Optimization Approaches for Heat Integration and Distillation Network Synthesis

By

Lingxun Kong

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Chemical Engineering)

at the

UNIVERSITY OF WISCONSIN-MADISON

2019

Date of final oral examination: 07/25/2019

The dissertation is approved by the following members of the Final Oral Committee:

Christos T. Maravelias, Professor, Chemical Engineering

Regina M. Murphy, Professor, Chemical Engineering

Ross E. Swaney, Associate Professor, Chemical Engineering

James Luedtke, Associate Professor, Industrial and Systems Engineering

Abstract

We propose optimization-based approaches for heat integration and distillation network synthesis. For heat integration, we propose a mixed-integer nonlinear programming (MINLP) model for simultaneous process synthesis and utility targeting with general (i.e. classified and unclassified) streams and phase changes. To incorporate capital cost in the optimization, we propose an MINLP model based on the concept of composite-curve-based area targeting, which allows us to estimate both utility and heat exchanger area costs during simultaneous process synthesis and heat integration. Finally, we introduce a transshipment-based approach for simultaneous process synthesis and heat exchanger network synthesis. All three models are tailored to deal with variable stream temperature, flowrate, and hot/cold identity.

For distillation column design, we propose a McCabe-Thiele-inspired MINLP approach for both simple and complex columns. The approach utilizes piecewise linear functions to approximate complex VLE relationship and thus ensures tractability without jeopardizing accuracy. The piecewise linear approximating (PWLA) function is obtained by solving a mixed-integer linear programming (MILP) model which is developed separately. The proposed fitting model determines the optimal location of the break points, allowing us to (1) minimize fitting error given a fixed number of segments, or (2) find the minimum number of function segments for a given error tolerance.

Finally, we study a new challenging problem in distillation network synthesis: how to synthesize a distillation sequence when streams can have variable, including zero, component flows, which is common when the distillation network is simultaneously designed with the reactor network. To address this, we introduce a superstructure-based method for distillation sequence synthesis using

a “matrix” representation. Departing from existing methods, we consider more than one stream that couples the reactor and distillation networks and the proposed method determines the optimal destinations of these streams in the distillation network. We also generalize the Underwood method to account for zero-flows and unknown key components prior to optimization.

Acknowledgment

No words can express my gratitude to my advisor, Professor Christos Maravelias. He is a beacon, not only guiding me through my PhD journey, but also giving me lifelong directions. More than a mentor, he is a perfect role model. I believe that everything I learn from him will continue benefiting me for the rest of my life.

My sincere gratitude to other members of my thesis committee, Professors Regina Murphy, Ross Swaney, and James Luedtke, for their time reading my thesis, attending my defense, and more importantly, giving me valuable feedbacks. Many thanks to Professors James Rawlings, Alberto Del Pia, and Jose Caballero for their advice and comments.

Thanks to members of the Maravelias Group who kindly proofread my papers, answered my doubts, and offered me advice. Special thanks to my peers and best friends Ho Jae Lee and Xinyue Peng, who have been tremendously supportive and helpful for the past five years.

I also want to thank Ge for her patient and unyielding support. None of these would have happened without her understanding along the way. Finally, I would like to thank my parents and my family for their endless love and support, which put me through all the hard time in this journey. This thesis is dedicated to them.

Table of Content

List of Figures

List of Tables

Chapter 1
Introduction

In this chapter, we provide a critical review of optimization-based process synthesis. First, we discuss the state of the arts and the challenges of superstructure-based optimization methods. Then, we review approaches for heat integration and discuss the interactions between the heat exchanger network and the process (i.e. reactor and distillation networks). Then, we review literatures on distillation column design and distillation network synthesis. Finally, we present the outline of this thesis.

1.1 Superstructure-based Process Synthesis[1]

A superstructure refers to a process diagram that includes all potentially useful processing units and all relevant connections. By solving the superstructure optimization model, all process alternatives embedded are evaluated, and the best alternative is selected along with the optimal operation conditions for each unit.

A superstructure has the advantage of simultaneous consideration of complex interactions between all design decisions. However, a rich and complex superstructure is necessary to ensure the inclusion of the globally optimal solution. Moreover, in many cases, rigorous unit operation models are needed to obtain accurate results. Therefore, the resulting optimization model is generally a large-scale mixed-integer nonlinear programming (MINLP) problem. Most of the resulting MINLP problems are nonconvex after relaxing the integer variables (henceforth referred

[1] The contents of this section appear in Kong et al. *Comp. & Chem. Engr*. **2016**

to as nonconvex MINLP problems). Therefore, global optimization algorithms (solvers) are needed to solve these inherently difficult problems. The recent advances in computational algorithms and solvers allow us to solve small to medium scale nonconvex MINLP models. For large and/or highly nonconvex problems, problem specific solution methods are needed.

In terms of superstructure generation and modeling, early works focused on the combination of simple yet promising structures formulated in advance using engineering judgment [1] and the combination of superstructure subsystems (e.g. reaction network, separation network, and heat recovery network) each created independently [2-11]. Recently, Wu et al. [12] formalized the representation, generation, and modeling of superstructure-based process synthesis. Kong et al. introduced the flow-based formulation for modeling unit operations. In the flow-based formulation, the selection of unit operations in the superstructure is carried out via the (de)activation of inlet flows to the units. The unit models are constructed so that when the inlet flows are deactivated (i.e. equal to zero), the key output variables including outlet flows and unit cost are all equal to zero. In this way, if the inlet flows are deactivated, then the unit is completely deactivated. In this thesis, we adopt the flow-based formulation to model superstructure unit (de)activations.

1.2 Heat Integration and Heat Exchanger Network Synthesis[2]

The ever-increasing interest in renewable-energy-based processes has highlighted once again the importance of heat integration and heat exchanger network synthesis (HENS). Since the concept of pinch analysis was first proposed by Linnhoff and coworkers,[13] hundreds of papers have been published to address different aspects of heat exchanger network synthesis. This body of literature has been extensively reviewed.[14-16]

[2] The contents of this section appear in Kong and Maravelias, *I&ECR*. **2018**

Two types of approaches are available to solve a HENS problem: sequential and simultaneous methods. Sequential approaches involve decomposing the HENS into subproblems that are solved successively. In general, these subproblems can be defined as: (1) finding the minimum utility consumption (utility target) with a given heat recovery approach temperature (HRAT), (2) estimating the minimum number of exchanger units (or minimum matches between hot and cold streams) based on the utility target, and (3) obtaining the heat exchanger network (HEN) with minimum capital cost.

One classic approach to solve HENS problem is pinch analysis.[13,17,18] In this approach, the original problem is partitioned into temperature intervals and the minimum utility usage along with the energy recovery bottlenecks (i.e., pinch points) are obtained through composite curves. Based on the pinch locations, HEN is decomposed into subnetworks that are synthesized using design guidelines and heuristic rules.

Alternatively, these decomposed subproblems can be formulated into mathematical models and solved sequentially. Part of the results from one sub-problem are used as input parameters in the next. Specifically, the utility target subproblem can be formulated as a linear programming (LP) problem.[19-23] Other mixed-integer formulations were later proposed to allow additional structural constraints.[24] Using the utility target, a mixed integer linear programming problem (MILP) can be solved to obtain the minimum number of exchangers, or the minimum number of matches between hot and cold streams.[24-27] Finally, a nonlinear programming (NLP) model is solved based on the previous targets to obtain a HEN with minimum capital (area) cost.[28] Several approaches have been proposed to deal with multiple targets in one step: Floudas et al.[28] introduced a match-network hyperstructure model that minimizes both exchanger unit and area cost; Colberg and Morari[29] developed a transshipment-based NLP model for simultaneous utility and heat exchanger area

targeting; Jezowski et al.[30] proposed a transportation-based LP model that targets both utility and area.

While these sequential HEN design methods are simple and effective, they have two shortcomings. First, they do not consider all the tradeoffs (e.g., energy vs. capital) within the HEN. Since the decisions made in the previous steps are used as parameters/constraints in the later steps, the solutions are, in general, globally suboptimal. Second, their scope is limited: given are a set of hot streams and set of cold streams with their inlet and outlet temperatures and heat capacity flow rates. In other words, these approaches can only be applied when the process flowsheet has already been synthesized and optimized. However, it has been shown that the simultaneous optimization of process and heat exchanger network can lead to overall economic benefits.[31]

One line of research attempts to address the first challenge by enabling simultaneous HENS. Instead of decomposing the HEN design into subproblems, these approaches try to find the optimal network in one step by solving, in general, mixed-integer nonlinear programming (MINLP) models. Yuan et al.[32] proposed an MINLP formulation based on a superstructure without stream splitting; Floudas and Ciric[33] extended the hyperstructure model to simultaneously determine the HEN with minimum annualized cost; Yee and Grossmann[34] developed another MINLP model that is based on a stagewise superstructure representation for simultaneous HENS. The work by Yee and Grossmann[34] has later been extended to consider additional features in HENS and for HEN retrofit.[35-44] Alternatively, transportation-based MILP approaches have been proposed for one-step HENS.[45,46] More recently, Hong et al.[47] developed a transshipment-based MINLP model for simultaneous HENS.

Another line of research focuses on integrating process synthesis with heat integration to address the second challenge. Specifically, the heat integration module of these approaches determines the

utility usage while allowing process stream inlet/outlet temperatures and flow rates to vary so that it can be simultaneously solved with a process model. Probably the most famous simultaneous process synthesis and utility targeting approach is the one by Duran and Grossman,[48] which is an NLP model based on the so call "pinch location method". This approach has been employed over the past few decades as the heat integration module in subsequent works.[49-54] Some MINLP-based models were also proposed for simultaneous process synthesis and utility targeting: Grossmann et al.[55] proposed a disjunctive model for isothermal and non-isothermal streams; Navarro-Amorós et al.[56] and Zhang et al.[57] proposed transshipment-based models; Quirante et al.[58] proposed a disjunctive reformulation to the pinch location method.

1.3 Distillation Network Synthesis and Modeling[3]

The separation (distillation) network is a critical subsystem in process synthesis that has been extensively studied. In a chemical plant, a separation network is usually needed to separate the products, byproducts, reactants and other components in the reactor effluents. When the mixture(s) to be separated contains more than two components, a sequence of distillation columns is needed. It has been shown that optimizing the distillation sequence leads to energy and capital savings.

Superstructure-based approaches have been widely used for distillation network synthesis (distillation sequencing). Ideally, a distillation superstructure should contain all the alternative configurations to be considered. Numerous methods have been proposed for the modeling of distillation superstructures, including work on generating all possible distillation configurations. One approach is to use an algorithmic procedure to generate a superstructure that includes all the

[3] The contents of this section appear in Kong and Maravelias, *AIChE J.*, accepted, and Kong and Maravelias, *Comp. & Chem. Engr*., submitted.

feasible configurations [59], followed by solving a nonlinear optimization problem for each configuration to minimize an objective of interest (e.g. the total vapor duty). Finally, the best configuration is obtained by comparing the performance of these different configurations. Instead of enumerating and evaluating all the feasible designs, another line of works solve MINLP models not only to generate all feasible configurations, but also to identify the configuration with minimum vapor duty in one step [60,61].

In any distillation sequencing models, a sub-model is needed to describe each distillation unit. In general, there are three types of approaches to model distillation units. First, shortcut methods are frequently used to predict the key design parameters such as minimum reflux ratio, minimum number of trays, and minimum energy demand. Classical short-cut approaches include the Fenske's equation [62] for the minimum number of trays, the Underwood's method [63] for calculating the minimum reflux ratio, and Gilliland's [64] and Smoker's [65] equations for predicting the number of trays. Extending and generalizing the Underwood method, new shortcut procedures were later developed for complex columns [66,67] and batch distillation [68]. Shortcut methods were also proposed to obtain the minimum energy demand for non-ideal systems and more complex distillation systems such as reactive distillation and thermally coupled distillation. Second, rigorous methods are based on material and energy balances and equilibrium calculation for every tray. The enthalpy and vapor-liquid equilibrium (VLE) are described by equations of state (EOS) or thermodynamic models such as UNIQUAC [69] and UNIFAC [70]. For example, the mixed-integer nonlinear programming model by Viswanathan and Grossmann [71] is applicable to predict optimal locations of the feeds and the number of trays required for a distillation column with multiple feeds. Third, graphical methods are usually derived from material and energy balances and are based on a graphical representation of the vapor-liquid-equilibrium relationship. Early methods were

proposed by Rodebus [72], Ponchon and Savarit [73,74], and McCabe and Thiele [75]. The method by McCabe and Thiele is probably the most well-known graphical design method, which is frequently used for conceptual design and analysis of the distillation of binary mixtures. Extensions of the McCabe-Thiele method were later proposed to handle non-ideal mixtures [76-78], complex distillation columns [79] (e.g., columns with multiple feed streams) , and multicomponent distillation [80].

1.4 Thesis Outline

The remainder of this thesis is structured as follows. In Chapters 2, 3 and 4, we present three optimization-based heat integration approaches. They are tailored to address various problems arise from simultaneous process synthesis and heat integration. In Chapter 5, we introduce an approach to fit discrete data points with continuous piecewise linear approximating functions. While the topic in Chapter 5 is not directly related to chemical engineering, the approach facilities the effective solution of chemical engineering problems. In Chapter 6, we present a novel optimization-based approach for distillation column design. In Chapters 7 and 8, we present approaches for distillation network synthesis, which are tailored for simultaneous reactor and distillation network synthesis. Finally, in Chapter 9, we conclude and discuss future directions. We use bold upper case for sets and subsets, upper case for variables, and lower case for parameters.

Chapter 2

Simultaneous Process Synthesis and Heat Integration with Unclassified Hot/Cold Process Streams[4]

2.1 Motivation

One basic assumption in all previous approaches reviewed in Chapter 1 is that we are given a set of hot streams to be cooled and a set of cold streams to be heated. However, in many cases, especially in the context of superstructure optimization, some process streams cannot be classified a priori as hot or cold streams. To illustrate, consider the outlet stream from one of two alternative reactors (A or B) which is sent to one of the two alternative separation units (C or D) for purification, where units A, B, C and D operate at different temperatures. As shown in Figure 2-1, we cannot determine if the stream that connects the reactor and the separator is a hot or cold stream before solving the model.

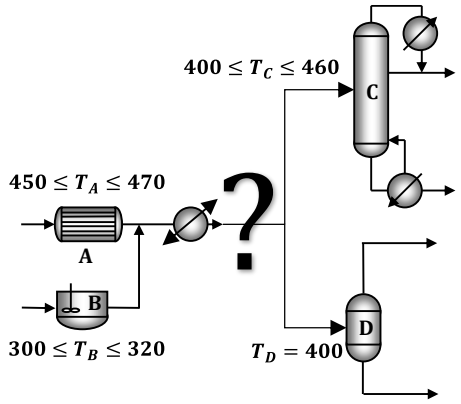


Figure 2-1. An unclassified hot/cold process stream in a process superstructure.

[4] The contents of this chapter appear in Kong et al. *Comp. & Chem. Engr.* **2017**

Accordingly, in this chapter we present a general heat integration model that not only handles variable stream temperatures and flow rates, but also handles unclassified hot/cold streams.

2.2 Heat Integration Model

The proposed heat integration model is based upon the linear programing (LP) transshipment model proposed by Papoulias and Grossmann [81]. The LP transshipment model utilizes the concept of temperature intervals to predict the minimum utility consumption for a process with fixed stream conditions. Under the assumption of constant heat capacity of process streams, the temperature intervals are constructed from the known inlet temperatures of the streams and utilities. Hot streams and utilities are treated as source nodes, while cold streams and utilities are considered sink nodes. Heat is transferred like a "commodity" between sources and sinks via temperature intervals, which can be seen as intermediate "warehouses". Any residue heat is cascaded down to the next interval at a lower temperature. Heat integration using this approach maximizes heat exchange among streams, while satisfies the second law of thermodynamics and minimum approach temperature as a heat exchanger design requirement. As a result of the fixed stream temperatures and flow rates, the only variables in this LP model are the residue heat at each interval and the heat duties of hot and cold utilities.

To account for variable stream temperatures and flow rates, Navarro-Amorós et al. [56] extended the LP transshipment model and proposed a disjunctive model. In their work, "dynamic" temperature intervals are constructed through an implicit ordering of stream inlet temperatures. Extending and generalizing this concept, we propose a model that not only allows variable stream conditions, but also handles streams that cannot be classified as hot or cold a priori.

Given a process stream with fixed composition, the heat capacity can be assumed constant if it does not vary significantly with temperature. When the temperature dependence has to be taken into account, the stream can be divided into sub-streams, each with constant heat capacity, so that a piecewise linear approximation of the temperature-enthalpy (T-H) relation is obtained. If a stream composition is not fixed, then the stream or sub-stream heat capacity can be defined as a function (variable) of composition. Thus, all cases of variable heat capacities can be addressed, albeit at the expense of computational performance.

2.2.1 Classification of Hot/Cold Streams

The hot/cold identity of an unclassified process stream can be determined by comparing the stream inlet (TIN_s) and outlet ($TOUT_s$) temperatures; e.g., if a stream has a higher temperature at its inlet than at its outlet, it is a hot stream. The comparison of temperatures is carried out using a pair of classification binary variables, W_s^H and W_s^C, as shown in Eqs. (2.1) – (2.4). If process stream $s \in \mathbf{S}^P$ is a hot stream, $W_s^H = 1$ and $W_s^C = 0$, while if it is a cold stream, $W_s^C = 1$ and $W_s^H = 0$:

$$W_s^H + W_s^C = 1 \qquad s \in \mathbf{S^P} \qquad (2.1)$$

$$TIN_s - TOUT_s = T_s^+ - T_s^- \qquad s \in \mathbf{S^P} \qquad (2.2)$$

$$T_s^+ \leq \alpha_s^U W_s^H \qquad s \in \mathbf{S^P} \qquad (2.3)$$

$$T_s^- \leq \alpha_s^U W_s^C \qquad s \in \mathbf{S^P} \qquad (2.4)$$

where α_s^U are valid upper-bounds on the absolute differences between stream inlet and outlet temperatures. Eq. (2.1) ensures that each stream is classified either as hot or cold. Non-negative variables T_s^+ and T_s^- are used in Eq. (2.2) to model the temperature difference between the inlet and outlet of each process stream, and they are (de)activated by the corresponding classification binaries in Eqs. (2.3) and (2.4).

Next, variables (x_s) related to an unclassified stream are disaggregated into two counter parts, x_s^H and x_s^C. If the stream is classified to be a hot stream, then we enforce that $x_s^H = x_s$ and $x_s^C = 0$, while if it is a cold stream we enforce that $x_s^C = x_s$ and $x_s^H = 0$. In this way we can formulate constraints for the "hot" and "cold" parts separately [82,83].

$$\begin{bmatrix} W_s^H \\ \neg W_s^C \\ x_s = x_s^H \end{bmatrix} \underline{\vee} \begin{bmatrix} \neg W_s^H \\ W_s^C \\ x_s = x_s^C \end{bmatrix} \qquad s \in \mathbf{S}^P \qquad (2.5)$$

The linear reformulation of the above disjunctions will be discussed in the next subsections. For consistency, throughout the chapter, superscript H/C is used to denote disaggregated variables for hot/cold streams.

2.2.2 Dynamic Interval Construction

In the proposed model, temperature intervals cannot be directly constructed since the stream inlet (and outlet) temperatures are not known prior to solving the model. To address this, we construct "dynamic" intervals in which the variable stream inlet temperatures are ordered from high to low. In general, the minimum number of interval boundaries (i.e. number of intervals plus one) required should be the same as the number of streams including process streams and utilities. Here we use set $\mathbf{K}$ to denote interval boundaries and $\mathbf{K}^I = \mathbf{K}\backslash\{0\}$ to denote temperature intervals. We introduce non-negative variables T_k to denote the temperature at the interval boundaries, ordered as follows:

$$T_k \leq T_{k-1} \qquad k \in \mathbf{K}^I \qquad (2.6)$$

Since interval boundaries are defined by stream inlet temperatures, each boundary temperature, T_k, corresponds to one distinct inlet temperature (TIN_s).

This one-to-one matching can be done using a pair of binary variables, $X_{s,k}^H$ and $X_{s,k}^C$, which are equal to 1 only when TIN_s is equal to T_k. We first enforce that each stream inlet temperature must be assigned to one interval boundary:

$$\sum_k X_{s,k}^H = W_s^H \qquad s \in \mathbf{S}^P \qquad (2.7H)$$

$$\sum_k X_{s,k}^C = W_s^C \qquad s \in \mathbf{S}^P \qquad (2.7C)$$

Depending on the hot/cold identity of each process stream, we allow only one $X_{s,k}^H$ or one $X_{s,k}^C$ to be equal to 1 at any interval. The above constraints connect $X_{s,k}^H$ and $X_{s,k}^C$ to the classification binaries so that the disjunction in Eq. (2.5) is enforced.

