.1021/ie010389e`, doi:10.1021/ie010389e, arXiv:https://doi.org/10.1021/ie010389e.

Glasser, D., Crowe, C., Hildebrandt, D., 1987. A geometric approach to steady flow reactors: the attainable region and optimization in concentration space. Industrial & Engineering Chemistry Research 26, 1803–1810. URL: `https://doi.org/10.1021/ie00069a014`, doi:10.1021/ie00069a014, arXiv:https://doi.org/10.1021/ie00069a014.

Hohmann, Edward Charles, J., 1971. Optimum networks for heat exchange. Doctor of philos-

ophy. University of Southern California. URL: `http://digitallibrary.usc.edu/cdm/ref/collection/p15799coll17/id/211425`.

Horn, F.J.M., 1964. Attainable and Non-attainable Regions in Chemical Reaction Technique, in: Chemical Reaction Engineering (Proceedings of the Third European Symposium held in Amsterdam, 15-17 September, 1964). 1 ed.. Pergamon Press, London, U.K.. chapter 6.4, pp. 293–302.

Kays, W., London, A., 1984. Compact Heat Exchangers. Krieger Publishing Company. URL: `https://books.google.ca/books?id=A08qAQAAMAAJ`.

Khumalo, N., Glasser, D., Hildebrandt, D., Hausberger, B., Kauchali, S., 2006. The application of the attainable region analysis to comminution. Chemical Engineering Science 61, 5969 – 5980. URL: `http://www.sciencedirect.com/science/article/pii/S0009250906003083`, doi:https://doi.org/10.1016/j.ces.2006.05.012.

Linnhoff, B., Hindmarsh, E., 1983. The pinch design method for heat exchanger networks. Chemical Engineering Science 38, 745 – 763. URL: `http://www.sciencedirect.com/science/article/pii/0009250983801857`, doi:https://doi.org/10.1016/0009-2509(83)80185-7.

Martín, M., Adams, T.A., 2019. Challenges and future directions for process and product synthesis and design. Computers & Chemical Engineering 128, 421 – 436. URL: `http://www.sciencedirect.com/science/article/pii/S0098135418312997`, doi:https://doi.org/10.1016/j.compchemeng.2019.06.022.

Ming, D., Glasser, D., Hildebrandt, D., Glasser, B., Metzger, M., 2016. Attainable region theory: an introduction to choosing an optimal reactor. John Wiley & Sons, Inc., Hoboken, NJ.

Muvhiiwa, R.F., Lu, X., Hildebrandt, D., Glasser, D., Matambo, T., 2018. Applying thermodynamics to digestion/gasification processes: the attainable region approach. Journal of Thermal Analysis and Calorimetry 131, 25–36. URL: `https://doi.org/10.1007/s10973-016-6063-9`, doi:10.1007/s10973-016-6063-9.

Nicol, W., Hernier, M., Hildebrant, D., Glasser, D., 2001. The attainable region and process synthesis: reaction systems with external cooling and heating: The effect of relative cost of reactor volume to heat exchange area on the optimal process layout. Chemical Engineering Science 56, 173 – 191. URL: `http://www.sciencedirect.com/science/article/pii/S0009250900001962`, doi:https://doi.org/10.1016/S0009-2509(00)00196-2.

Nicol, W., Hildebrandt, D., Glasser, D., 1997. Process synthesis for reaction systems with cooling via finding the attainable region. Computers & Chemical Engineering 21, S35 – S40. URL: `http://www.sciencedirect.com/science/article/pii/S0098135497874753`, doi:https://doi.org/10.1016/S0098-1354(97)87475-3. supplement to Computers and Chemical Engineering.

Pahor, B., Irsic, N., Kravanja, Z., 2000. Minlp synthesis and /211425. attainable region analysis of reactor networks in overall process schemes using more compact reactor superstructure. Computers & Chemical Engineering 24, 1403 – 1408. URL: `http://www.sciencedirect.com/science/article/pii/S0098135400003811`, doi:https://doi.org/10.1016/S0098-1354(00)00381-1.

Papoulias, S.A., Grossmann, I.E., 1983. A structural optimization approach in process synthesis—ii: Heat recovery networks. Computers & Chemical Engineering 7, 707 – 721. URL: `http://www.sciencedirect.com/science/article/pii/0098135483850236`, doi:https://doi.org/10.1016/0098-1354(83)85023-6.

Rardin, R.L., 2016. Optimization in Operations Research. 2 ed., Pearson.

Sanyal, R., Ziegler, G.M., 2010. Construction and analysis of projected deformed products. Discrete & Computational Geometry 43, 412–435. URL: `https://doi.org/10.1007/s00454-009-9146-6`, doi:10.1007/s00454-009-9146-6.

Tian, Y., Demirel, S.E., Hasan, M.F., Pistikopoulos, E.N., 2018. An overview of process systems engineering approaches for process intensification: State of the art. Chemical Engineering and Processing - Process Intensification 133, 160 – 210. URL: `http://www.sciencedirect.com/science/article/pii/S0255270118302782`, doi:https://doi.org/10.1016/j.cep.2018.07.014.

Yee, T., Grossmann, I., 1990. Simultaneous optimization models for heat integration—ii. heat exchanger network synthesis. Computers & Chemical Engineering 14, 1165 – 1184. URL: `http://www.sciencedirect.com/science/article/pii/0098135490850108`, doi:https://doi.org/10.1016/0098-1354(90)85010-8.

Ziegler, G., 2012. Lectures on Polytopes. Graduate Texts in Mathematics, Springer New York. URL: `https://books.google.ca/books?id=xd25TXSSUcgC`.