Then, through Eq. (2.8) it is ensured that only one stream has its inlet temperature assigned to each boundary.

$$\sum_{s\in\mathbf{S}}(X_{s,k}^H + X_{s,k}^C) = 1 \qquad k \in \mathbf{K} \qquad (2.8)$$

Finally, we disaggregate the inlet temperatures into hot/cold parts as well as into interval boundaries (i.e. $TIN_{s,k}^H$ and $TIN_{s,k}^C$) via Eq. (2.9).

$$TIN_s = \sum_{k\in\mathbf{K}}(TIN_{s,k}^H + TIN_{s,k}^C) \qquad s \in \mathbf{S} \qquad (2.9)$$

These disaggregated inlet temperatures should be equal to the boundary temperatures only when the corresponding $X_{s,k}^H$ or $X_{s,k}^C$ is equal to one, while they are forced to zero otherwise,

$$TIN_{s,k}^H \le \beta_s^U X_{s,k}^H \qquad s \in \mathbf{S},\ k \in \mathbf{K} \qquad (2.10H)$$

$$TIN_{s,k}^H \le T_k \qquad s \in \mathbf{S},\ k \in \mathbf{K} \qquad (2.11H)$$

$$TIN_{s,k}^H \ge T_k - \beta_s^U\left(1 - X_{s,k}^H\right) \qquad s \in \mathbf{S},\ k \in \mathbf{K} \qquad (2.12H)$$

$$TIN_{s,k}^C \le \beta_s^U X_{s,k}^C \qquad s \in \mathbf{S},\ k \in \mathbf{K} \qquad (2.10C)$$

$$TIN_{s,k}^{C} \leq T_k - \delta \qquad s \in \mathbf{S},\ k \in \mathbf{K} \qquad (2.11C)$$

$$TIN_{s,k}^{C} \geq (T_k - \delta) - (\beta_s^U - \delta)\left(1 - X_{s,k}^{C}\right) \qquad s \in \mathbf{S},\ k \in \mathbf{K} \qquad (2.12C)$$

where β_s^U are valid upper bounds on the stream inlet temperatures. If a process stream is hot ($W_s^H = 1$, $W_s^C = 0$), then Eq. (2.7H) enforces exactly one $X_{s,k}^H$ to be equal to one, while Eq. (2.7C) forces all the $X_{s,k}^C$ to zero. Disaggregated variables $TIN_{s,k}^H$ and $TIN_{s,k}^C$ are (de)activated as follows: Eq. (2.10H) forces $TIN_{s,k}^H$ to zero when the corresponding $X_{s,k}^H$ is equal to zero; Eqs. (2.11H) and (2.12H) enforce that $TIN_{s,k}^H = T_k$ only when $X_{s,k}^H = 1$; Eq. (2.10C) enforces that all $TIN_{s,k}^C$ to be equal to zero; and Eqs. (2.11C) and (2.12C) are not active. Finally, the non-zero $TIN_{s,k}^H$ connects the stream inlet temperature (TIN_s) and the boundary temperature (T_k) via Eq. (2.9). Note that all the cold part of the disaggregated inlet temperatures are shifted by the heat recovery approach temperature (δ) as a heat exchanger design requirement.

2.2.3 Outlet Temperature Assignment

While the stream outlet temperatures ($TOUT_s$) are not used to construct the temperature intervals, they must be correctly positioned with respect to the interval boundaries in order to calculate heat duties at each interval. Therefore, we introduce another pair of binary variables ($Y_{s,k}^H / Y_{s,k}^C$), which are equal to one if the outlet temperature is within interval k.

First, we enforce that each stream outlet temperature is located at exactly one interval:

$$\sum_{k \in \mathbf{K^I}} Y_{s,k}^H = W_s^H \qquad s \in \mathbf{S}^P \qquad (2.13H)$$

$$\sum_{k \in \mathbf{K^I}} Y_{s,k}^C = W_s^C \qquad s \in \mathbf{S}^P \qquad (2.13C)$$

Then, similar to the previous section, the stream outlet temperatures are disaggregated into $TOUT_{s,k}^H / TOUT_{s,k}^C$, according to the corresponding $Y_{s,k}^H / Y_{s,k}^C$. If $Y_{s,k}^H = 1$, then $TOUT_{s,k}^H \in$

$[T_k, T_{k-1}]$; if $Y_{s,k}^C = 1$, then $TOUT_{s,k}^C \in [T_k - \delta, T_{k-1} - \delta]$; while if $Y_{s,k}^H / Y_{s,k}^C$ are zero, the corresponding $TOUT_{s,k}^H / TOUT_{s,k}^C$ must be zero,

$$TOUT_s = \sum_{k \in \mathbf{K}^I} (TOUT_{s,k}^H + TOUT_{s,k}^C) \qquad s \in \mathbf{S}^P \qquad (2.14)$$

$$TOUT_{s,k}^H \leq \gamma_s^U Y_{s,k}^H \qquad s \in \mathbf{S}^P, k \in \mathbf{K}^I \qquad (2.15H)$$

$$TOUT_{s,k}^H \leq T_{k-1} \qquad s \in \mathbf{S}^P, k \in \mathbf{K}^I \qquad (2.16H)$$

$$TOUT_{s,k}^H \geq T_k - \gamma_s^U \left(1 - Y_{s,k}^H\right) \qquad s \in \mathbf{S}^P, k \in \mathbf{K}^I \qquad (2.17H)$$

$$TOUT_{s,k}^C \leq \gamma_s^U Y_{s,k}^C \qquad s \in \mathbf{S}^P, k \in \mathbf{K}^I \qquad (2.15C)$$

$$TOUT_{s,k}^C \leq T_{k-1} - \delta \qquad s \in \mathbf{S}^P, k \in \mathbf{K}^I \qquad (2.16C)$$

$$TOUT_{s,k}^C \geq (T_k - \delta) - (\gamma_s^U - \delta)\left(1 - Y_{s,k}^C\right) \qquad s \in \mathbf{S}^P, k \in \mathbf{K}^I \qquad (2.17C)$$

where γ_s^U are valid upper bounds on the stream outlet temperatures. Again, Eq. (2.14) disaggregates the outlet temperatures, and Eqs. (2.15H) and (2.15C) force the disaggregated temperatures to zero when the corresponding binaries are equal to zero. The main difference compared to the "inlet" counterpart is that equality is not enforced here when $Y_{s,k}^H / Y_{s,k}^C$ is equal to one (see Eqs. (2.16H) – (2.17H) and (2.16C) – (2.17C)). Figure 2-2 provides an example of the assignment of stream inlet and outlet temperatures to temperature intervals.

2.2.4 Heat Duty Calculations

In order to calculate the heat duty of a given stream s at interval k, we have to consider the following three possible cases:

(1) the stream completely spans (i.e. goes through) interval k,

(2) the stream partially spans interval k, and

(3) the stream does not span any portion of interval k.

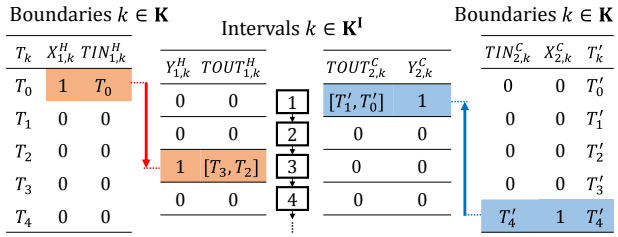


Figure 2-2. Temperature assignment of a hot stream and a cold stream to temperature intervals.

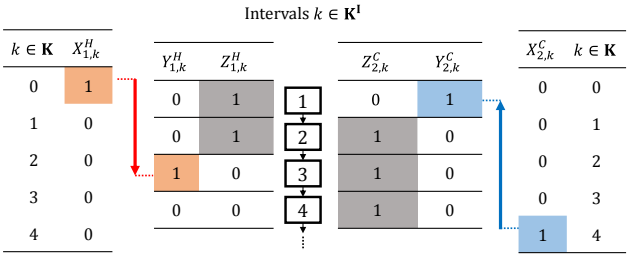


Figure 2-3. Values of binary variables *X*, *Y*, and *Z*.

Case (2) is true when $Y^H_{s,k}/Y^C_{s,k}$ are equal to one, whereas to represent cases (1) and (3) we need additional binary variables. We introduce binary variables $Z^H_{s,k}/Z^C_{s,k}$, which are equal to one when stream s goes through interval k (i.e. case (1) is true). The constraints that relate $Z^H_{s,k}/Z^C_{s,k}$ with binary variables X and Y are as follows,

$$Z^H_{s,k} = Z^H_{s,k-1} + X^H_{s,k-1} - Y^H_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.18\text{H})$$

$$Z^C_{s,k-1} = Z^C_{s,k} + X^C_{s,k-1} - Y^C_{s,k-1} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.18\text{C})$$

They ensure that if a stream starts at boundary $k' \in \mathbf{K}$ and ends somewhere within interval $k'' \in \mathbf{K}^I$, then it must have gone through all the intervals from $k' + 1$ to $k'' - 1$ for hot stream and from $k'' + 1$ to k' for cold stream. The logic is illustrated in Figure 2-3.

Thus, when $Z^H_{s,k}/Z^C_{s,k}$ are equal to one, case (1) is true, while when neither $Y^H_{s,k}/Y^C_{s,k}$ nor $Z^H_{s,k}/Z^C_{s,k}$ are equal to one, case (3) is true. The heat duty of a stream at interval k is disaggregated into $Q^H_{s,k}$

and $Q^C_{s,k}$, and $Q^H_{s,k}/Q^C_{s,k}$ are further disaggregated into two mutually exclusive counterparts: $Q1^H_{s,k}/Q1^C_{s,k}$ representing heat duties in case (1), and $Q2^H_{s,k}/Q2^C_{s,k}$ representing those in case (2). Note that the heat duties in case (3) should always be zero and are therefore not considered in the following constraints:

$$Q^H_{s,k} = Q1^H_{s,k} + Q2^H_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.19\text{H})$$

$$Q1^H_{s,k} \le \theta 1^U_s Z^H_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.20\text{H})$$

$$Q2^H_{s,k} \le \theta 2^U_s Y^H_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.21\text{H})$$

$$Q^C_{s,k} = Q1^C_{s,k} + Q2^C_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.19\text{C})$$

$$Q1^C_{s,k} \le \theta 1^U_s Z^C_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.20\text{C})$$

$$Q2^C_{s,k} \le \theta 2^U_s Y^C_{s,k} \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.21\text{C})$$

For illustration, let us assume we are given a hot stream s at interval k. Eqs. (2.19C) – (2.21C) ensure that $Q^C_{s,k}$, $Q1^C_{s,k}$, and $Q2^C_{s,k}$ are zero when s is a hot stream. If the stream goes through interval k (case (1)), Eqs. (2.19H) – (2.21H) ensure that $Q^H_{s,k} = Q1^H_{s,k}$; if it partially spans the interval (case (2)), $Q^H_{s,k} = Q2^H_{s,k}$ is enforced; and if it does not span any portion of the interval (case (3)), we obtain $Q^H_{s,k} = 0$.

The next step is to relate the heat duties to the temperature differences and heat capacity flows (F_s). For simplicity, we first assume that s is a hot stream ($Q^C_{s,k} = Q1^C_{s,k} = Q2^C_{s,k} = 0$). The heat duties in case (1), $Q1^H_{s,k}$, can be determined as follows,

$$Q1^H_{s,k} + Q1^{H,D}_{s,k} = F_s(T_{k-1} - T_k) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.22\text{H})$$

$$Q1^{H,D}_{s,k} \le \theta 1^U_s (1 - Z^H_{s,k}) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.23\text{H})$$

where $Q1_{s,k}^{H,D}$ is a nonnegative slack variable that takes the value of the right-hand-side (RHS) of Eq. (2.22H) if case (1) is not true (i.e. $Z_{s,k}^{H} = 0$). If case (1) is true, $Q1_{s,k}^{H}$ is equal to the heat capacity flow multiplied by the temperature difference between the interval boundaries.

Similarly, the heat duty in case (2), $Q2_{s,k}^{H}$, is calculated as follows,

$$Q2_{s,k}^{H} + Q2_{s,k}^{H,D} = F_s\left(T_{k-1} - TOUT_{s,k}^{H}\right) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.24H)$$

$$Q2_{s,k}^{H,D} \leq \theta 2_s^U (1 - Y_{s,k}^{H}) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.25H)$$

where $Q2_{s,k}^{H,D}$ is a nonnegative slack variable that is equal to the RHS of Eq. (2.24H) if case (2) is not true (i.e. $Y_{s,k}^{H} = 0$). When case (2) is true, $Q2_{s,k}^{H,D}$ is equal to zero and the heat duty is determined from the stream heat capacity flow and the difference between the outlet temperature and boundary temperature.

Similar constraints are introduced for the cold streams:

$$Q1_{s,k}^{C} + Q1_{s,k}^{C,D} = F_s(T_{k-1} - T_k) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.22C)$$

$$Q1_{s,k}^{C,D} \leq \theta 1_s^U (1 - Z_{s,k}^{C}) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.23C)$$

$$Q2_{s,k}^{C} + Q2_{s,k}^{C,D} - Q2_{s,k}^{C,P} = F_s\left(TOUT_{s,k}^{C} - T_k + \delta\right) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.24C)$$

$$Q2_{s,k}^{C,D} \leq \theta 2_s^U (1 - Y_{s,k}^{C}) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.25C)$$

$$Q2_{s,k}^{C,P} \leq \theta 2_s^U (1 - Y_{s,k}^{C}) \qquad s \in \mathbf{S}^P,\ k \in \mathbf{K}^I \qquad (2.26C)$$

where $Q1_{s,k}^{C,D}$ and $Q2_{s,k}^{C,D}$ are the cold counter part of $Q1_{s,k}^{H,D}$ and $Q2_{s,k}^{H,D}$, respectively. The reason we need a pair of slack variables ($Q2_{s,k}^{C,D}$ and $Q2_{s,k}^{C,P}$) for cold streams is that the RHS of Eq. (2.24C)

can be either positive or negative depending on the sign of $\left(TOUT_{s,k}^{C} - T_k + \delta\right)$. The heat duty calculation is further illustrated using an example in Figure 2-4.

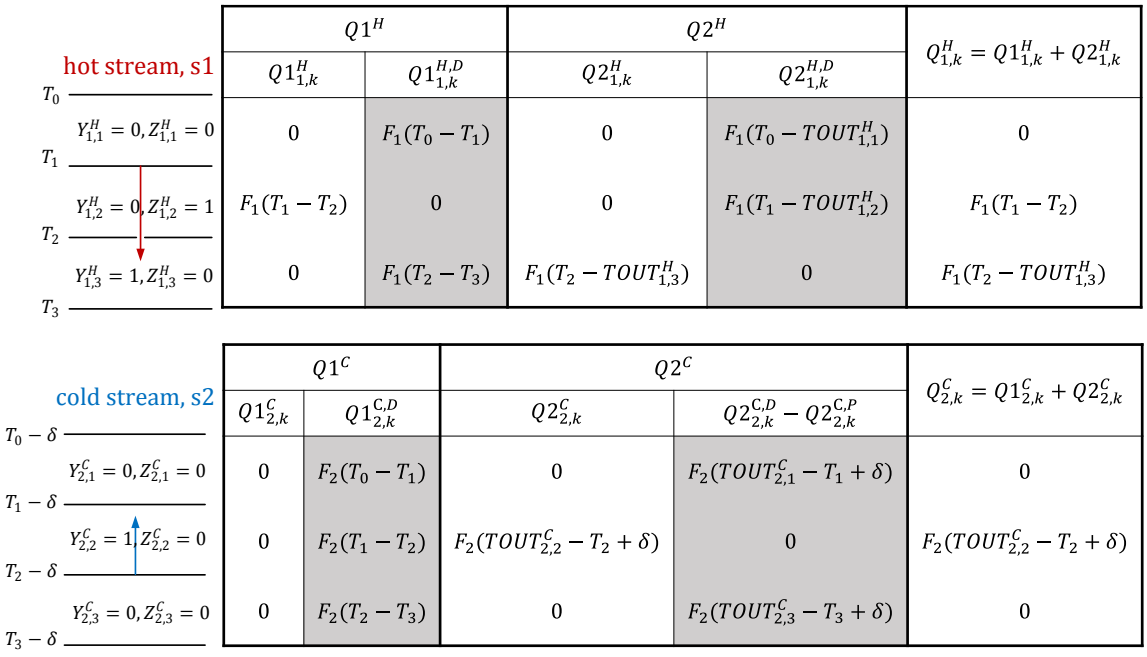
For now, we assume a single isothermal hot utility at the highest temperature interval, and a single cold utility that enters and exits at the lowest interval. Therefore, constraints in Section 2.2.4 are only written for the process streams ($s \in \mathbf{S}^P$). In Section 2.3, we will extend our discussion on how to calculate heat duties if multiple utility streams are present.

2.2.5 Generalized Transshipment Model

Using $Q_{s,k}^{H}$ and $Q_{s,k}^{C}$ from the previous section, we write the heat flow balance at interval k:

$$R_k - R_{k-1} = \sum_{s\in\mathbf{S}}(Q_{s,k}^{H} - Q_{s,k}^{C}) \qquad k \in \mathbf{K}^I \tag{2.27}$$

where R_{k-1} and R_k are residual heats entering and exiting interval k, respectively. We enforce $R_0 = R_{|\mathbf{K}|} = 0$ so that no residue heat is added to the highest temperature interval or received from lowest temperature interval.



$Q1^H$		$Q2^H$		$Q_{1,k}^H = Q1_{1,k}^H + Q2_{1,k}^H$
$Q1_{1,k}^H$	$Q1_{1,k}^{H,D}$	$Q2_{1,k}^H$	$Q2_{1,k}^{H,D}$	
0	$F_1(T_0 - T_1)$	0	$F_1(T_0 - TOUT_{1,1}^H)$	0
$F_1(T_1 - T_2)$	0	0	$F_1(T_1 - TOUT_{1,2}^H)$	$F_1(T_1 - T_2)$
0	$F_1(T_2 - T_3)$	$F_1(T_2 - TOUT_{1,3}^H)$	0	$F_1(T_2 - TOUT_{1,3}^H)$

$Q1^C$		$Q2^C$		$Q_{2,k}^C = Q1_{2,k}^C + Q2_{2,k}^C$
$Q1_{2,k}^C$	$Q1_{2,k}^{C,D}$	$Q2_{2,k}^C$	$Q2_{2,k}^{C,D} - Q2_{2,k}^{C,P}$	
0	$F_2(T_0 - T_1)$	0	$F_2(TOUT_{2,1}^C - T_1 + \delta)$	0
0	$F_2(T_1 - T_2)$	$F_2(TOUT_{2,2}^C - T_2 + \delta)$	0	$F_2(TOUT_{2,2}^C - T_2 + \delta)$
0	$F_2(T_2 - T_3)$	0	$F_2(TOUT_{2,3}^C - T_3 + \delta)$	0

Figure 2-4. Heat duty calculations in the heat cascade for a hot stream (s1) and a cold stream (s2).

For minimum utility targeting, the objective function can be written as follows,

$$\min \sum_{s\in \mathbf{S}^{HU}} \sum_{k\in \mathbf{K}^I} \mu_s Q_{s,k}^H + \sum_{s\in \mathbf{S}^{CU}} \sum_{k\in \mathbf{K}^I} \mu_s Q_{s,k}^C \tag{2.28}$$

where μ_s is the unit cost of each utility stream. This expression can account for multiple utilities.

If we consider the entire chemical facility, the objective function must be modified to include terms representing the capital and operating cost of the process,

$$\min \sum_{s\in \mathbf{S}^{HU}} \sum_{k\in \mathbf{K}^I} \mu_s Q_{s,k}^H + \sum_{s\in \mathbf{S}^{CU}} \sum_{k\in \mathbf{K}^I} \mu_s Q_{s,k}^C + f(\Psi) \tag{2.29}$$

where Ψ is a vector of variables such as flow rates, temperatures, pressures and equipment sizes, and $f(\Psi)$ is a cost function that relates these variables to the annualized costs. This objective is used in the examples presented in Section 2.7.

2.3 Extensions

In this section, we discuss how the model can be modified to handle multiple utilities and isothermal streams.

2.3.1 Multiple Utilities

If the utility is an isothermal stream (e.g. medium pressure steam), it exchanges heat with other streams at a constant temperature, which means that it defines an interval boundary on which it exchanges heat with other streams. Therefore, the isothermal utility usage is (de)activated by binary variables $X_{s,k}^H / X_{s,k}^C$,

$$Q_{s,k}^H \leq \theta_s^U X_{s,k}^H \qquad s \in \mathbf{S}^{IHU},\ k \in \mathbf{K}^I \tag{2.30H}$$

$$Q_{s,k}^C \leq \theta_s^U X_{s,k}^C \qquad s \in \mathbf{S}^{ICU},\ k \in \mathbf{K}^I \tag{2.30C}$$

where $Q^C_{s,k} = 0$ for isothermal hot utilities ($s \in \mathbf{S}^{IHU}$) and $Q^H_{s,k} = 0$ for isothermal cold utilities ($s \in \mathbf{S}^{ICU}$).

If the utility is a non-isothermal stream (e.g. hot water) that could span multiple temperature intervals, then, in the general case, is treated as a hot/cold non-isothermal process stream. In other words, we need binary variables X, Y, and Z, and constraints (2.7) – (2.26C) to determine the heat duty of the non-isothermal utility at each interval. Since utility streams can always be defined as hot or cold streams and their inlet and outlet temperatures are usually given, we can use preprocessing to reduce the search space and improve the solution process (see details in Section 2.5).

2.3.2 Isothermal Streams

So far we have assumed that all process streams have different inlet and outlet temperatures, i.e., we have only considered sensible heat. If we are given an isothermal stream ($s \in \mathbf{S}^{IP}$), which exchanges heat through evaporation or condensation, we assume that the exchange takes place at the boundary it defines. Thus, similar to the case of isothermal utilities, the amount of heat exchanged is constrained via,

$$Q^H_{s,k} \leq \theta^U_s X^H_{s,k} \qquad s \in \mathbf{S}^{IP},\ k \in \mathbf{K}^I \qquad (2.31\text{H})$$

$$Q^C_{s,k} \leq \theta^U_s X^C_{s,k} \qquad s \in \mathbf{S}^{IP},\ k \in \mathbf{K}^I \qquad (2.31\text{C})$$

where the disaggregated heat duties, $Q^H_{s,k}/Q^C_{s,k}$, are related to stream mass flow rate ($\widehat{F}_s$) and latent heat (λ_s),

$$\widehat{F}_s \lambda_s = \sum_k (Q^H_{s,k} + Q^C_{s,k}) \qquad s \in \mathbf{S}^{IP} \qquad (2.32)$$

If the hot/cold classification of the isothermal stream is unknown, we include a dummy temperature difference between the stream inlet and outlet in order to classify the stream:

$$TIN_s - TOUT_s = W_s^H - W_s^C \qquad s \in \mathbf{S}^{IP} \qquad (2.33)$$

which can be seen as a special case of Eq. (2.2). Note that the temperature difference is only for the classification purpose and does not affect the heat duty calculation.

2.4 Phase Changes

In general, a process stream might undergo phase changes and therefore a single stream could involve up to three regions: liquid, vapor, and two-phase [39,42,52]. If a stream undergoes phase changes, we can no longer assume constant heat capacity. A more accurate yet tractable approach is to assume constant heat capacity only within each region [52]. In other words, the T-H diagram is represented by line segments (see Figure 2-5). The process stream with phase changes will then be divided into sub-streams to represent the regions it spans so that each sub-stream has constant heat capacity. Note that if we assume that a process stream with phase changes contains only a pure substance and the stream pressure is fixed, then the phase changes should take place at a constant temperature, and therefore the sub-stream corresponding to the two-phase region will be an isothermal stream.

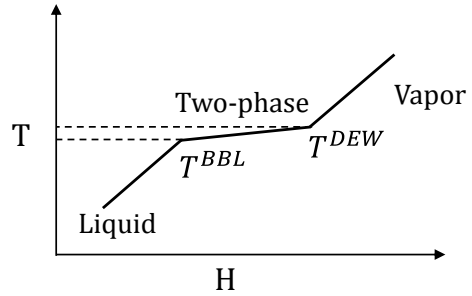


Figure 2-5. An example of piecewise linear representation of the T-H diagrams.

One of the major difficulties in modeling process streams with phase changes in the context of variable stream conditions is the detection of phase changes. Since the stream inlet and outlet temperatures are allowed to vary, the phases that a stream spans are unknown before solving the model. Moreover, the dew point and bubble point of a given stream are related to the stream composition and pressure, and they are therefore decision variables, which further increases the difficulty in determining what phases the stream spans. To address this issue, we propose a new disjunctive model for phase detection. Kamath et. al. [52] proposed a disjunctive programming reformulation resulting into a nonlinear programming (NLP) model. Our reformulation results into mixed integer linear constraints, and it is formulated so that it can be integrated with the heat integration model proposed in this chapter.

Given is a stream that potentially undergoes phase changes denoted as the "parent" stream; regardless of the phases it eventually spans, we first divide it into up to three sub-streams, LQ, VP, and 2P, to represent the liquid, vapor, and two-phase regions, respectively. Since a given stream can start and end at one of the three regions, there are a total of nine possible cases (see Table 1 and Figure 2-6).

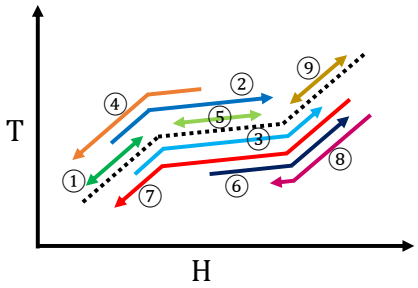


Figure 2-6. A graphic representation of the nine cases (numbered in circles) from Table 2-1.

Table 2-1. Nine possible cases for a process stream s with potential phase changes.

Case	Start	Finish	$TIN_s \in$	$TOUT_s \in$	$TIN_{s'}$			$TOUT_{s'}$		
					LQ	2P	VP	LQ	2P	VP
1	LQ	LQ	$[0, T^{BBL}]$	$[0, T^{BBL}]$	TIN	0	0	$TOUT$	0	0
2	LQ	2P	$[0, T^{BBL}]$	$[T^{BBL}, T^{DEW}]$	TIN	T^{BBL}	0	T^{BBL}	$TOUT$	0
3	LQ	VP	$[0, T^{BBL}]$	$[T^{DEW}, \gamma^U]$	TIN	T^{BBL}	T^{DEW}	T^{BBL}	T^{DEW}	$TOUT$
4	2P	LQ	$[T^{BBL}, T^{DEW}]$	$[0, T^{BBL}]$	T^{BBL}	TIN	0	$TOUT$	T^{BBL}	0
5	2P	2P	$[T^{BBL}, T^{DEW}]$	$[T^{BBL}, T^{DEW}]$	0	TIN	0	0	$TOUT$	0
6	2P	VP	$[T^{BBL}, T^{DEW}]$	$[T^{DEW}, \gamma^U]$	0	TIN	T^{DEW}	0	T^{DEW}	$TOUT$
7	VP	LQ	$[T^{DEW}, \beta^U]$	$[0, T^{BBL}]$	T^{BBL}	T^{DEW}	TIN	$TOUT$	T^{BBL}	T^{DEW}
8	VP	2P	$[T^{DEW}, \beta^U]$	$[T^{BBL}, T^{DEW}]$	0	T^{DEW}	TIN	0	$TOUT$	T^{DEW}
9	VP	VP	$[T^{DEW}, \beta^U]$	$[T^{DEW}, \gamma^U]$	0	0	TIN	0	0	$TOUT$

Depending on the regions where the parent stream starts and ends (see the 2nd and 3rd columns in Table 2-1), which can be inferred by comparing the parent stream inlet/outlet temperatures with the dew point and bubble point (see the 4th and 5th columns), we enforce the sub-stream inlet and outlet temperatures as follows:

If the parent stream starts/ends in a particular region, the inlet/outlet temperature of the sub-stream corresponding to this region will be equal to the parent stream inlet/outlet temperature, respectively;

If the parent stream crosses the boundary of a region, the inlet and/or outlet temperature of the corresponding sub-stream will be set to T^{BBL} or T^{DEW}; and

If the parent stream does not exist at a region, both inlet and outlet temperatures of that sub-stream will be set equal to zero.

To illustrate, we consider case 2 in which the stream starts as a liquid and ends in the two-phase region. In this case, the parent stream inlet temperature is no greater than the bubble point ($TIN_s \leq T^{BBL}$) while the outlet temperature must be in between the bubble point and dew point ($T^{BBL} \leq TOUT_s \leq T^{DEW}$). We assign the parent stream inlet and outlet temperatures to the liquid sub-stream inlet ($TIN_s = TIN_{LQ}$) and two-phase sub-stream outlet temperatures ($TOUT_s = TOUT_{2P}$), respectively. Further, temperatures of both the liquid sub-stream outlet and two-phase sub-stream

inlet are set to the bubble point ($TIN_{2P} = TOUT_{LQ} = T^{BBL}$). Finally, we force inlet and outlet temperatures of the irrelevant vapor sub-stream to zero ($TIN_{VP} = TOUT_{VP} = 0$). This approach ensures that all the relevant sub-stream temperatures are correctly positioned with respect to the dew point and bubble point, and the heat loads used in the heat integration model are properly calculated using the approach discussed in Section 2.2.4.

The complete mixed-integer formulation of the logic conditions shown in Table 2-1 can be found in Kong et al [84]. Here, to illustrate the basic ideas, we show how to relate the parent stream inlet/outlet temperature to those of the sub-streams.

Given a subset of process streams that potentially undergo phase changes ($s \in \mathbf{S}^{PC}$), we first identify the regions that each stream can go through and define a set of sub-streams ($s' \in \mathbf{S}_s^{SUB}$) only for these regions. Then, we introduce binary variables $U_{s'}$ and $V_{s'}$, defined over $s' \in \mathbf{S}_s^{SUB}$, to represent the starting and ending regions, respectively. Since a given parent stream (s) can only start and end at one of its sub-streams, the following constraints are enforced:

$$\sum_{s' \in \mathbf{S}_s^{SUB}} U_{s'} = 1 \qquad s \in \mathbf{S}^{PC} \qquad (2.34)$$

$$\sum_{s' \in \mathbf{S}_s^{SUB}} V_{s'} = 1 \qquad s \in \mathbf{S}^{PC} \qquad (2.35)$$

Next we disaggregate the inlet and outlet temperatures of the parent streams into sub-streams (see Eqs. (2.36) and (2.37)). Note that $TIN_{s'}^{P}$ and $TOUT_{s'}^{P}$ are only used here to relate the parent stream and sub-stream inlet and outlet temperatures, and they are different from the sub-stream inlet and outlet temperatures from Table 2-1. Here Eqs. (2.38) and (2.39) ensure that $TIN_{s'}^{P}$ and $TOUT_{s'}^{P}$ are equal to zero when the corresponding binary variables are zero, and Eqs. (2.40) – (2.43) disaggregate the sub-stream inlet and outlet temperatures into hot/cold counter-parts and into

temperature intervals if and only if the corresponding binaries are equal to 1. Thus, the parent stream temperatures are only assigned to the correct sub-streams.

$$TIN_s = \sum_{s' \in \mathbf{S}_s^{\mathbf{SUB}}} TIN_{s'}^P \qquad s \in \mathbf{S}^{PC} \qquad (2.36)$$

$$TOUT_s = \sum_{s' \in \mathbf{S}_s^{\mathbf{SUB}}} TOUT_{s'}^P \qquad s \in \mathbf{S}^{PC} \qquad (2.37)$$

$$TIN_{s'}^P \le \beta_{s'}^U U_{s'} \qquad s \in \mathbf{S}^{PC}, s' \in \mathbf{S}_s^{SUB} \qquad (2.38)$$

$$TOUT_{s'}^P \le \gamma_{s'}^U V_{s'} \qquad s \in \mathbf{S}^{PC}, s' \in \mathbf{S}_s^{SUB} \qquad (2.39)$$

$$TIN_{s'}^P \le \sum_k \left(TIN_{s',k}^C + TIN_{s',k}^H\right) + \beta_{s'}^U (1 - U_{s'}) \qquad s \in \mathbf{S}^{PC}, s' \in \mathbf{S}_s^{SUB} \qquad (2.40)$$

$$TIN_{s'}^P \ge \sum_k \left(TIN_{s',k}^C + TIN_{s',k}^H\right) - \beta_{s'}^U (1 - U_{s'}) \qquad s \in \mathbf{S}^{PC}, s' \in \mathbf{S}_s^{SUB} \qquad (2.41)$$

$$TOUT_{s'}^P \le \sum_k \left(TOUT_{s',k}^C + TOUT_{s',k}^H\right) + \gamma_{s'}^U (1 - V_{s'}) \qquad s \in \mathbf{S}^{PC}, s' \in \mathbf{S}_s^{SUB} \qquad (2.42)$$

$$TOUT_{s'}^P \ge \sum_k \left(TOUT_{s',k}^C + TOUT_{s',k}^H\right) - \gamma_{s'}^U (1 - V_{s'}) \qquad s \in \mathbf{S}^{PC}, s' \in \mathbf{S}_s^{SUB} \qquad (2.43)$$

It is important to note that although sub-streams are originated from their parent streams, they are treated as independent streams in the heat integration model. When counting the required intervals, we should consider the number of sub-streams instead of the number of parent streams. Each sub-stream has its own hot/cold identity, which is consistent with that of its parent stream, through temperature relationships. Therefore, the extension to phase change can also handle unclassified process streams and is compatible with any aforementioned extensions.

2.5 Integration with Process Synthesis

While the unit operations can be modeled using various methods [85-88], in this work we adopt a "flow based" formulation [87,89]. First, a unit selection binary variable (Y_i^{UNIT}) is introduced for each

alternative unit ($i \in \mathbf{I}$) so that $Y_i^{UNIT} = 1$ if the unit is selected. If the unit is not selected ($Y_i^{UNIT} = 0$), we force the inlet component flows of that unit to zero. Note that according to the approach of Kong et al. [90], if the unit inlet component flows are equal to zero, not only the unit outlet flows are forced to zero by material balances, the heat duty and cost of that unit are also forced to zero.

In the process model, we define a set of streams ($s \in \mathbb{S}$) that connect the inlet and outlet between two processing units or supply/remove materials to/from the process. The set of process streams used in the heat integration model ($\mathbf{S}^P$) consists of a subset of $\mathbb{S}$ which potentially requires heating/cooling, and a set of streams representing the heat duties of processing units. The process model is linked to the heat integration model via linking variables: TIN_s, $TOUT_s$, and F_s for all $s \in \mathbf{S}^P$ (see Figure 2-7). They are simultaneously determined and optimized by the process and the heat integration. Stream temperatures, TIN_s and $TOUT_s$, are related to the temperatures of the units that it connects (see Figure 2-8). If a subset of streams potentially undergoes phase changes, the extension to phase changes ensures that the inlet and outlet temperatures of each sub-stream are correctly calculated when entering the heat integration model.

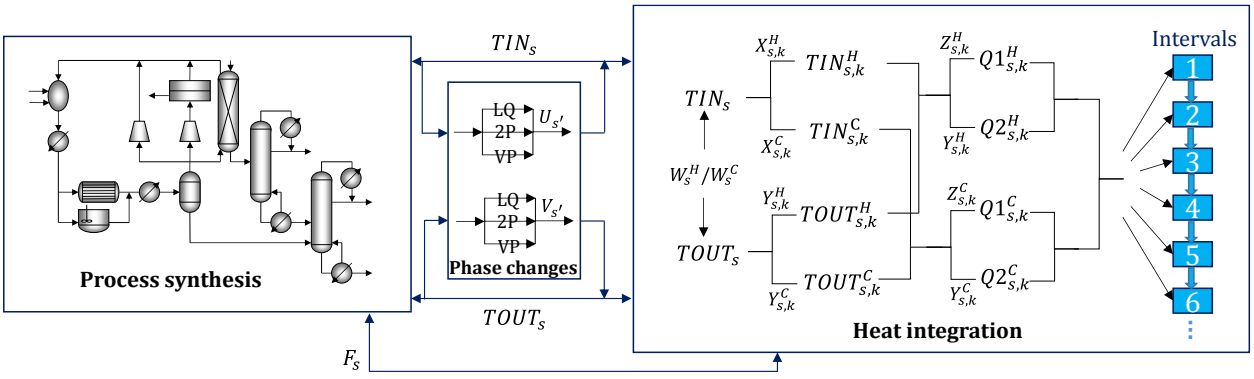


Figure 2-7. Integrating process synthesis and heat integration.

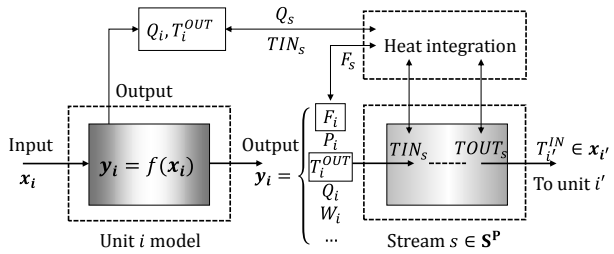


Figure 2-8. Relationship between stream temperatures and unit models.

Finally, to reduce the cardinality of $\mathbf{S}^P$ and thus the number of temperature intervals, which will in turn lead to smaller models, we carefully define the subset of process streams. The idea is illustrated in Figure 2-9, where the original superstructure representation with $\mathbf{S}^P = \{2, 3, 4, 5\}$ is replaced with an alternative representation with $\mathbf{S}^P = \{1, 5\}$. Note that the two representation have the same feasible solutions embedded on them.

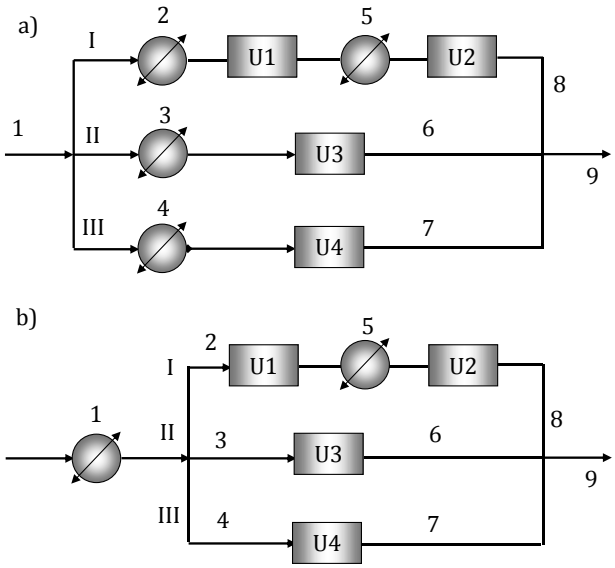


Figure 2-9. Superstructure example with four units (U1 – U4) and nine streams (1 – 9).

2.6 Illustrative Example

Four main components (A, B, C, and D) are involved in the process. Components A and B are used as raw materials to produce component C in the first reaction (RXN 1), which can be carried

out in two alternative isothermal continuous stir-tank reactors (CSTR1 and CSTR2) employing different catalysts. The effluent stream of the reactor is sent to a flash tank to separate unreacted A and B from intermediate C. Unless stated otherwise, the separations in this example are assumed sharp (i.e., all the light components exit at the top while all the heavy components come out from the bottom). Unreacted A and B are collected at the top and recycled, while the bottom of the flash contains pure C, which is sent to another isothermal continuous stir-tank reactor (CSTR3) to produce final product D. This second reaction (RXN 2) is assumed to be an equilibrium reaction, and the equilibrium constant is a function of temperature. The reaction is assumed endothermic and therefore the reactor requires heating. Finally, unreacted C is separated from D in one of three alternative separation technologies before recycled back to CSTR3. Each unit has a cost which is approximated by a power-law function of the total inlet flow. It is assumed that the feed stream (stream 1) flow rates are 2 kmol/s of A and 1 kmol/s of B, and we are selling final product D at a price of $1700/ton. The objective is to maximize profit, which takes into account the revenue, cost of materials, unit capital cost, and utility cost.

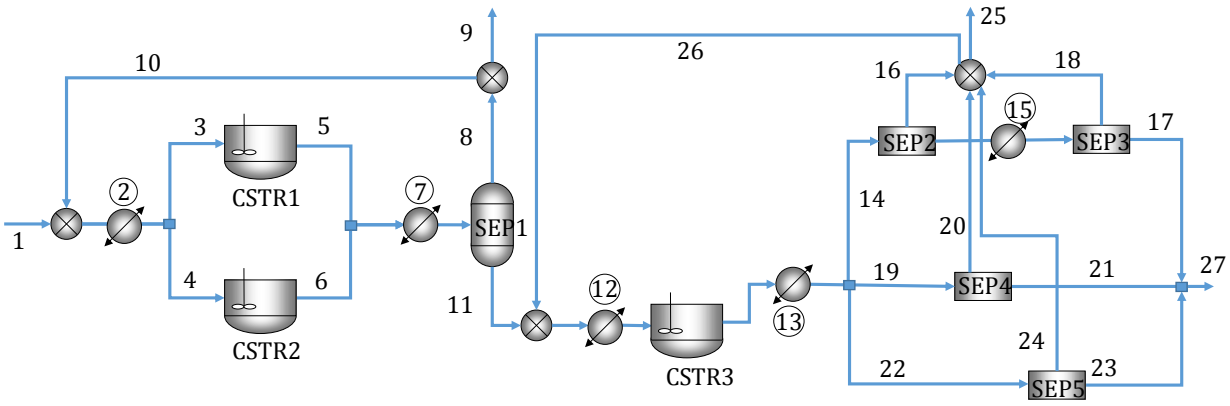


Figure 2-10. A superstructure chemical process. Streams in the chemical process are numbered.

Table 2-2. Unit specifications for superstructure example

Reactors	RXN	Temperature (K)	Conversion	Unit cost pre-factor, κ (mol^{-0.6}yr^{-0.4}$)
CSTR 1	$A + B \rightarrow C$	500	0.9	0.9
CSTR 2	$A + B \rightarrow C$	400	0.8	0.85
CSTR 3	$C \rightleftharpoons D$	330 – 400	variable	1
Separators	Top/Bottom		-	
SEP 1	AB/C	430		1
SEP 2	C/D	380		1.1
SEP 3	C/D	340	-	1.1
SEP 4	C/D	360		0.9
SEP 5	C/D	350		0.8

The split fractions in SEP 2 are 0.6 and 0 for component C and D, respectively. The conversion is with respect to the limiting component B. Cost pre-factor κ relates the total molar flow at the inlet (F_i^T) to the annualized cost: $C_i = \kappa_i \left(F_i^T\right)^{0.6}$.

There are a total of five process streams that require heating or cooling ($\{2,7,12,13,15\}$ in Figure 2-10). Together with two isothermal streams that represent the heat duties of SEP1 and CSTR3, the total number of streams in the heat integration is seven, leading to eight temperature intervals. Due to the selection of alternative units and recycling, the hot/cold identities of streams 2, 7, 12, and 13 are unknown a priori.

The resulting MINLP model consists of 1267 variables (298 binaries) and 1645 constraints. It was solved to global optimality in 34.2 seconds with an objective of $14.5 MM/yr. CSTR1 is selected for the first reaction, and RXN1 takes place at 500 K with a 0.9 conversion of reactant B. The effluent stream is subsequently sent to a flash tank operating at 430 K. After mixing with the recycle stream, intermediate C is converted to D in CSTR3 at 386 K. The equilibrium constant is determined to be 0.84 and the single-pass conversion of C in RXN2 is 0.46. Following the second reaction, SEP5 is selected for the final product purification. The final product D is produced at a rate of 0.92 kmol/s, leading to an overall yield of 92% with respect to limiting reactant B. After heat integration, the process requires 14 MW external heating and while no cooling is required.

Note that while the recycle ratio for the first reaction is 0.18, all the unreacted component C are recycled back to CSTR3 to maximize production of D. Another interesting observation is that after mixing with the recycle stream, the feed to CSTR3 is already at the reactor operating temperature, which makes stream 12 redundant in the heat integration (see Figure 2-11). This shows that our model is not only able to minimize energy consumption through heat integration between process streams, but also account for temperatures and energy flows of the overall process.

For comparison we generate a superstructure using the alternative representation shown in Figure 2-9a. Now the process streams that require heating/cooling are introduced right before the processing units and therefore the total number of process streams increases to 10, which leads to 11 temperature intervals. The resulting model yields the same optimal cost, as expected, but it requires a solution time of 523 seconds, which is one order-of-magnitude longer compared to that in the previous representation.

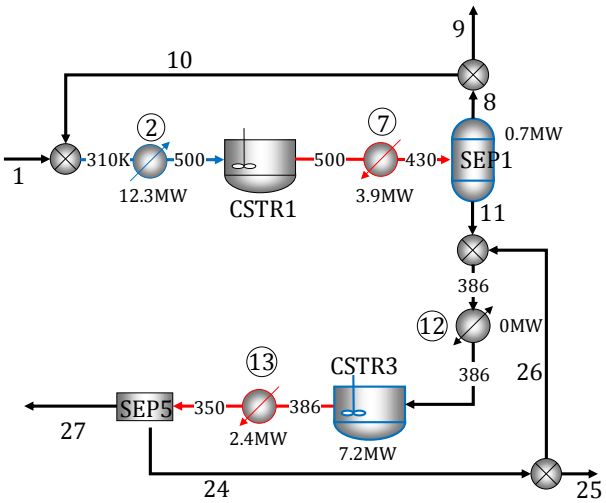


Figure 2-11. Process structure in the optimal solution.

2.7 Conclusions

We proposed a targeting model that can be integrated with models for process synthesis. Importantly, the proposed model accounts for process streams that cannot be classified as hot or cold, a feature that often arises in process synthesis problems, but has not been addressed in the literature. The model is based on the introduction of stream classification binary variables and employs a "dynamic" temperature grid onto which (variable) stream inlet and outlet temperatures are mapped. Through temperature and heat duty variable disaggregation, we calculate heat duties for each stream in each interval. The model was extended to handle multiple utilities, isothermal and non-isothermal streams, and process streams with phase changes. Finally, a superstructure representation technique and several preprocessing methods were discussed to aid the solution process. The proposed methods allow us to address general process synthesis problems where the type of temperature-changing operations is unknown prior to optimization.

Chapter 3
Simultaneous Utility and Heat Exchanger Area Targeting[5]

3.1 Motivation

Despite the work in the area, all previously proposed approaches to simultaneous process synthesis and heat integration, including the one introduced in Chapter 2, do not consider utility and area targeting. Accordingly, in this chapter we propose a mathematical-programming-based model for simultaneous utility and heat exchanger area targeting with variable stream conditions, which can be integrated with process synthesis (see Figure 3-1). The model is built upon the composite-curve-based area targeting method by Townsend and Linnhoff.[91] We introduce "dynamic" enthalpy intervals to account for variable stream temperatures and flowrates. Without obtaining the heat exchanger network, we estimate the heat exchanger areas at each interval[92], which are then related to the capital investment.

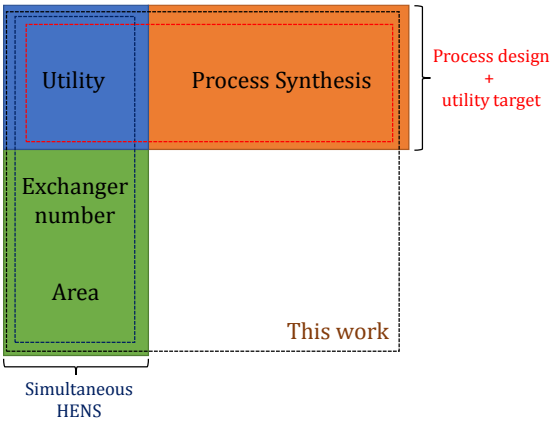


Figure 3-1. A classification of work in the field of heat integration.

[5] The contents of this chapter appear in Kong et al. *I&ECR.* **2017**

3.2 Composite-curve-based Area Targeting

In the composite-curve-based area targeting method with fixed stream conditions, the composite curves that include all the process streams and utilities are first obtained.[91] Since utilities are included, two composite curves are aligned vertically on both ends (i.e. there are no "uncovered" sections). The curves are then divided vertically into different sections, namely enthalpy intervals (see Figure 3-2). The boundaries of these enthalpy intervals are associated with the stream inlet or outlet temperatures. This approach assumes completely countercurrent stream matching (i.e. "vertical" matching), that is, heat can only be exchanged between hot and cold streams in the same enthalpy interval. Therefore, a true area target can be achieved only if all the streams have equal film heat transfer coefficients. On the other hand, if the heat transfer coefficients are different, the area target can be overestimated. Nevertheless, the composite-curve-based approach provides a reasonable approximation of heat exchanger areas when the differences among stream heat transfer coefficients are within one order of magnitude.[93] Further, the error in area calculation will lead to a small percent error in the entire heat exchanger network, which in turn, leads to an even smaller error in the total process cost. This level of error is negligible when the goal of the optimization is to evaluate structural alternatives.

The temperature profiles of all the hot and cold streams in each interval follow the temperature profile of the corresponding composite curve. Therefore, log-mean temperature differences (ΔT_k^{LM}) are the same for all stream pairs at interval k. The resulting heat exchanger network (HEN) at each interval will involve stream splitting and heat exchangers are placed in parallel at the branches of stream splits. Upon splitting, the flowrates and exchanger heat load at each branch are chosen so that isothermal mixing is achieved. Therefore, the area of each individual exchanger at each interval can be calculated using ΔT_k^{LM} and the total heat exchanger area (A^{TOT}) is the summation

of all the individual exchanger areas from all intervals. As shown in Eq. (3.1), A^{TOT} can be alternatively estimated from the stream heat duties at each interval ($Q_{i,k}^{H}$ and $Q_{j,k}^{C}$) without generating the exchanger network,

$$A^{TOT} = \sum_k \frac{1}{\Delta T_k^{LM}} \left(\sum_i \frac{Q_{i,k}^{H}}{\eta_i} + \sum_j \frac{Q_{j,k}^{C}}{\eta_j}\right) \tag{3.1}$$

where η_i and η_j are the heat transfer coefficients of hot and cold streams, respectively.

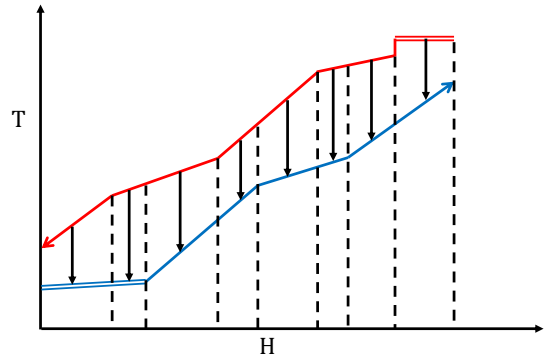


Figure 3-2. Enthalpy intervals in the composite-curve-based area targeting.

When the area target is combined with utility target and simultaneous process synthesis, the tradeoffs within the HEN and the interactions between the HEN and the process can be accounted for using an economic metric.

3.3 Proposed Targeting Model

The proposed targeting model reproduces the composite-curve-based method using a mathematical representation to allow variable stream temperatures and flowrates. For now, we assume that each stream has constant heat capacity. We also assume one shell per heat exchanger and do not consider shell targeting. We use **I** to denote the set of hot streams, which includes hot process streams ($\mathbf{I}^{P}$) and hot utilities ($\mathbf{I}^{HU}$); and **J** to denote the set of cold streams, including cold process streams ($\mathbf{J}^{P}$) and cold utilities ($\mathbf{J}^{CU}$).

3.3.1 Dynamic Enthalpy Grid

Under the assumption of constant heat capacity, the composite curves are represented by line segments in the temperature-enthalpy graph. Therefore, to construct the composite curves we only need to find the temperature and enthalpy levels at the points where the composite curves change slope (i.e. the stream inlets or outlets). When the stream conditions are all fixed, the hot and cold composite curves can be readily constructed from the stream temperatures and enthalpies. However, in our model the variable temperature and flowrate lead to unknown ordering of stream inlet and outlet temperatures, making the definition of enthalpy intervals nontrivial.

Accordingly, to reproduce the enthalpy intervals in composite-curve based targeting, we introduce a "dynamic" enthalpy grid onto which stream temperatures and enthalpies are mapped, in the correct order. Each grid point corresponds to one interval boundary and thus is defined by one stream inlet or outlet. Since each stream has one inlet and one outlet, the number of grid points is, at most, twice the total number of streams, including utilities. However, we can reduce the number of grid points to $2(|\mathbf{I}| + |\mathbf{J}|) - 2$ due to the overlap at the first and last grid point (see Figure 3-3). In other words, we allow two stream inlets or outlets at the first and last grid point while all other grid points correspond to exactly one stream inlet or outlet. Here we use $\mathbf{K}$ to denote a set of grid points and $\mathbf{K}^I = \mathbf{K}\backslash\{0\}$ to denote the set of enthalpy intervals. Following this convention, interval k is in between grid points $k-1$ and k.

The mapping is based on a set of binary variables that represents the mapping of a stream inlet/outlet and onto a grid point:

(1) $X_{i,k}^H / X_{j,k}^C$ equal to one if inlet of stream i/j is at grid point k

(2) $Y_{i,k}^H / Y_{j,k}^C$ equal to one if outlet of stream i/j is at grid point k, and

(3) $Z^H_{i,k}/\ Z^C_{j,k}$ equal to one if stream i/j spans enthalpy interval k

If the inlet of a hot stream is at grid point $k1$ and its outlet is at $k2$ ($k1 > k2$), it will span intervals from $k2 + 1$ to $k1$; and if a cold stream inlet is at grid point $k3$ and its outlet is at $k4$ ($k4 > k3$), it will span intervals from $k3 + 1$ to $k4$. This is enforced by the following constraints,

$$Z^H_{i,k} = Z^H_{i,k-1} - X^H_{i,k-1} + Y^H_{i,k-1} \qquad i \in \mathbf{I}, k \in \mathbf{K}^I \qquad (3.2)$$

$$Z^C_{j,k} = Z^C_{j,k-1} + X^C_{j,k-1} - Y^C_{j,k-1} \qquad j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (3.3)$$

Further, each stream inlet or outlet should be assigned to exactly one grid point,

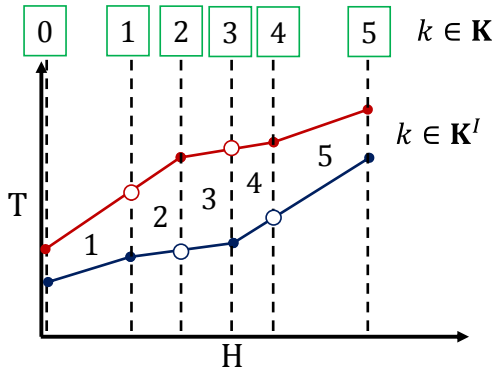


Figure 3-3. Composite curves and enthalpy grid for two hot streams and two cold streams.

$$\sum_k X^H_{i,k} = 1 \qquad i \in \mathbf{I} \qquad (3.4)$$

$$\sum_k Y^H_{i,k} = 1 \qquad i \in \mathbf{I} \qquad (3.5)$$

$$\sum_k X^C_{j,k} = 1 \qquad j \in \mathbf{J} \qquad (3.6)$$

$$\sum_k Y^C_{j,k} = 1 \qquad j \in \mathbf{J} \qquad (3.7)$$

Finally, to aid the solution process we constrain the number of stream inlets and outlets mapped to each grid point. In general, if we define the number of grid points as $2(|\mathbf{I}| + |\mathbf{J}|) - 2$, then the first

and last grid points will each be assigned two stream inlet/outlet while all the other grid points are assigned one,

$$\sum_i (X_{i,k}^H + Y_{i,k}^H) + \sum_j (X_{j,k}^C + Y_{j,k}^C) = \delta_k \qquad k \in \mathbf{K} \tag{3.8}$$

where $\delta_k = 1$ for $k \in \mathbf{K}\backslash\{0, |\mathbf{K}|\}$ and $\delta_k = 2$ for $k = \{0, |\mathbf{K}|\}$ due to the overlap.

3.3.2 Temperatures and Heat Duties

First, we introduce a pair of "grid" temperature variables (T_k^H and T_k^C) to represent the temperatures of the hot and cold composite curves at grid point k. They are ordered as follows,

$$T_k^H \geq T_{k-1}^H \qquad k \in \mathbf{K} \tag{3.9}$$

$$T_k^C \geq T_{k-1}^C \qquad k \in \mathbf{K} \tag{3.10}$$

To satisfy the laws of thermodynamics, the hot and cold composite curves should not "cross". Under the constant heat capacity assumption, the pinch locations can only be at the grid points. Therefore, we only need to introduce a non-negative minimum difference (ϵ) between T_k^H and T_k^C at each grid point to prevent the hot and cold composite curves from crossing,

$$T_k^H - T_k^C \geq \epsilon \qquad k \in \mathbf{K} \tag{3.11}$$

The parameter ϵ can be any sufficiently small positive number (e.g. 1K) since the approach temperature (the minimum of $T_k^H - T_k^C$ for all $k \in \mathbf{K}$) will be determined by optimization.

Second, using the binary variables and logic introduced in Section 3.3.1 we propose a mixed-integer formulation to map the stream inlet and outlet temperatures to the grid. Specifically, each inequality below becomes equality when the binary variable is one, while it is relaxed when the binary is zero,

$$TIN_i^H - \alpha 1_{i,k}(1 - X_{i,k}^H) \leq T_k^H \leq TIN_i^H + \alpha 2_{i,k}(1 - X_{i,k}^H) \qquad i \in \mathbf{I}, k \in \mathbf{K} \tag{3.12}$$

$$TOUT_i^H - \beta 1_{i,k}(1 - Y_{i,k}^H) \le T_k^H \le TOUT_i^H + \beta 2_{i,k}(1 - Y_{i,k}^H) \qquad i \in \mathbf{I}, k \in \mathbf{K} \qquad (3.13)$$

$$TIN_j^C - \alpha 1_{j,k}(1 - X_{j,k}^C) \le T_k^C \le TIN_j^C + \alpha 2_{j,k}(1 - X_{j,k}^C) \qquad j \in \mathbf{J}, k \in \mathbf{K} \qquad (3.14)$$

$$TOUT_j^C - \beta 1_{j,k}(1 - Y_{j,k}^C) \le T_k^C \le TOUT_j^C + \beta 2_{j,k}(1 - Y_{j,k}^C) \qquad j \in \mathbf{J}, k \in \mathbf{K} \qquad (3.15)$$

where $\alpha 1/\alpha 2$ and $\beta 1/\beta 2$ are nonnegative parameters for the big-M constraints. These parameters represent the largest difference between a stream temperature and a grid temperature. They are determined from the bounds on each pair of stream temperature and grid temperature, and are set to zero in the case that the temperature difference is always negative. For example, $\alpha 1_{i,k}$ can be chosen as $\max(0, \overline{TIN}_i^H - \underline{T}_k^H)$; while $\beta 2_{j,k} = \max(0, \overline{T}_k^C - \underline{TOUT}_j^C)$. Note that throughout this work, we use overbars/underbars to represent the upper/lower bounds on the corresponding variables. If T_1^H is always larger than TIN_2^H (i.e. $\underline{T}_1^H \ge \overline{TIN}_2^H$), then $\alpha 1_{2,1}$ is set to zero and the constraints for this pair become: $TIN_2^H \le T_1^H \le TIN_1^H + \alpha 2_{2,1}(1 - X_{2,1}^H)$. Also note that for each pair of stream and grid point, $\alpha 1/\beta 1$ can be very different from $\alpha 2/\beta 2$ since they represent relative temperature differences. The mapping is illustrated using an example in Figure 3-4.

Through Eqs. (3.2) – (3.15), stream temperatures are assigned to the appropriate grid temperatures when the corresponding binary variables *X* or Y are equal to one. Note that each grid point k has two corresponding temperatures, T_k^H and T_k^C, one for the hot and one for the cold composite curve. The mapping is performed for only one of the two temperatures at each grid point.

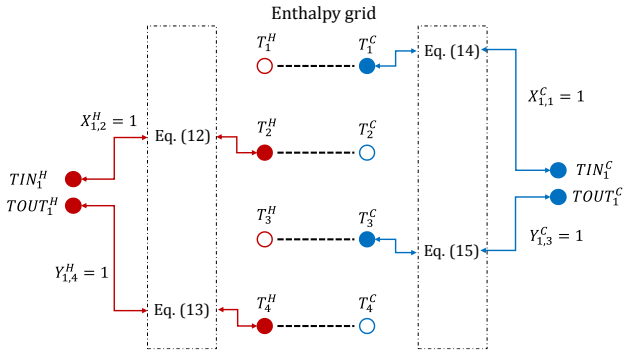


Figure 3-4. An example of mapping one hot stream and one cold stream onto an enthalpy grid.

If we define $2(|\mathbf{I}| + |\mathbf{J}|) - 2$ grid points and enforce Eq. (3.8), then at each grid point one of the two points on the composite curves does not correspond to any stream inlet or outlet (shown as hollow points in Figure 3-3 and Figure 3-4). It is of great importance to detect these "hollow points" in order to calculate all the temperatures in the grid. Accordingly, we introduce binary variables U_k^H/U_k^C that are equal to one when there is no hot/cold stream inlet or outlet locates at grid point k.

$$U_k^H = 1 - \sum_i (X_{i,k}^H + Y_{i,k}^H) \qquad k \in \mathbf{K} \qquad (3.16)$$

$$U_k^C = 1 - \sum_j (X_{j,k}^C + Y_{j,k}^C) \qquad k \in \mathbf{K} \qquad (3.17)$$

If $U_k^H = 1$, the slope of the hot composite curve at grid point k should not change; while if $U_k^C = 1$, the slope of the cold composite curve remains unchanged at this point. Mathematically, it implies that the total heat capacity flowrates ($F_k^{H,TOT}$ and $F_k^{C,TOT}$) of the two intervals adjacent to grid point k are equal (Figure 3-5). Accordingly, we formulate the following mixed-integer constraints in which the heat capacity flowrate equality is enforced if and only if the binary variable U is equal to one,

$$F_k^{H,TOT} + F_k^{H,TD} = F_{k+1}^{H,TOT} + F_k^{H,TP} \qquad k \in \mathbf{K}^I \setminus \{|\mathbf{K}|\} \qquad (3.18)$$

$$F_k^{H,TD} \leq \zeta_k^H (1 - U_k^H) \qquad k \in \mathbf{K}^I \qquad (3.19)$$

$$F_k^{H,TP} \le \zeta_k^H (1 - U_k^H) \qquad k \in \mathbf{K}^I \qquad (3.20)$$

$$F_k^{C,TOT} + F_k^{C,TD} = F_{k+1}^{C,TOT} + F_k^{C,TP} \qquad k \in \mathbf{K}^I \setminus \{|\mathbf{K}|\} \qquad (3.21)$$

$$F_k^{C,TD} \le \zeta_k^C (1 - U_k^C) \qquad k \in \mathbf{K}^I \qquad (3.22)$$

$$F_k^{C,TP} \le \zeta_k^C (1 - U_k^C) \qquad k \in \mathbf{K}^I \qquad (3.23)$$

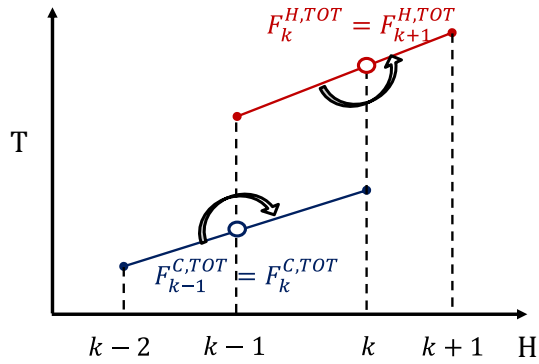


Figure 3-5. Heat capacity flowrate equality enforced at the adjacent intervals to the hollow points where no stream enters or finishes at the composite curve.

where $F_k^{H,TD}, F_k^{H,TP}, F_k^{C,TD}$, and $F_k^{C,TP}$ are nonnegative slack variables that are (de)activated by U, and ζ_k^H / ζ_k^C are upper bounds on total heat capacity flowrates at interval k. For instance, when $U_{k1}^H = 1$, no hot stream inlet or outlet locates at grid point $k1$. Thus, Eqs. (3.19) and (3.20) enforce $F_{k1}^{H,TD} = F_{k1}^{H,TP} = 0$, and Eq. (3.18) enforces total-heat-capacity-flowrate equality (i.e. $F_{k1}^{H,TOT} = F_{k1+1}^{H,TOT}$). On the other hand, when $U_{k1}^H = 0$, $F_{k1}^{H,TD}$ and $F_{k1}^{H,TP}$ can take positive values so that the equality is relaxed.

When both stream temperatures and heat capacity flowrates are variables, the calculation of total heat capacity flowrate at each interval becomes nontrivial. Recall that we introduced binary variables Z to detect if a stream spans a given interval or not. Here we introduce variables $F_{i,k}^{H,P}$ and $F_{j,k}^{C,P}$ to represent the heat capacity flowrate of streams i/j in interval k. If a hot stream i spans

interval k (i.e. $Z_{i,k}^{H} = 1$), then $F_{i,k}^{H,P}$ will be equal to the actual stream heat capacity flowrate (F_i^H); while if the stream does not span interval k, $F_{i,k}^{H,P}$ will be forced to zero and F_i^H will be equal to a slack variable ($F_{i,k}^{H,D}$) instead. Finally, the summation of $F_{i,k}^{H,P}$ for all the hot streams in interval k yields the total heat capacity flowrate $F_k^{H,TOT}$,

$$F_{i,k}^{H,P} + F_{i,k}^{H,D} = F_i^H \qquad i \in \mathbf{I}, k \in \mathbf{K}^I \qquad (3.24)$$

$$F_{i,k}^{H,D} \leq \theta_{i,k}^{H}(1 - Z_{i,k}^{H}) \qquad i \in \mathbf{I}, k \in \mathbf{K}^I \qquad (3.25)$$

$$F_{i,k}^{H,P} \leq \theta_{i,k}^{H} Z_{i,k}^{H} \qquad i \in \mathbf{I}, k \in \mathbf{K}^I \qquad (3.26)$$

$$F_k^{H,TOT} = \sum_i F_{i,k}^{H,P} \qquad k \in \mathbf{K}^I \qquad (3.27)$$

where $\theta_{i,k}^{H}$ are upper bounds on $F_{i,k}^{H,P}$. Similar constraints are formulated for the cold streams:

$$F_{j,k}^{C,P} + F_{j,k}^{C,D} = F_j^C \qquad j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (3.28)$$

$$F_{j,k}^{C,D} \leq \theta_{j,k}^{C}(1 - Z_{j,k}^{C}) \qquad j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (3.29)$$

$$F_{j,k}^{C,P} \leq \theta_{j,k}^{C} Z_{j,k}^{C} \qquad j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (3.30)$$

$$F_k^{C,TOT} = \sum_j F_{j,k}^{C,P} \qquad k \in \mathbf{K}^I \qquad (3.31)$$

where $\theta_{j,k}^{C}$ are upper bounds on $F_{j,k}^{C,P}$.

Finally, the heat duty of each individual stream ($Q_{i,k}^{H}$ or $Q_{j,k}^{C}$) can be calculated from the temperature differences and the heat capacity flowrates at the interval (Figure 3-6). Note that the enthalpy level at each grid point (H_k) can be simultaneously calculated by accumulating the total stream heat duties at each interval, assuming $H_0 = 0$ and H_k increases with k.

$$H_k - H_{k-1} = \sum_i Q_{i,k}^{H} = \sum_i F_{i,k}^{H,P}(T_k^H - T_{k-1}^H) \qquad k \in \mathbf{K}^I \qquad (3.32)$$

$$H_k - H_{k-1} = \sum_j Q_{j,k}^C = \sum_j F_{j,k}^{C,P}(T_k^C - T_{k-1}^C) \qquad k \in \mathbf{K}^I \qquad (3.33)$$

In fact, it is not necessary to calculate H_k in order to estimate the areas. As shown in Eq. (3.1), if all the heat transfer coefficients are given, we only need to calculate stream heat duties at each interval and the log-mean temperature differences. Therefore, Eqs. (3.32) – (3.33) can be equivalently written as a single constraint that enforces energy balance at each interval:

$$\sum_i F_{i,k}^{H,P}(T_k^H - T_{k-1}^H) = \sum_j F_{j,k}^{C,P}(T_k^C - T_{k-1}^C) \qquad k \in \mathbf{K}^I \qquad (3.34)$$

For now we assume that all streams, including utilities, are non-isothermal. In Section 3.4, we will discuss how isothermal streams can be handled. Please also note that, unlike the approach of Duran, Grossmann [94], the use of dynamic enthalpy intervals means we have to introduce binary variables and formulate the model as an MINLP.

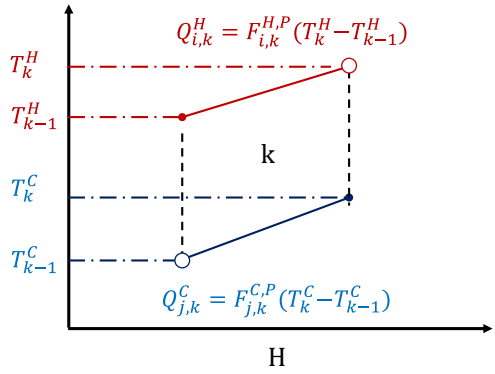


Figure 3-6. Heat duty calculations at interval k.

3.3.3 Areas and Objective

Recall that in order to estimate the exchanger area at each interval a log-mean temperature difference (ΔT_k^{LM}) must be calculated (Eq. (3.1)). To avoid the numerical difficulties when the temperature differences on two boundaries of an interval are equal, here we use an approximation of the log-mean temperature difference [95]:

$$\Delta T_k^{LM} = \frac{2}{3}\sqrt{\Delta T_k \Delta T_{k-1}} + \frac{1}{6}(\Delta T_k + \Delta T_{k-1}) \qquad k \in \mathbf{K}^I \qquad (3.35)$$

where $\Delta T_k = T_k^H - T_k^C$ for all $k \in \mathbf{K}$.

Even with the approximation, temperature difference calculations in Eq. (3.35) are still nonlinear. To further reduce computational complexity, arithmetic mean temperature differences, $\Delta T_k^{LM} = (\Delta T_k + \Delta T_{k-1})/2$, can be used instead.

Since $Q_{i,k}^H$ and $Q_{j,k}^C$ are already calculated via Eqs. (3.32) and (3.33) and the film heat transfer coefficients are assumed to be constants, exchanger areas ($Area_k$ and A^{TOT}) can be calculated through Eq. (3.1).

Finally, the objective is to minimize the total cost, which includes the hot and cold utility costs, and the area cost,

$$\min \sum_{i \in \mathbf{I}^{HU}} v_i^H Q_i^{H,U} + \sum_{j \in \mathbf{J}^{CU}} v_j^C Q_j^{C,U} + f(Area) \qquad (3.36)$$

where v_i^H and v_j^C are prices of hot and cold utilities, respectively, $Q_i^{H,U} = \sum_k Q_{i,k}^H$ and $Q_j^{C,U} = \sum_k Q_{j,k}^C$ are the heat duties of hot/cold utilities, and $f(Area)$ represents a cost function of heat exchanger areas:

- $f(Area) = \kappa(A^{TOT})^{\phi}$, where the area cost is calculated from the power-law function of the total area,
- $f(Area) = \sum_k [\kappa(Area_k)^{\phi}]$, where the area cost is a summation of area cost from all intervals, or
- $f(Area) = \sum_{ij} \mu V_{i,j} + \kappa A^{TOT}$, which is a linear approximation of the nonlinear area cost,

where parameters κ, ϕ, and μ represent the cost prefactor, exponent of the power-law function, and the fixed cost for stream matching, respectively; and $V_{i,j}$ is a binary variable which is one if

the matching between streams i and j exists. It is activated via heat exchange ($Q^{EX}_{i,j,k}$) between streams i and j at interval k,

$$Q^{EX}_{i,j,k} \leq \xi_{i,j} V_{i,j} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (3.37)$$

which is related to the stream heat duties,

$$Q^{H}_{i,k} = \sum_j Q^{EX}_{i,j,k} \qquad i \in \mathbf{I}, k \in \mathbf{K}^I \qquad (3.38)$$

$$Q^{C}_{j,k} = \sum_i Q^{EX}_{i,j,k} \qquad j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (3.39)$$

Since the exchanger network configuration is not determined, the economy of scale is represented by a power law function of total exchanger area (option 1), or the exchanger area at each interval (option 2). Alternatively, in the last option the economy of scale is accounted for by adding a fixed cost for each pair of streams i and j on top of a linear cost of total area. The last approach is linear, but requires additional variables and constraints.

3.4 Extensions

3.4.1 Isothermal Streams and Multiple Utilities

As presented in Eqs. (3.32) – (3.34), the stream heat duties are calculated as the product of heat capacity flowrates and the interval temperature differences. If isothermal streams (e.g. steam) are present, using the same inlet and outlet temperatures leads to difficulties when constructing the enthalpy grid and when calculating heat duties.

Accordingly, we introduce a small temperature difference (here assumed 1K) between the inlet and outlet of an isothermal stream, which means that, for isothermal streams, F^H_i / F^C_j represent the latent heat flowrates instead of heat capacity flowrates. The temperature difference, in principle,

can be chosen differently, e.g., define a 0.1K temperature difference and let F_i^H/F_j^C be 10 times the latent heat flowrates. In general, using a smaller temperature difference can lead to slightly more accurate results, but the computation can become harder because the resulting F_i^H/F_j^C can be very large, which leads to poor scaling.

As mentioned previously, utilities are included in the set **I** and **J**. Therefore, the heat duties and exchanger areas for multiple utilities are simultaneously calculated. Note that it is necessary to define F_i^H/F_j^C for utilities for the calculation of $F_{i,k}^{H,P}/F_{j,k}^{C,P}$, which are then used to obtain heat duties and areas in each interval these utilities span. Thus, isothermal and non-isothermal utilities will be handled like isothermal and non-isothermal process streams, respectively, when calculating heat duties and exchanger areas.

3.4.2 Unclassified Streams

So far, we have assumed that any process stream in the model either belongs to $\mathbf{I}^P$ or $\mathbf{J}^P$. However, it is possible that some streams cannot be classified as hot or cold prior to solving the optimization problem. This can happen, for example, in the superstructure-based process optimization when alternative units operate at different temperatures.

Following the approach by Kong et. al.[84], we extend the model to consider unclassified process streams. First, we use **S** to denote the set of unclassifed process streams. Variables previously defined over $i \in \mathbf{I}$ or $j \in \mathbf{J}$ are now also defined over $s \in \mathbf{S}$ (e.g. $X_{s,k}^H$).

In addition, for each unclassified stream we introduce a pair of classification binaries (W_s^H and W_s^C) to represent the hot/cold identities. The classification binaries, W_s^H/W_s^C, will be equal to one if the stream is hot/cold, while they will be set to zero otherwise. Using these classification binaries, we determine the stream identities by comparing the inlet and outlet temperatures:

$$W_s^H + W_s^C = 1 \qquad s \in \mathbf{S} \qquad (3.40)$$

$$TIN_s - TOUT_s = T_s^+ - T_s^- \qquad s \in \mathbf{S} \qquad (3.41)$$

$$T_s^+ \leq \gamma_s W_s^H \qquad s \in \mathbf{S} \qquad (3.42)$$

$$T_s^- \leq \gamma_s W_s^C \qquad s \in \mathbf{S} \qquad (3.43)$$

where T_s^+ and T_s^- are nonnegative slack variables, and γ_s is an upper bound on the difference between inlet and outlet temperatures.

Using classification binaries, we modify constraints in Section 3.3. Since each stream should be assigned to only one grid point and each stream is either classified as hot or cold, Eqs. (3.4) – (3.7) for unclassified streams become:

$$\sum_k X_{s,k}^H = W_s^H \qquad s \in \mathbf{S} \qquad (3.44)$$

$$\sum_k Y_{s,k}^H = W_s^H \qquad s \in \mathbf{S} \qquad (3.45)$$

$$\sum_k X_{s,k}^C = W_s^C \qquad s \in \mathbf{S} \qquad (3.46)$$

$$\sum_k Y_{s,k}^C = W_s^C \qquad s \in \mathbf{S} \qquad (3.47)$$

For all the remaining constraints in Section 3.3, we need to replace subscripts "i" and "j" with "s", and replace sets "**I**" and "**J**" with "**S**". For example, Eq. (3.26) becomes: $F_{s,k}^{H,P} \leq \theta_{s,k}^H Z_{s,k}^H, s \in \mathbf{S}, k \in \mathbf{K}^I$. Through these constraints, if s' is a hot stream ($W_{s'}^H = 1$ and $W_{s'}^C = 0$), then Eqs. (3.46) and (3.47) force all the $X_{s',k}^C$ and $Y_{s',k}^C$ to zero, which further set $Z_{s',k}^C$ and continuous variables with superscript "C" (e.g. $F_{s',k}^{C,P}$ and $Q_{s',k}^C$) to zero for all $k \in \mathbf{K}^I$; while if s' is a cold stream, Eqs. (3.44) and (3.45) deactivate the "hot" counterpart of these variables with superscript "H".

Since the number of grid points required is a function of the total number of streams, following the rules in Section 3.3 we introduce a grid with $2(|\mathbf{S}| + |\mathbf{I}| + |\mathbf{J}|) - 2$ grid points. Therefore, regardless of the number of hot and cold streams in the final solution, the enthalpy grid can be constructed by linking binary variables X and Y with classification binaries, and the areas and heat duties at each interval can be calculated from constraints in Section 3.3.

3.4.3 Integration with Process Synthesis

The heat integration model can be solved in conjunction with a process synthesis model. In this case, the stream temperatures and heat capacity flowrates are the linking variables that couple the heat integration and process synthesis modules (Figure 3-7). The sets of hot and cold process streams ($\mathbf{I}^P$ and $\mathbf{J}^P$) in the heat integration module include a subset of streams in the process and a set of streams that represents the heat duties of processing units.[84] For additional details on how to integrate a process model with a heat integration model, the reader is referred to our previous works.[84,90] Given the heat transfer coefficients and utility prices, the integrated model finds the optimal stream temperature and heat capacity flowrate that optimize the overall process economy. The utility consumption and exchanger area are obtained from the heat integration module, while the annualized cost of the processing units, material costs, and revenue are calculated from the process synthesis model.

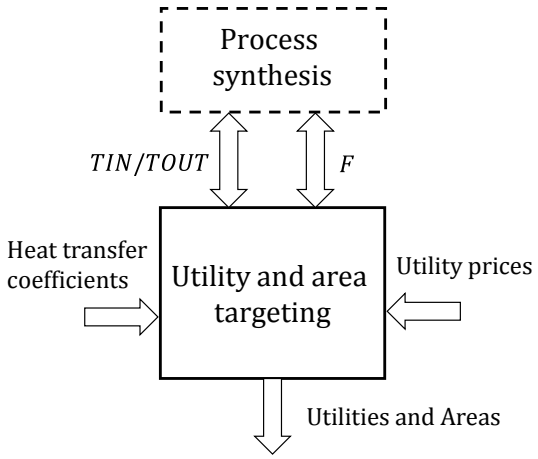


Figure 3-7. Integration of process synthesis and heat integration.

3.5. Solution Methods

3.5.1 Number of Grid Points

As mentioned before, Eq. (3.8) improves the solution efficiency by introducing a one-to-one correspondence between stream inlet/outlet and grid points for all but the first and last grid points. However, it can be modified to further reduce solution time. For example, if two or more streams have the same inlet or outlet temperatures, then Eq. (3.8) will result in two or more grid points having the same temperatures. In this case, a more effective approach is to define fewer grid points and allow streams with same temperatures to "share" one grid point.

However, by using fewer than $2(|\mathbf{I}| + |\mathbf{J}|) - 2$ grid points, some of the constraints must be modified accordingly. First, Eq. (3.8) must be updated since more than one stream inlet/outlet temperatures can be assigned to the same grid point,

$$1 \leq \sum_i (X_{i,k}^H + Y_{i,k}^H) + \sum_j (X_{j,k}^C + Y_{j,k}^C) \leq \delta \qquad k \in \mathbf{K}\backslash\{0, |\mathbf{K}|\} \tag{3.8b}$$

while for the first and last grid points the equation remains unchanged. The parameter δ represents the maximum number of streams with the same temperatures.

Other necessary modifications are to remove Eqs. (3.16) and (3.17), and rewrite Eqs. (3.19) – (3.20) and (3.22) – (3.23) as follows:

$$F_k^{H,TD} \le \zeta_k^H \sum_i (X_{i,k}^H + Y_{i,k}^H) \qquad k \in \mathbf{K}^I \qquad (3.19b)$$

$$F_k^{H,TP} \le \zeta_k^H \sum_i (X_{i,k}^H + Y_{i,k}^H) \qquad k \in \mathbf{K}^I \qquad (3.20b)$$

$$F_k^{C,TD} \le \zeta_k^C \sum_j (X_{j,k}^C + Y_{j,k}^C) \qquad k \in \mathbf{K}^I \qquad (3.22b)$$

$$F_k^{C,TP} \le \zeta_k^C \sum_j (X_{j,k}^C + Y_{j,k}^C) \qquad k \in \mathbf{K}^I \qquad (3.23b)$$

3.5.2 Utility Targeting and Preprocessing

The proposed model can be modified to only minimize utility consumption, by simply removing the cost function of exchanger area, $f(Area)$, from the objective. In this way, the exchanger areas are still calculated, although they are not included in the objective. Alternatively, Eq. (3.35) can be removed from the optimization model and the area calculation can be performed after the utility-targeting problem is solved.

Compared to the utility and area targeting model (P1), the utility targeting model (P0) is less complex and can be solved significantly faster. By removing constraint (3.35) and $f(Area)$ from the objective function, we reduce the number of variables and constraints, especially nonlinear constraints. The nonlinear terms in both models are summarized in Table 3-1.

Table 3-1. Comparisons of nonlinearities between P0 and P1.

	Types of nonlinearities
P0 (Utility targeting)	Bilinear: Eqs. (3.32) – (3.34)
P1 (Utility + area targeting)	Bilinear: Eqs. (3.1), (3.32) – (3.34) Power: Eqs. (3.35) and (3.36)

We can trivially show that the optimal solution of the utility targeting model (P0) is a feasible but suboptimal solution of the utility and area targeting model (P1), and since P0 is in general easier

to solve than P1, for some large instances we could start by solving P0, then use the solutions from P0 to aid the solution of P1.

Here we use $(\boldsymbol{X}^*)_{\mathrm{P0}}$ to denote the globally optimal solution of P0 with a given minimum value of the approach temperature (ϵ). Thus, the utility consumption and cost in $(\boldsymbol{X}^*)_{\mathrm{P0}}$ provide lower bounds on the utility consumption and cost in P1. If multiple utilities are present, $(\boldsymbol{X}^*)_{\mathrm{P0}}$ cannot provide lower bounds on each individual utility stream because the individual utility usages that lead to minimum utility cost do not necessary lead to minimum utility consumption. Nevertheless, based on $(\boldsymbol{X}^*)_{\mathrm{P0}}$ we can calculate valid lower bounds on utility costs.

Further, we can use the total area calculated by $(\boldsymbol{X}^*)_{\mathrm{P0}}$ as an upper bound on the total area in P1. Note that although $(A^{TOT})^*_{\mathrm{P0}}$ does not necessary represent the largest total area in any feasible design in P1, it is used as a cutoff to reduce the feasible design space while preserving the optimal solution. This is because $(\boldsymbol{X}^*)_{\mathrm{P0}}$ is a feasible (but suboptimal) solution for P1, and any solution that is better must have higher utility cost and lower area cost. In addition, we can provide an upper bound on the objective of P1 as follows:

$$(Z^*)_{\mathrm{P1}} \leq (Z^*)_{\mathrm{P0}} + f(Area)^*_{\mathrm{P0}} \tag{3.48}$$

where $(Z^*)_{\mathrm{P0}}$ and $f(Area)^*_{\mathrm{P0}}$ are the optimum objective and area cost in P0, respectively.

In summary, the preprocessing procedure includes solving a smaller model (P0) to obtain a suboptimal design (for P1), and then based on the optimal solution of P0 reduce the search space by generating variable bounds and eliminating designs that are not as good as $(\boldsymbol{X}^*)_{\mathrm{P0}}$. As shown in the examples in Section 3.6, implementing this procedure can lead to reduction in solution time.

3.5.3 Variable Bounds

The proposed model is a nonconvex MINLP, which means global optimization solvers should be used to find the globally optimal solution. The generation of tight variable bounds results in tighter convex relaxations of nonconvex constraints, leading to potential improvements when using global optimization solvers [96]. Further, the parameters ($\alpha, \beta, \zeta,$ and θ) in the variable upper bound constraints in Section 3.3 are related to the variable upper bounds and thus smaller parameters can lead to tighter relaxations. Accordingly, we introduce the following methods to calculate bounds and some important parameters.

As discussed in Section 3.3.2, parameters α and β are calculated from the upper and lower bounds on the stream inlet/outlet and grid temperatures which are inputs to the model. If our model is used in conjunction with a model for process design, bounds can be inferred from design and unit operation specifications (e.g. the reaction temperature in the reactor). Second, in terms of bounds on grid temperatures, since T_k^H and T_k^C are ordered stream inlet or outlet temperatures, a valid choice is $\overline{T}_k^H = \max_i(\overline{TIN}_i^H, \overline{TOUT}_i^H)$, $\underline{T}_k^H = \min_i(\underline{TIN}_i^H, \underline{TOUT}_i^H)$, $\overline{T}_k^C = \max_j(\overline{TIN}_j^C, \overline{TOUT}_j^C)$, and $\underline{T}_k^C = \min_j(\underline{TIN}_j^C, \underline{TOUT}_j^C)$ for all $k \in \mathbf{K}$. However, in some cases we can obtain tighter bounds by analyzing bounds on stream inlet and outlet temperatures. For example, if we know that cooling water has a lower outlet temperature than any other stream, then T_1^C and T_2^C will be equal to the cooling water inlet and outlet temperatures, respectively. Therefore, $\overline{T}_1^C/\underline{T}_1^C$ and $\overline{T}_2^C/\underline{T}_2^C$ can be updated to be equal to the bounds on cooling water inlet and outlet temperatures, respectively. Similar procedure can be applied whenever we know a priori the matching between a grid temperature and a stream temperature, and in such cases, we propagate bounds on stream temperatures to bounds on grid temperatures.

Parameters $\theta_{i,k}^{H}$ and $\theta_{j,k}^{C}$, which are the upper bounds on $F_{i,k}^{H,P}$ and $F_{j,k}^{C,P}$, respectively, are the same as the upper bounds on the corresponding F_i^H and F_j^C. For process streams ($\mathbf{I}^P$ and $\mathbf{J}^P$), upper bounds on stream heat capacity flowrates are given as inputs to the model; while for utilities, bounds on F_i^H and F_j^C are related to the utility usage ($Q_i^{H,U}$ and $Q_j^{C,U}$) since the temperature difference between a utility stream inlet and outlet is usually known. Therefore, in order to calculate all $\theta_{i,k}^{H}$ and $\theta_{j,k}^{C}$, we first need to estimate upper bounds on $Q_i^{H,U}$ and $Q_j^{C,U}$. The maximum utility usages ($\bar{Q}_i^{H,U}$ and $\bar{Q}_j^{C,U}$) can be obtained by assuming all process stream heat duties are satisfied by utilities:

$$\bar{Q}_i^{H,U} = \sum_{j \in \mathbf{J}_i^P} \bar{F}_j^C (\overline{TOUT}_j^C - \underline{TIN}_j^C) \qquad i \in \mathbf{I}^{HU} \tag{3.49}$$

$$\bar{Q}_j^{C,U} = \sum_{i \in \mathbf{I}_j^P} \bar{F}_i^H (\overline{TIN}_i^H - \underline{TOUT}_i^H) \qquad j \in \mathbf{J}^{CU} \tag{3.50}$$

where $\bar{Q}_i^{H,U}/\bar{Q}_j^{C,U}$ are calculated from bounds on stream temperatures and heat capacity flowrates; $\mathbf{J}_i^P$ is the set of cold streams whose inlet temperatures are lower than the inlet temperature of hot utility i; and $\mathbf{I}_j^P$ is the set of hot streams whose outlet temperatures are higher than the outlet temperature of cold utility j. In other words, $\mathbf{I}_j^P$ and $\mathbf{J}_i^P$ are the streams that can potentially exchange heat with a given utility. If the hot and cold utilities are at the highest and lowest temperatures, respectively, then $\mathbf{J}_i^P = \mathbf{J}^P$ and $\mathbf{I}_j^P = \mathbf{I}^P$. Once $\bar{Q}_i^{H,U}$ and $\bar{Q}_j^{C,U}$ are determined, $\theta_{i,k}^{H}, i \in \mathbf{I}^{HU}$ and $\theta_{j,k}^{C}, j \in \mathbf{J}^{CU}$ can be back calculated:

$$\theta_{i,k}^{H} = \bar{Q}_i^{H,U}/(TIN_i^H - TOUT_i^H) \qquad i \in \mathbf{I}^{HU}, k \in \mathbf{K}^I \tag{3.51}$$

$$\theta_{j,k}^{C} = \bar{Q}_j^{C,U}/(TOUT_j^C - TIN_j^C) \qquad j \in \mathbf{J}^{CU}, k \in \mathbf{K}^I \tag{3.52}$$

where the inlet and outlet temperatures of utilities are assumed fixed.

Finally, the upper bounds on total heat capacity flowrates at interval k, ζ_k^H/ζ_k^C, must be determined. One valid choice is $\zeta_k^H = \sum_{i\in\mathbf{I}_k} \bar{F}_i^H$ and $\zeta_k^C = \sum_{j\in\mathbf{J}_k} \bar{F}_j^C$ for all $\in \mathbf{K}^I$, where $\mathbf{I}_k$ and $\mathbf{J}_k$ are the sets of hot and cold streams that could potentially span interval k, respectively.

In general, the bounds obtained using the approaches in this section are not the tightest, but when combined with the bounds calculated in Section 3.5.3 lead to tighter models and improve computational performance. A preprocessing algorithm that includes the procedures described in Section 3.5.2 and 3.5.3 can be found in the supporting information of [97].

3.6 Illustrative Example

The specifications on stream temperatures are shown in Figure 3-8, while heat capacity flowrates (thereafter referred to as "flows"), and heat transfer coefficients are given in the supporting information of [97]. An isothermal hot utility is assumed to be available at the highest temperature (330°C), while a non-isothermal cold utility enters at 20°C and exits at 30°C.

Since there are 8 streams in total (including utilities), we define an enthalpy grid with 14 grid points. We use Eq. (3.35) to calculate the temperature difference at each interval. The minimum value of approach temperature (ϵ) is assumed to be 5°C. The area cost function is chosen as: $f(Area) = \$40000/\text{yr} \cdot \text{m}^{-1.6}(A^{TOT})^{0.8}$, which approximates the area cost as a power-law function of the total exchanger area.

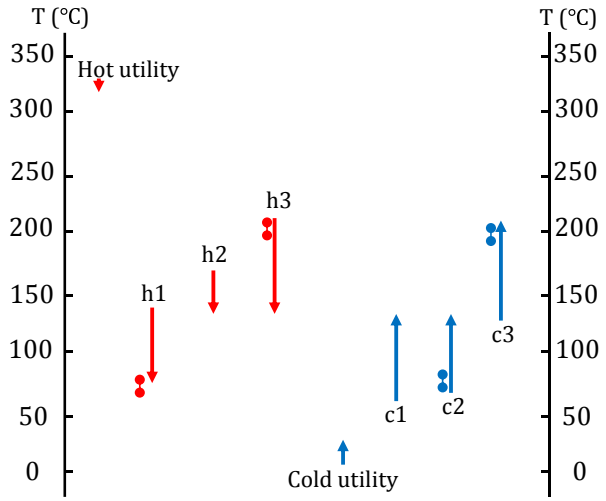


Figure 3-8. Stream optimal temperatures.

Before solving the simultaneous utility and area targeting model (P1), we solve the utility targeting model (P0) to obtain variable bounds. P0 comprises of 719 variables (328 binary) and 1132 constraints (312 nonlinear entries). A globally optimal solution is obtained in 5 seconds. The hot and cold utility consumptions and costs, and the objective value of P0 are then used to lower-bound the corresponding variables in P1. Using the temperatures and heat duties from $(\boldsymbol{X}^*)_{\mathrm{P0}}$, we calculate the areas and area costs and use them as upper bounds in P1.

Next, we solve P1 to simultaneously obtain the utility and area target. P1 includes 747 variables (328 binary) and 1160 constraints (391 nonlinear entries). The globally optimal solution is obtained in 140 seconds. On the other hand, solving P1 directly without generating bounds using $(\boldsymbol{X}^*)_{\mathrm{P0}}$, requires about 22% longer solution time (180 seconds).

In this example, several streams have the same inlet or outlet temperature, which leads to symmetric solutions. To address this, we introduce fewer grid points and allow streams with same temperatures to share one grid point. Specifically, $TIN_{h1}^{H}, TOUT_{h2}^{H}$, and $TOUT_{h3}^{H}$ can be mapped to one grid point; and $TOUT_{c1}^{C}, TOUT_{c2}^{C}$, and TIN_{c3}^{C} can share one grid point. Therefore, the total number of grid points required can be reduced to 10. With the modifications introduced in Section

3.5.1, we solve the example again using 10 grid points. The model now includes only 828 constraints and 531 variables (232 binary variables). The solution time is reduced to 26 seconds.

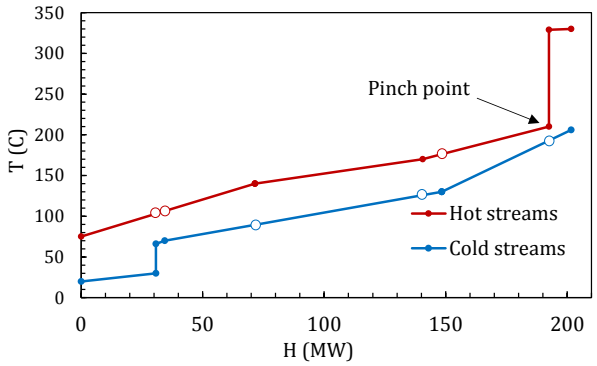


Figure 3-9. Hot and cold composite curves from the optimal solution.

Optimal stream temperatures are shown in Figure 3-8, while other information such as optimal flowrates and utility consumptions can be found in the supporting information of [97]. The total area is 6606 m^2, resulting in a \$4.5MM/yr area cost. The results can be alternatively shown as hot and cold composite curves (see Figure 3-9), from which the locations of the pinch are determined. The approach temperature in this example is found to be 17°C, which is larger than the pre-specified ϵ of 5°C.

3.7 Conclusions

We proposed an MINLP model for simultaneous utility and exchanger area targeting. The model accounts for variable stream temperatures and flowrates, allowing it to be used for simultaneous process synthesis and heat integration. The model represents the composite-curve-based area targeting method. Stream heat duties and exchanger areas are calculated in each interval. We present three alternatives to account for the cost of exchanger area. One possible future work direction is to improve the current area calculation to obtain more accurate area targets. The model can be extended to handle isothermal streams and multiple utilities, as well as streams that cannot

be classified as hot/cold a priori. Furthermore, we discussed several solution techniques, including a preprocessing algorithm for the calculation of tight variable bounds. Finally, we showed how the proposed model can be integrated with a process models and used for simultaneous process synthesis and heat integration.

Chapter 4
Simultaneous Chemical Process and Heat Exchanger Network Synthesis[6]

4.1 Motivation

The approach in Chapter 3 has two important limitations: (1) the nonvertical heat transfer cannot be accounted for, and (2) the estimation on heat exchanger capital cost is not based on an actual heat exchanger network. To address these two limitations, we introduce a new approach for simultaneous process synthesis and heat exchanger network synthesis (HENS) with accurate area and area cost estimation. The objective function is the minimization of the total annualized cost (TAC). Specifically, the TAC of HEN includes both operating cost (i.e., utility cost) and capital investment (i.e., cost of exchanger units).

4.2 Transshipment-based Area Calculation

The expanded-transshipment model by Papoulias and Grossmann[20] provides a simple yet effective way to consider constrained (i.e., forbidden or required) matches in utility targeting when stream temperatures and flow rates are fixed. In that model, heat is treated as a commodity to be transferred from source nodes (i.e., hot streams) to sink nodes (i.e., cold streams) via some intermediate "warehouses" (i.e., temperature intervals). At each interval, hot and cold streams are linked by a variable representing the heat exchange (Q_{ijk}) between hot stream i and cold stream j. Heat residual (R_{ik}) is introduced for each hot stream at interval k, representing the amount of heat

[6] The contents of this chapter appear in Kong et al. *I&ECR.* **2018**

that is still available to be used in the lower temperature intervals. Therefore, the heat exchange is allowed ($Q_{ijk} > 0$) only when the cold stream j is present at interval k, and hot stream i is present either at interval k or at a higher temperature interval.

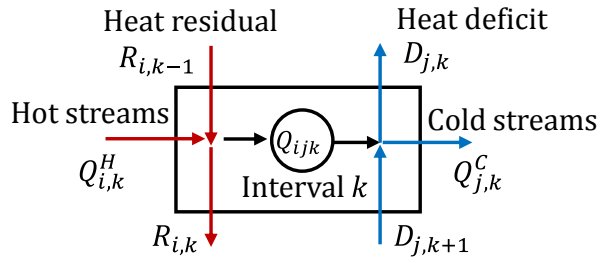


Figure 4-1. Heat cascade for area targeting NLP.

Colberg and Morari[29] incorporated the expanded transshipment model and the composite-curve-based area targeting[13] into a transshipment-based NLP for simultaneous utility and area targeting (thereafter referred to as "area targeting NLP"). The heat cascade in the area targeting NLP is introduced so that both hot and cold streams have the ability to cascade heat across intervals (Figure 4-1). The ability of hot streams to provide heat to cold streams at lower-temperature intervals is modeled through "heat residual", $R_{i,k}$. The cold counter-part is denoted as "heat deficit", $D_{j,k}$, which represents the ability of cold streams to receive heat from intervals at higher temperatures. This modified heat cascade enables the calculation of inlet and outlet temperatures of each heat exchanger unit and thereby the exchanger area.

Unlike the original composite-curve-based area targeting that is solely based on the concept of enthalpy intervals, the approach by Colberg and Morari[29] relies on both temperature and enthalpy intervals (Figure 4-2) and allows heat to be cascaded across intervals so that non-vertical heat transfer is considered, and thus the area calculation is in general accurate. Another advantage of the area targeting NLP is that it considers constrained matching and keeps track of the number of stream matches. However, the area targeting NLP estimates the total heat exchanger area, that is,

it does not yield a design of a realistic network that corresponds to the target. This is because the HEN obtained from the area targeting NLP (see Figure 4-2B) can be overly complicated (e.g., same pair of streams exchanges heat at several intervals), which in general does not correspond to the HEN with minimum TAC.

As pointed out in the previous section, the area targeting NLP assumes fixed stream inlet/outlet temperatures and flow rates. This allows the a priori construction of temperature and enthalpy intervals and the calculation of parametric stream heat duties at each interval. The relaxation of this assumption leads to numerous difficulties in the temperature intervals and heat cascade construction, as will be further illustrated in Section 4.3.

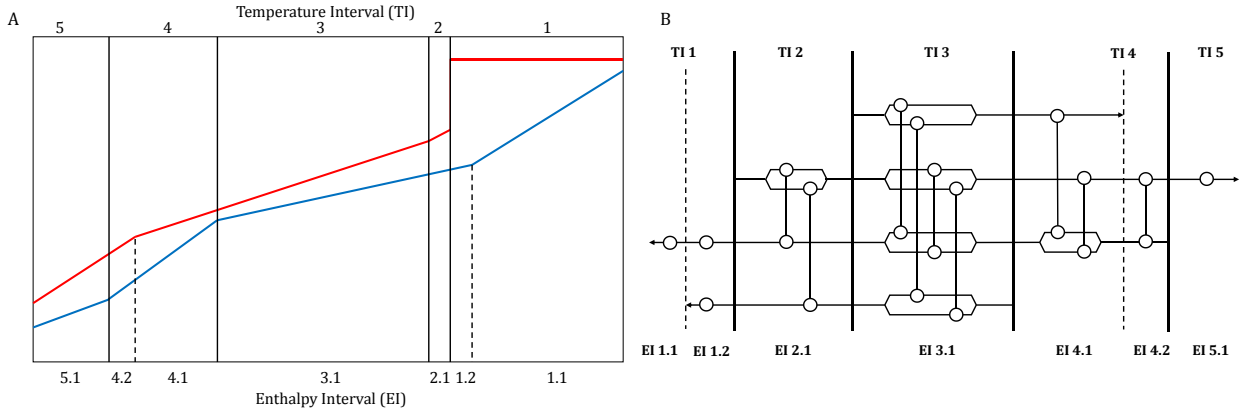


Figure 4-2. An example of (A) the construction of temperature and enthalpy intervals in area targeting NLP and (B) the corresponding heat exchanger network.

Therefore, to extend and generalize the area targeting NLP into a heat exchanger network synthesis model that can be integrated with a process synthesis model, we must address the following main challenges:

(1) Generalize the area targeting NLP to account for variable stream temperatures and flow rates;

(2) Calculate area and cost of each individual heat exchanger in the network; and

(3) Design a realistic heat exchanger network.

The generalization to variable stream temperatures and flow rates is accomplished via the adoption of a heat cascade that is built upon a dynamic temperature grid.[84] The exchanger network design and cost estimation of exchanger units is accomplished via the introduction of additional binary variables and mixed-integer constraints.

4.3 Proposed Model

Without a loss of generality, we make the following assumptions:

(1) Given are a set of hot streams and a set of cold streams with bounds on their inlet/outlet temperatures and flow rates;

(2) Process stream heat capacity and film heat transfer coefficient are given constants;

(3) Cost data of heat exchangers and utilities are given; and

(4) No stream splitting is allowed in the network.

In particular, the last assumption is introduced here for tractability and simplicity. While better solutions might be found by allowing stream splitting, it is not always attractive from a practical standpoint due to more complex operation and additional cost for control.[31] Nevertheless, in Section 4.5, we show that this assumption can be relaxed and the model can be extended to handle stream splitting. Also in Section 4.5, we show how the assumption of given hot/cold streams can be relaxed to deal with more general (i.e., unclassified) streams.

Next, we introduce how the heat cascade is constructed, and then show how the HEN is designed, including how to estimate exchanger area and count the number of exchanger units.

4.3.1 Heat Cascade Construction

The heat cascade in this work is similar to that in the area targeting NLP[29]. At temperature interval $k \in \mathbf{K}$, heat balances are written separately for hot streams ($i \in \mathbf{I}$) and cold streams ($j \in \mathbf{J}$),

$$R_{i,k} + \sum_j Q_{i,j,k} = R_{i,k-1} + Q_{i,k}^H \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (\mathbf{4}.1)$$

$$D_{j,k} + \sum_i Q_{i,j,k} = D_{j,k+1} + Q_{j,k}^C \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (\mathbf{4}.2)$$

where $Q_{i,j,k}$ is the amount of heat exchanged between streams i and j at interval k, and $Q_{i,k}^H$ and $Q_{j,k}^C$ are the heat duty of streams i and j, respectively. Note here that both $Q_{i,k}^H$ and $Q_{j,k}^C$ are variables whose values depend on the relationship between streams and intervals. For example, if a stream i does not span any portion of interval k, $Q_{i,k}^H$ should be set to zero. However, this stream-interval relationship is usually unknown a priori in the context of variable stream temperatures because the boundaries of temperature intervals are also unknown (since they are constructed based on **variable** stream inlet temperatures). Furthermore, the unknown ordering of stream inlet temperatures prevents us from determining the stream membership at each interval, which eventually leads to difficulties in the calculation of $Q_{i,k}^H$ and $Q_{j,k}^C$.

To address this challenge, Kong et al.[84] introduced a model to construct a dynamic temperature grid onto which the stream inlet temperatures can be mapped and ordered. First, nonnegative grid temperatures, T_k, are introduced to represent the temperature of the interval boundaries, and are ordered as follows:

$$T_k \geq T_{k+1} \qquad k \in \mathbf{K} \qquad (4.3)$$

Then, the following binary variables are used to match a stream to an interval:

(1) $X_{i,k}^H / X_{j,k}^C = 1$ if inlet of stream i/j is at grid point (boundary) k;

(2) $Y_{i,k}^{H}/Y_{j,k}^{C} = 1$ if outlet of stream i/j is at interval k; and

(3) $Z_{i,k}^{H}/Z_{j,k}^{C} = 1$ if stream i/j goes through interval k.

When $X_{i,k}^{H}/X_{j,k}^{C} = 1$, we enforce that the shifted stream inlet temperature is equal to the grid temperature (T_k), which is equivalent to the boundary temperature of an interval. Further, interval k is bounded by grid temperatures T_{k-1} and T_k. Through disaggregation, inlet temperatures TIN_i^H and TIN_j^C are assigned to T_k using the following constraints:

$$TIN_i^H = \sum_k TIN_{i,k}^{H,D} \qquad i \in \mathbf{I} \qquad (4.4)$$

$$TIN_{i,k}^{H,D} \geq T_k - \alpha_i(1 - X_{i,k}^{H}) \qquad i \in \mathbf{I},\ k \in \mathbf{K} \qquad (4.5)$$

$$TIN_{i,k}^{H,D} \leq T_k \qquad i \in \mathbf{I},\ k \in \mathbf{K} \qquad (4.6)$$

$$TIN_{i,k}^{H,D} \leq \alpha_i X_{i,k}^{H} \qquad i \in \mathbf{I},\ k \in \mathbf{K} \qquad (4.7)$$

$$TIN_j^C = \sum_k TIN_{j,k}^{C,D} \qquad j \in \mathbf{J} \qquad (4.8)$$

$$TIN_{j,k}^{C,D} \geq T_k - MAT - \alpha_j(1 - X_{j,k}^{C}) \qquad j \in \mathbf{J},\ k \in \mathbf{K} \qquad (4.9)$$

$$TIN_{j,k}^{C,D} \leq T_k - MAT \qquad j \in \mathbf{J},\ k \in \mathbf{K} \qquad (4.10)$$

$$TIN_{j,k}^{C,D} \leq \alpha_j X_{j,k}^{C} \qquad j \in \mathbf{J},\ k \in \mathbf{K} \qquad (4.11)$$

where $TIN_{i,k}^{H,D}$ and $TIN_{j,k}^{C,D}$ are disaggregated inlet temperatures, α_i/α_j are upper bounds on stream inlet temperature, and MAT is the variable minimum approach temperature. Detailed discussion of the above constraints can be found in Kong et al.[84].

Similarly, binaries $Y_{i,k}^{H}$ and $Y_{j,k}^{C}$ are introduced to assign outlet temperatures ($TOUT_i^H$ and $TOUT_j^C$) to interval k via stream disaggregation:

$$TOUT_i^H = \sum_k TOUT_{i,k}^{H,D} \qquad i \in \mathbf{I} \qquad (4.12)$$

$$TOUT_{i,k}^{H,D} \geq T_k - \beta_i(1 - Y_{i,k}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.13)$$

$$TOUT_{i,k}^{H,D} \leq T_{k-1} \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.14)$$

$$TOUT_{i,k}^{H,D} \leq \beta_i Y_{i,k}^H \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.15)$$

$$TOUT_j^C = \sum_k TOUT_{j,k}^{C,D} \qquad j \in \mathbf{J} \qquad (4.16)$$

$$TOUT_{j,k}^{C,D} \geq T_k - MAT - \beta_j(1 - Y_{j,k}^C) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.17)$$

$$TOUT_{j,k}^{C,D} \leq T_{k-1} \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.18)$$

$$TOUT_{j,k}^{C,D} \leq \beta_j Y_{j,k}^C \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.19)$$

where $TOUT_{i,k}^{H,D}$ and $TOUT_{j,k}^{C,D}$ are disaggregated outlet temperatures, and β_i/β_j are upper bounds on stream outlet temperature. The main difference between inlet and outlet temperature assignment is that the inlet temperature is assigned to a point (T_k) while the outlet temperature is assigned to an interval (between T_{k-1} and T_k).

To complete the mapping, the following constraints ensure that the stream inlet/outlet is only assigned to one grid point/interval:

$$\sum_k X_{i,k}^H = 1 \qquad i \in \mathbf{I} \qquad (4.20)$$

$$\sum_k X_{j,k}^C = 1 \qquad j \in \mathbf{J} \qquad (4.21)$$

$$\sum_k Y_{i,k}^H = 1 \qquad i \in \mathbf{I} \qquad (4.22)$$

$$\sum_k Y_{j,k}^C = 1 \qquad j \in \mathbf{J} \qquad (4.23)$$

As shown in Figure 4-3, $Z_{i,k}^H/Z_{j,k}^C$ is coupled with the corresponding $X_{i,k}^H/X_{j,k}^C$ and $Y_{i,k}^H/Y_{j,k}^C$ through the following constraints:

$$Z_{i,k}^H = Z_{i,k-1}^H + X_{i,k-1}^H - Y_{i,k}^H \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \tag{4.24}$$

$$Z_{j,k-1}^C = Z_{j,k}^C + X_{j,k-1}^C - Y_{j,k-1}^C \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \tag{4.25}$$

If $Z_{i,k}^H$ or $Z_{j,k}^C$ is equal to one, the hot or cold stream completely spans (i.e., goes through) interval k, while if $Y_{i,k}^H$ or $Y_{j,k}^C$ is one, the stream partially spans intervals k, and if neither Y nor Z is one, the stream does not span any portion of interval k. At this point, the relationship between streams and intervals is implicitly carried by these binaries, which allows us to formulate constraints to calculate heat duty ($Q_{i,k}^H$ and $Q_{j,k}^C$) at each interval.

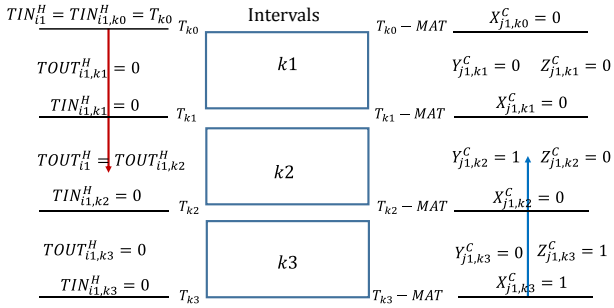


Figure 4-3. Example of stream inlet/outlet temperatures mapped onto the temperature grid via binary variables.

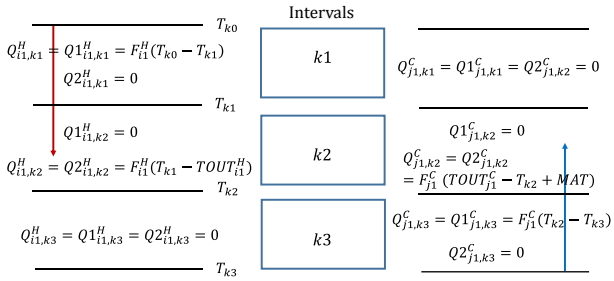


Figure 4-4. Example of heat duty calculation for a hot stream ($i1$) and a cold stream ($j1$).

For simplicity, we present the calculation for hot streams. The calculation for cold streams is very similar and will be presented later. First, we disaggregate $Q_{i,k}^{H}$ into $Q1_{i,k}^{H}$, representing the heat duty when a stream goes through interval k, and $Q2_{i,k}^{H}$, representing the heat duty when a stream partially spans interval k. In this manner, $Q1_{i,k}^{H}$ and $Q2_{i,k}^{H}$ for hot process streams can be calculated differently:

$$Q_{i,k}^{H} = Q1_{i,k}^{H} + Q2_{i,k}^{H} \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.26)$$

$$Q1_{i,k}^{H} \geq F_{i}^{H}(T_{k-1} - T_{k}) - \gamma_{i,k}(1 - Z_{i,k}^{H}) \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.27)$$

$$Q1_{i,k}^{H} \leq F_{i}^{H}(T_{k-1} - T_{k}) \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.28)$$

$$Q1_{i,k}^{H} \leq \gamma_{i,k} Z_{i,k}^{H} \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.29)$$

$$Q2_{i,k}^{H} \geq F_{i}^{H}\left(T_{k-1} - TOUT_{i,k}^{H,D}\right) - \gamma_{i,k}(1 - Y_{i,k}^{H}) \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.30)$$

$$Q2_{i,k}^{H} \leq F_{i}^{H}\left(T_{k-1} - TOUT_{i,k}^{H,D}\right) + \gamma_{i,k}(1 - Y_{i,k}^{H}) \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.31)$$

$$Q2_{i,k}^{H} \leq \gamma_{i,k} Y_{i,k}^{H} \qquad i \in \mathbf{I}^{P},\ k \in \mathbf{K}^{I} \qquad (4.32)$$

where $\gamma_{i,k}$ is an upper bound on heat duty. Heat duties for hot utilities are determined as:

$$Q_{i,k}^{H} \leq \gamma_{i,k} X_{i,k-1}^{H} \qquad i \in \mathbf{I}^{HU},\ k \in \mathbf{K}^{I} \qquad (4.33)$$

Similar constraints are introduced for the cold streams:

$$Q_{j,k}^{C} = Q1_{j,k}^{C} + Q2_{j,k}^{C} \qquad j \in \mathbf{J},\ k \in \mathbf{K}^{I} \qquad (4.34)$$

$$Q1_{j,k}^{C} \geq F_{j}^{C}(T_{k-1} - T_{k}) - \gamma_{j,k}(1 - Z_{j,k}^{C}) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^{I} \qquad (4.35)$$

$$Q1_{j,k}^{C} \leq F_{j}^{C}(T_{k-1} - T_{k}) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^{I} \qquad (4.36)$$

$$Q1_{j,k}^{C} \leq \gamma_{j,k} Z_{j,k}^{C} \qquad j \in \mathbf{J},\ k \in \mathbf{K}^{I} \qquad (4.37)$$

$$Q2_{j,k}^{C} \geq F_j^C\left(TOUT_{j,k}^{C,D} - T_k + MAT\right) - \gamma_{j,k}(1 - Y_{j,k}^{C}) \qquad j \in \mathbf{J},\, k \in \mathbf{K}^I \qquad (4.38)$$

$$Q2_{j,k}^{C} \leq F_j^C\left(TOUT_{j,k}^{C,D} - T_k + MAT\right) + \gamma_{j,k}(1 - Y_{j,k}^{C}) \qquad j \in \mathbf{J},\, k \in \mathbf{K}^I \qquad (4.39)$$

$$Q2_{j,k}^{C} \leq \gamma_{j,k} Y_{j,k}^{C} \qquad j \in \mathbf{J},\, k \in \mathbf{K}^I \qquad (4.40)$$

The heat duty calculation for both hot and cold streams is illustrated in Figure 4-4. Note that while the big-M constraints in this section (e.g., Eqs. (4.27) – (4.28)) can be alternatively formulated using a convex hull reformulation,[98] we found that, in this problem, the big-M reformulation tends to be computationally faster than the convex hull reformulation.

Finally, the heat residual ($R_{i,k}$) and heat deficit ($D_{j,k}$) should be constrained so that they can take nonzero values only in the intervals where heat cascade is allowed. Specifically, for a hot stream i that starts at grid point k, $R_{i,k}$ can be nonzero only from interval $k+1$ to interval $|\mathbf{K}| - 1$; while for a cold stream j that starts at grid point k, $D_{j,k}$ can be nonzero from interval 2 to interval k:

$$R_{i,k} \leq \hat{\gamma}_{i,k} \sum_{k' \leq k-1} X_{i,k'}^{H} \qquad i \in \mathbf{I}, k \in \mathbf{K}^I \qquad (\mathbf{4}.41)$$

$$D_{j,k} \leq \hat{\gamma}_{j,k}(1 - \sum_{k' \leq k-1} X_{j,k'}^{C}) \qquad j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (\mathbf{4}.42)$$

$$R_{i,k} = 0 \qquad i \in \mathbf{I}, k = |\mathbf{K}| \qquad (\mathbf{4}.43)$$

$$D_{j,k} = 0 \qquad j \in \mathbf{J}, k = 1 \qquad (\mathbf{4}.44)$$

where $\hat{\gamma}_{i,k}$ and $\hat{\gamma}_{j,k}$ are upper bounds on $R_{i,k}$ and $D_{j,k}$, respectively.

Through the dynamic temperature grid, the relationship between a stream and an interval is determined, allowing the membership information (i.e., the sets of streams that can provide/receive/cascade heat at each interval) to be implicitly determined via binary variables,

which further enables us to calculate stream heat duties at each interval and complete the heat balances (Eqs. (4.1) and (4.2)) in the heat cascade.

4.3.2 Heat Exchanger Network Synthesis

Recall that we introduced variable $Q_{i,j,k}$ to represent the heat exchanged between streams i and j at interval k. For now, it is treated as the heat duty of an individual heat exchanger, as shown in Figure 4-2B. To estimate the area, $A_{i,j,k}$, of such heat exchanger, we must calculate the inlet and outlet temperatures at its hot ($T_{i,k}^{H,HI}$ and $T_{i,k}^{H,LO}$) and cold ($T_{j,k}^{C,HI}$ and $T_{j,k}^{C,LO}$) ends. These temperatures are related to the stream inlet/outlet temperatures or the grid temperatures, depending on the relationship between the stream and a given interval.

For clarity, we focus on the temperature calculation of a hot stream, while the calculation of a cold stream is similar. First, the process stream inlet temperature (TIN_i^H) corresponds to the exchanger inlet temperature ($T_{i,k}^{H,HI}$) at the first interval this stream spans:

$$T_{i,k}^{H,HI} \geq TIN_i^H - \alpha_i(1 - X_{i,k-1}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.45)$$

$$T_{i,k}^{H,HI} \leq TIN_i^H + \alpha_i(1 - X_{i,k-1}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.46)$$

As shown in Figure 4-5, stream $i1$ starts at point $k0$ (i.e., $X_{i1,k0}^H = 1$) and the first interval it spans is $k1$. Therefore, Eqs. (4.45) – (4.46) enforce that $T_{i1,k1}^{H,HI} = TIN_{i1}^H$.

A hot stream i that spans interval k is not necessary to exchange all its heat duty ($Q_{i,k}^H$) with cold streams in k. Through Eq. (4.1), we allow its heat residual ($R_{i,k}$) to be cascaded to the next interval while the cold counterpart has heat deficit, $D_{j,k}$, to be cascaded to the previous interval. Therefore, at each interval the exchanger outlet temperature ($T_{i,k}^{H,LO}$) depends on how much heat is still

available (i.e., $R_{i,k}$) and the stream heat capacity flow rate (F_i^H). There are two cases that require different temperature calculations:

(1) The process stream completely spans interval k; and

(2) The process stream does not completely span but can cascade heat to interval k.

In case (1), $T_{i,k}^{H,LO}$ is equal to the grid temperature T_k plus $R_{i,k}/F_i^H$. Since case (1) is true when $Z_{i,k}^H = 1$, we introduce the following constraints:

$$T_{i,k}^{H,LO} \geq T_k + \frac{R_{i,k}}{F_i^H} - \alpha_i(1 - Z_{i,k}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.47)$$

$$T_{i,k}^{H,LO} \leq T_k + \frac{R_{i,k}}{F_i^H} + \alpha_i(1 - Z_{i,k}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.48)$$

As shown in Figure 4-6, stream $i1$ spans the entire interval $k1$, and thus $T_{i1,k1}^{H,LO} = T_{k1} + R_{i1,k1}/F_{i1}^H$.

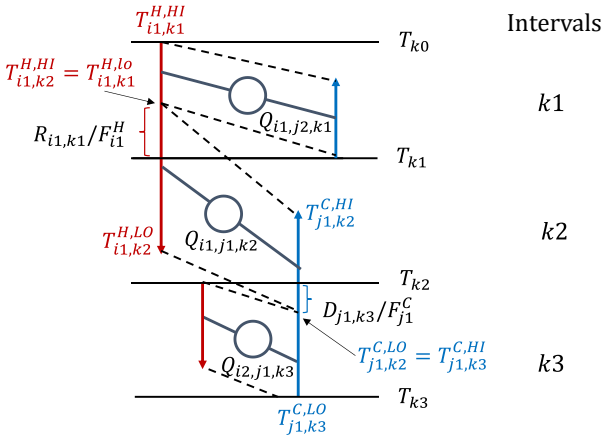


Figure 4-5. Example of exchanger temperature calculation with two hot and two cold streams.

In case (2), the stream does not span the entire interval k (i.e., $Z_{i,k}^H = 0$), but heat cascade is allowed. In other words, if a hot stream ends at interval k (i.e., $Y_{i,k}^H = 1$), case (2) is true for interval k and all the intervals below ($\sum_{k' \leq k} Y_{i,k'}^H = 1$). In this case, the exchanger outlet temperature, $T_{i,k}^{H,LO}$, is determined via the process stream outlet temperature ($TOUT_i^H$) and $R_{i,k}/F_i^H$:

$$T_{i,k}^{H,LO} \geq TOUT_i^H + \frac{R_{i,k}}{F_i^H} - \beta_i(1 - \sum_{k' \leq k} Y_{i,k'}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.49)$$

$$T_{i,k}^{H,LO} \leq TOUT_i^H + \frac{R_{i,k}}{F_i^H} + \beta_i(1 - \sum_{k' \leq k} Y_{i,k'}^H) \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.50)$$

As shown in Figure 4-6, hot stream $i1$ ends at interval $k2$ ($Y_{i1,k2}^H = 1$), but not all its heat duty is used in $k2$; and heat residual is cascaded to interval $k3$ (i.e., $R_{i1,k2} > 0$). Thus, case (2) is true for interval $k2$ and $T_{i1,k2}^{H,LO} = TOUT_{i1}^H + R_{i1,k2}/F_{i1}^H$. Note that case (2) is also true for interval $k3$, but by definition $R_{i1,k3} = 0$, and Eqs. (4.49) – (4.50) enforce that $T_{i1,k3}^{H,LO} = TOUT_{i1}^H$.

Last, to ensure that all heat duty of a stream is used, the exchanger outlet temperature at an interval should be equal to the exchanger inlet temperature at the next interval:

$$T_{i,k}^{H,HI} = T_{i,k-1}^{H,LO} \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.51)$$

The logic is very similar for the case of cold streams, as shown below:

$$T_{j,k}^{C,LO} \geq TIN_j^C - \alpha_j(1 - X_{j,k}^C) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.52)$$

$$T_{j,k}^{C,LO} \leq TIN_j^C + \alpha_j(1 - X_{j,k}^C) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.53)$$

$$T_{j,k}^{C,HI} \geq T_{k-1} - MAT - \frac{D_{j,k}}{F_j^C} - \alpha_j(1 - Z_{j,k}^C) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.54)$$

$$T_{j,k}^{C,HI} \leq T_{k-1} - MAT - \frac{D_{j,k}}{F_j^C} + \alpha_j(1 - Z_{j,k}^C) \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.55)$$

$$T_{j,k}^{C,HI} \geq TOUT_j^C - \frac{D_{j,k}}{F_j^C} - \beta_j \sum_{k' \leq k-1} Y_{j,k'}^C \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.56)$$

$$T_{j,k}^{C,HI} \leq TOUT_j^C - \frac{D_{j,k}}{F_j^C} + \beta_j \sum_{k' \leq k-1} Y_{j,k'}^C \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.57)$$

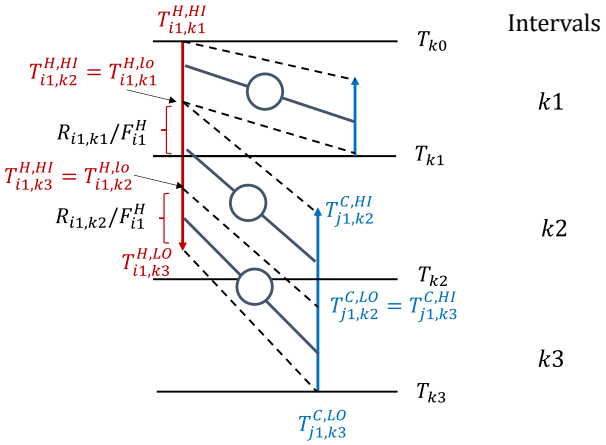


Figure 4-6. Example of exchanger temperature calculation with one hot and two cold streams.

$$T^{C,HI}_{j,k} = T^{C,LO}_{j,k-1} \qquad j \in \mathbf{J},\ k \in \mathbf{K}^{I} \tag{4.58}$$

Based on the inlet/outlet heat exchanger temperatures, we calculate the log-mean temperature difference using an approximation[95] to avoid numerical difficulties:

$$T^{LM}_{i,j.k} = \frac{2}{3}\left[\left(T^{H,HI}_{i,k} - T^{C,HI}_{j,k}\right)\left(T^{H,LO}_{i,k} - T^{C,LO}_{j,k}\right)\right]^{0.5} + \frac{1}{6}\left[\left(T^{H,HI}_{i,k} - T^{C,HI}_{j,k}\right) + \left(T^{H,LO}_{i,k} - T^{C,LO}_{j,k}\right)\right]$$

$$i \in \mathbf{I},\ j \in \mathbf{J},\ k \in \mathbf{K}^{I} \tag{4.59}$$

Given the film heat transfer coefficients (h_i and h_j), the area is determined as follows:

$$A_{i,j,k}T^{LM}_{i,j,k} = \left(\frac{1}{h_i} + \frac{1}{h_j}\right)Q_{i,j,k} \qquad i \in \mathbf{I},\ j \in \mathbf{J},\ k \in \mathbf{K}^{I} \tag{4.60}$$

If only the area target, A^{TOT}, is needed, it can be calculated as the summation of $A_{i,j,k}$ for all combinations of i, j, and k:

$$A^{TOT} = \sum_{i,j,k} A_{i,j,k} \tag{4.61}$$

However, as an inherent drawback of the composite-curve-based area calculation, the resulting HEN can be overcomplicated if we assume that each nonzero $Q_{i,j,k}$ corresponds to an exchanger unit (see Figure 4-2B). If two streams exchange heat at several consecutive intervals, then instead

of introducing multiple "small" heat exchangers at every interval, one heat exchanger unit is needed. This is shown in Figure 4-6 where streams $i1$ and $j1$ exchange heat at intervals $k2$ and $k3$ (i.e., $Q_{i1,j1,k2} > 0$ and $Q_{i1,j1,k3} > 0$), which should be combined and counted as one exchanger unit.

Accordingly, we develop a method to correctly estimate the number of heat exchangers and the area of each individual unit. First, we introduce variable $V_{i,j,k} \in \{0, 1\}$, which is one if streams i and j exchange heat at interval k:

$$\underline{\theta}_{i,j} V_{i,j,k} \leq Q_{i,j,k} \leq \overline{\theta}_{i,j} V_{i,j,k} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.62)$$

where $\underline{\theta}_{i,j}$ and $\overline{\theta}_{i,j}$ are lower and upper bounds on the amount of heat exchanged, respectively. These bounds can be used to enforce required/forbidden matches.

To detect the same match across consecutive intervals, we introduce two additional binary variables:

- $V^S_{i,j,k}$: equal to one if i and j start to exchange heat at interval k
- $V^E_{i,j,k}$: equal to one if the consecutive match between i and j ends at interval k

As shown in Figure 4-7, $V^S_{i,j,k}$ and $V^E_{i,j,k}$ indicate the first and last interval where consecutive matching happens. This balance of matching is enforced as follows:

$$V_{i,j,k} = V_{i,j,k-1} + V^S_{i,j,k} - V^E_{i,j,k-1} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.63)$$

Note that Eq. (4.63) is similar to Eq. (4.24), with $V^S_{i,j,k}$, $V^E_{i,j,k}$, and $V_{i,j,k}$ being the counterparts of $X^H_{i,k}$, $Y^H_{i,k}$, and $Z^H_{i,k}$, respectively. Eq. (4.63) is valid even when matching does not occur in consecutive intervals (see Figure 4-7), in which case both $V^S_{i,j,k}$ and $V^E_{i,j,k}$ are equal to one at the

same interval. A "real" heat exchanger unit exists whenever V_{ijk}^{S} or V_{ijk}^{E} is equal to one. Without a loss of generality, we choose V_{ijk}^{S} to represent the existence of a heat exchanger and use it to estimate the cost of exchanger units.

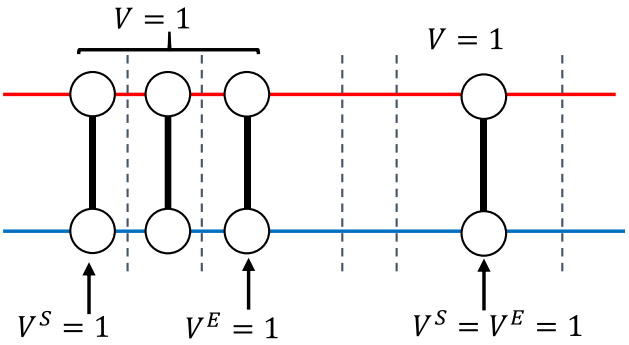


Figure 4-7. An example of binary variables V, V^S, and V^E in a HEN.

In addition, we introduce the following constraints to enforce that no stream splitting is allowed, although this assumption can be relaxed (see Section 4.5.1).

$$\sum_i V_{i,j,k} \leq 1 \qquad j \in \mathbf{J},\ k \in \mathbf{K}^I \qquad (4.64)$$

$$\sum_j V_{i,j,k} \leq 1 \qquad i \in \mathbf{I},\ k \in \mathbf{K}^I \qquad (4.65)$$

Finally, the objective function can be written as follows:

$$\min \sum_{i \in \mathbf{I}^{HU}} \nu_i^H \left(\sum_k Q_{i,k}^H\right) + \sum_{j \in \mathbf{J}^{CU}} \nu_j^C \left(\sum_k Q_{j,k}^C\right) + \nu^A \sum_{i,j,k} A_{i,j,k} + \nu^U \sum_{i,j,k} V_{i,j,k}^S \qquad (4.66)$$

where $\mathbf{I}^{HU}$ and $\mathbf{J}^{CU}$ are the sets of hot and cold utilities, respectively; and ν_i^H, ν_j^C, ν^A, and ν^U are prices of hot and cold utilities, exchanger area, and exchanger units, respectively. Note here that we use the linear approximation of the nonlinear heat exchanger area cost to account for economies of scale.

4.4 Solution Methods

4.4.1 Tightening Constraints

Since even the traditional heat exchanger network synthesis problem, without variable stream temperatures and flow rates, is computationally difficult, we propose tightening constraints to improve the solution of the proposed models using global optimization solvers.[96,99,100]

First, we introduce integer variable $N_{i,j}^{EX}$ to denote the number of heat exchangers for streams i and j across all intervals:

$$N_{i,j}^{EX} = \sum_k V_{i,j,k}^{S} = \sum_k V_{i,j,k}^{E} \qquad i \in \mathbf{I}, j \in \mathbf{J} \qquad (4.67)$$

To reduce the complexity of the resulting networks, we can bound $N_{i,j}^{EX}$, $N_{i,j}^{EX} \leq \overline{N}_{i,j}^{EX}$, where $\overline{N}_{i,j}^{EX}$ can be user-defined or originating from a HEN design requirement.

In addition, we introduce the following tightening constraints:

$$V_{i,j,k}^{S} \leq V_{i,j,k} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^{I} \qquad (4.68)$$

$$V_{i,j,k}^{E} \leq V_{i,j,k} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^{I} \qquad (4.69)$$

Note that in the continuous relaxation, $V_{i,j,k} = 0.5$, $V_{i,j,k-1} = 0.3$, $V_{i,j,k}^{S} = 0.6$, $V_{i,j,k-1}^{E} = 0.4$ satisfy all the constraints in Section 4.3, but this solution is cut off by Eqs. (4.68) and (4.69).

Similar constraints can be introduced for $X_{i,k}^{H}$ and $X_{i,k}^{C}$:

$$X_{i,k-1}^{H} \leq Y_{i,k}^{H} + Z_{i,k}^{H} \qquad i \in \mathbf{I}, k \in \mathbf{K}^{I} \qquad (4.70)$$

$$X_{j,k}^{C} \leq Y_{j,k}^{C} + Z_{j,k}^{C} \qquad j \in \mathbf{J}, k \in \mathbf{K}^{I} \qquad (4.71)$$

Finally, we introduce constraints that are only valid when no splitting is allowed. In this case, matching between the same pair of i and j at two consecutive intervals should be combined (see Figure 4-7), and thus, in the optimal solution, neither $V^S_{i,j,k}$ nor $V^E_{i,j,k}$ should be equal to one at two consecutive intervals:

$$V^S_{i,j,k} + V^S_{i,j,k-1} \leq 1 \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.72)$$

$$V^E_{i,j,k} + V^E_{i,j,k-1} \leq 1 \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.73)$$

In addition, when $V_{i,j,k}$ is equal to one for at least three consecutive intervals (e.g., at intervals $k-1$, k, and $k+1$), both $V^S_{i,j,k}$ and $V^E_{i,j,k}$ must be zero at interval k, which leads to the following constraint:

$$V_{i,j,k-1} + V_{i,j,k} + V_{i,j,k+1} + V^S_{i,j,k} + V^E_{i,j,k} \leq 3 \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.74)$$

To illustrate, Figure 4-8 shows the values of matching binaries for a HEN that satisfy all previous constraints but violate Eq. (4.74). Since some feasible but suboptimal solutions are eliminated, the search space is reduced. The effectiveness of these constraints will be illustrated in Section 4.6.

Intervals	$k1$	$k2$	$k3$	$k4$	$k5$
$V_{i1,j1,k}$	1	1	1	1	1
$V^S_{i1,j1,k}$	1	0	1	0	0
$V^E_{i1,j1,k}$	0	1	0	0	1

Figure 4-8. Example of matching variables of a HEN configuration: shaded entries violate Eq. (4.74).

4.4.2 Number of Grid Points

The number of points in the dynamic temperature grid is a user-defined parameter. On the one hand, a sufficient number of grid points is required for the design of an optimal HEN; but on the

other hand, an excessive number of points should be avoided because it increases the size of the model and, furthermore, leads to symmetric solutions. Therefore, steps are taken to introduce as few grid points as possible while ensuring feasibility and accuracy.

In general, the number of grid points should be at least equal to the number of streams (including utilities) in the HENS module. In that case, the stream inlets and grid points will have a one-to-one correspondence. However, when two or more streams are known a priori to have the same inlet temperature, they can "share" one grid point and therefore fewer points are required.

It is important to point out that, in some cases, additional points are required to ensure that good configurations are not excluded. As shown in Figure 4-9A, if we assume that this configuration is optimal, the proposed model will yield an equivalent one, shown in Figure 4-9B, if and only if an extra interval (grid point) is introduced. The additional interval allows streams $i1$ and $j1$ to exchange heat at two consecutive intervals ($k2$ and $k3$), which is equivalent, in terms of calculating area and counting number of units, to the configuration in Figure 4-9A.

Therefore, we use an iterative procedure to determine the number of grid points, similar to the determination of number of time points in continuous-time scheduling models.[101-103] Specifically, we start from the minimum number of grid points and increase it by 1 until there is no improvement in the objective function. However, we note that as the number of points increases, the model may not be solvable to global optimality within a reasonable time limit, which means that best known solution may be found by a model employing few grid points.

4.4.3 Symmetry-breaking Techniques

Symmetric solutions can arise when additional grid points are not "used" in the solution. For example, in Figure 4-9C and D, the optimal solution can be readily obtained with 3 grid points,

and symmetric solutions exist if 4 grid points are used. In Figure 4-9C, points $k0$, $k1$, and $k2$ are matched with the inlets of streams $i1$, $j2$, and $j1$, respectively; while in Figure 4-9D, grid points $k0$, $k1$, and $k3$ are matched with the stream inlets.

To eliminate potential symmetric solutions, we add a penalty term, $\mu(\sum_k T_k - \sum_i TIN_i^H - \sum_j TIN_j^C - |\mathbf{J}| \cdot MAT)$, to the objective function to reward matching inlet temperatures to grid temperatures with small indices. Parameter μ is a sufficiently small positive number to break symmetry while not altering the optimal solution. In general, a good choice of μ is between 10^{-5} and 10^{-4} of the estimated objective function value, an estimate of which can be obtained from the previous iteration with one less grid point. If the number of grid point is exactly equal to the number of streams, this term is zero due to the one-to-one correspondence. When additional points are introduced, this term breaks the symmetry by penalizing matching inlet temperatures with grid temperatures with large indices (e.g., the configuration in Figure 4-9D is penalized more heavily than the one in Figure 4-9C).

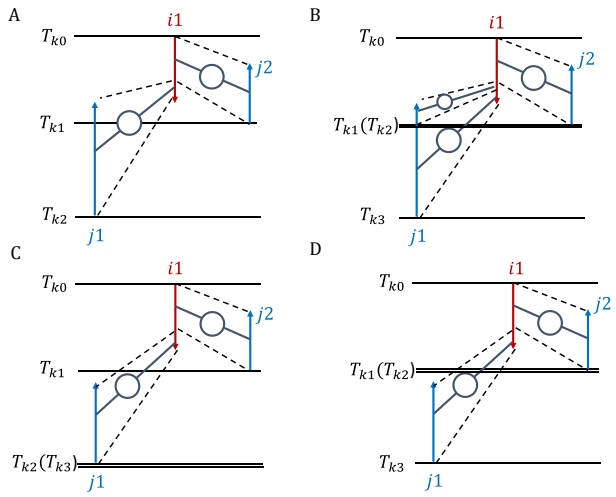


Figure 4-9. Four HEN configurations with one hot and two cold streams.

4.5. Extensions and Remarks

4.5.1 Stream Splitting

The model we have presented so far is based on the assumption that no stream splitting is allowed (Eqs. (4.64) – (4.65)). This assumption was made to not only reduce the complexity of the proposed approach, especially when it is solved in conjunction with a process synthesis model, but to also obtain HENs that are easier to construct. However, the proposed model can be extended to handle stream splitting.

When splitting is allowed, counting the number of heat exchangers becomes more complicated. When a stream splits and exchanges heat with another stream at several consecutive intervals, Eq. (4.63) itself is not sufficient to determine the number of exchanger units (see Figure 4-10). If consecutive matches can be combined into one exchanger, the heat capacity flow rate must be the same for that exchanger across multiple intervals (e.g., the green heat exchanger in Figure 4-10B). This idea has been proposed and referred to as "flow rate consistency" by Barbaro and Bagajewicz.[45] Here, we extend this idea and formulate constraints that are suitable for the proposed HENS model.

We introduce additional variables and constraints to keep track of the number of exchangers when splitting is allowed. Again, we illustrate for the case of hot streams, and present the constraints for cold streams later. First, we calculate the heat capacity flow rate for each pair of streams i and j at interval k ($F_{i,j,k}^{H,E}$):

$$Q_{ijk} = F_{i,j,k}^{H,E}(T_{i,k}^{H,HI} - T_{i,k}^{H,LO}) \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.75)$$

$$F_{i,j,k}^{H,E} \leq \gamma_{i,j,k}^{H} V_{i,j,k} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.76)$$

If the exchanger heat capacity flow rate at two adjacent intervals ($F_{i,j,k}^{H,E}$ and $F_{i,j,k-1}^{H,E}$) are the same, then only one exchanger is required (Figure 4-10B), while if they are different, more than one exchanger should be used (Figure 4-10A). This logic is enforced via the following constraints:

$$F_{i,j,k}^{H,E} - F_{i,j,k-1}^{H,E} = F_{i,j,k}^{H,P} - F_{i,j,k}^{H,N} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \tag{4.77}$$

$$F_{i,j,k}^{H,P} \leq \gamma_{i,j,k}^{H}(V_{i,j,k}^{S} + V_{i,j,k-1}^{E}) \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \tag{4.78}$$

$$F_{i,j,k}^{H,N} \leq \gamma_{i,j,k}^{H}(V_{i,j,k}^{S} + V_{i,j,k-1}^{E}) \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \tag{4.79}$$

where $F_{i,j,k}^{H,P}$ and $F_{i,j,k}^{H,N}$ are nonnegative slack variables that are (de)activated by the binary variables $V_{i,j,k}^{S}$ and $V_{i,j,k-1}^{E}$. To illustrate, we consider a HEN example in Figure 4-11 with one hot and two cold streams. Although $i1$ and $j1$ exchange heat at both $k2$ and $k3$, $F_{i1,j1,k2}^{H,E} \neq F_{i1,j1,k3}^{H,E}$ due to splitting at interval $k2$, and thus $F_{i1,j1,k3}^{H,P}$ is nonzero (Eq. (4.77)). Then, Eq. (4.78) forces $V_{i1,j1,k3}^{S}$ and/or $V_{i1,j1,k2}^{E}$ to be equal to one. When combined with Eq. (4.63), both $V_{i1,j1,k3}^{S}$ and $V_{i1,j1,k2}^{E}$ are equal to one. Therefore, $i1$ and $j1$ require two separate exchangers at intervals $k2$ and $k3$. On the other hand, at interval $k6$, the heat capacity flow is the same as at interval $k6$ (i.e., $F_{i1,j1,k5}^{H,E} = F_{i1,j1,k6}^{H,E}$). In this case, both slack variables on the right-hand-side of Eq. (4.77) are equal to zero. Thus, $V_{i1,j1,k6}^{S}$ and $V_{i1,j1,k5}^{E}$ are not constrained by Eqs. (4.78) – (4.79), but become zero to minimize the objective function.

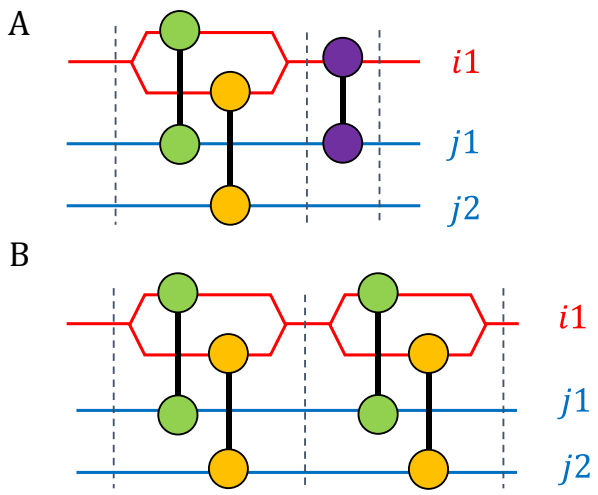


Figure 4-10. Illustration of two cases when splitting is allowed.

Similar constraints are introduced for cold streams:

$$Q_{ijk} = F^{C,E}_{i,j,k}(T^{C,HI}_{j,k} - T^{C,LO}_{j,k}) \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.80)$$

$$F^{C,E}_{i,j,k} \leq \gamma^{C}_{i,j,k} V_{i,j,k} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.81)$$

$$F^{C,E}_{i,j,k} - F^{C,E}_{i,j,k-1} = F^{C,P}_{i,j,k} - F^{C,N}_{i,j,k} \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.82)$$

$$F^{C,P}_{i,j,k} \leq \gamma^{C}_{i,j,k}(V^{S}_{i,j,k} + V^{E}_{i,j,k-1}) \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.83)$$

$$F^{C,N}_{i,j,k} \leq \gamma^{C}_{i,j,k}(V^{S}_{i,j,k} + V^{E}_{i,j,k-1}) \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}^I \qquad (4.84)$$

where $F^{C,E}_{i,j,k}$ is the heat capacity flow rate on the cold side of the pair i and j at interval k, and $F^{C,P}_{i,j,k}$ and $F^{C,N}_{i,j,k}$ are the corresponding nonnegative slack variables.

	k1	k2	k3	k4	k5	k6	k7
$V_{i1,j1,k}$	0	1	1	0	1	1	0
$V^{S}_{i1,j1,k}$	0	1	1	0	1	0	0
$V^{E}_{i1,j1,k}$	0	1	1	0	0	1	0
$F^{H,E}_{i1,j1,k} - F^{H,E}_{i1,j1,k-1}$	0	≠0	≠0	≠0	≠0	0	≠0

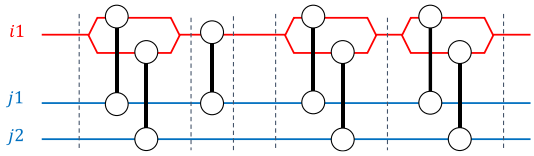


Figure 4-11. A heat exchanger network example with one hot and two cold streams with splitting.

4.5.2 Unclassified Streams

In a process synthesis problem, it is possible to have streams that can be either hot or cold, depending on the final solution, which means that for the HENS module these streams are unclassified. This arises from, for example, the selection of one of multiple alternative units operating at different temperatures. To address unclassified streams, Kong et al.[84] introduced classification binary variables, to implicitly carry the hot/cold identity of unclassified streams, and used them to (de)activate variables and constraints written for the hot and cold counterparts. These ideas can be readily implemented in the proposed model to consider unclassified streams.

We first define a set of unclassified streams ($s \in \mathbf{S}$), and introduce classification binaries (W_s^H / W_s^C) to represent stream identity (i.e., the stream is hot if $W_s^H = 1$ and cold if $W_s^C = 1$). Then, variables originally defined for a hot (e.g., $Z_{i,k}^H$) or a cold (e.g., $Z_{j,k}^C$) stream are both employed to model a single unclassified stream s (e.g., $Z_{s,k}^H$ and $Z_{s,k}^C$). In other words, if $\overline{\boldsymbol{x}}$ represents a vector of variables originally defined for both hot and cold streams (e.g., X, Y, Z, T, Q), then for an unclassified stream, two vectors $\boldsymbol{x}_s^H$ and $\boldsymbol{x}_s^C$ will be introduced to represent the hot and cold counterparts, respectively. In addition, we use both the hot and cold counterparts of the constraints for a single unclassified stream s. Based on the values of the classification binaries, one of the two vectors of variables will be set to zero, and thus the corresponding constraints will be trivially

satisfied, while the other vector will be nonzero, and the corresponding constraints will enforce the necessary constraints. For example, counterparts of Eqs. (4.24) and (4.25) are introduced for unclassified streams $s \in \mathbf{S}$, in which variables with subscripts "i" and "j" are replaced by variables with subscript "s":

$$Z^H_{s,k} = Z^H_{s,k-1} + X^H_{s,k-1} - Y^H_{s,k} \qquad s \in \mathbf{S},\ k \in \mathbf{K}^I \qquad (4.24U)$$

$$Z^C_{s,k-1} = Z^C_{s,k} + X^C_{s,k-1} - Y^C_{s,k-1} \qquad s \in \mathbf{S},\ k \in \mathbf{K}^I \qquad (4.25U)$$

If $W^H_s = 1$ and $W^C_s = 0$, then we set $X^C_{s,k}$, $Y^C_{s,k}$ and $Z^C_{s,k}$ to zero, thereby deactivating Eq. (4.25U). Meanwhile, $X^H_{s,k}$, $Y^H_{s,k}$ and $Z^H_{s,k}$ can be nonzero and Eq. (4.24U) is enforced. Following this logic, it is straightforward to modify other variables and constraints in Section 4.3, and extend the model to account for classified and unclassified streams.

4.5.3 Integration with Process Synthesis

The proposed HEN synthesis model can be solved in conjunction with a process synthesis model (Figure 4-12). These two submodels (modules) are coupled via linking variables: process stream inlet/outlet temperatures and heat capacity flow rates. Bounds on these variables are either given or can be calculated from other design specifications,[12,104] and they can be propagated to generate bounds on other continuous variables (e.g., $R_{i,k}$ and $D_{j,k}$) and to calculate parameters for the variable upper-bound constraints in Section 4.3. Streams in the HENS module comprise of a set of process streams that require heating/cooling and a set of streams representing heat duties of processing units. Since the number of gird points required in the dynamic temperature grid is directly related to the number of streams in the HENS module, we carefully define the set of streams in the process model and generate the process superstructure that minimize the number of streams that require heating or cooling. The idea has been presented in Kong et al.[84]. As shown in

Figure 4-13, the three heat exchangers with dashed boxes can be replaced by the one to their left so that fewer streams require heating or cooling.

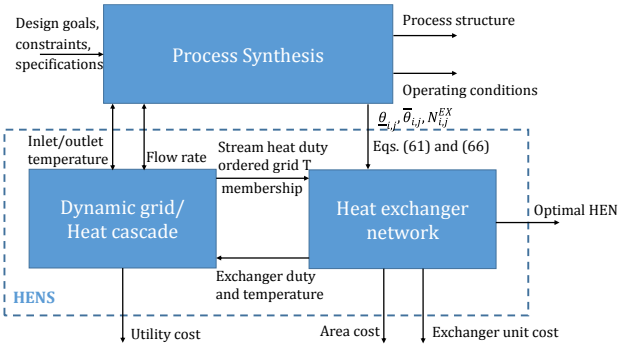


Figure 4-12. Integration process synthesis and heat exchanger network synthesis modules.

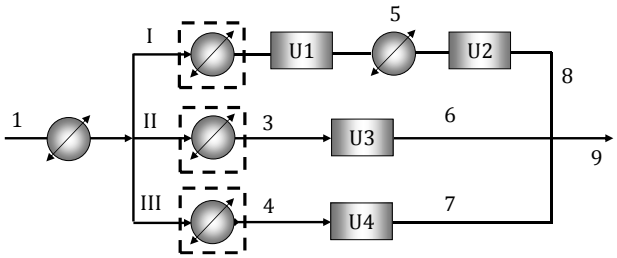


Figure 4-13. Superstructure example showing process stream reduction.

The optimization model for the integrated problem can be expressed in the following general form:

$$\begin{aligned} & \min \Phi(\boldsymbol{x},\boldsymbol{y}) + \Phi^{p}(\boldsymbol{w},\boldsymbol{z}) \\ s.t. \quad & g(\boldsymbol{x},\boldsymbol{y}) = 0 \\ & h(\boldsymbol{x},\boldsymbol{y}) \le 0 \\ & g^{p}(\boldsymbol{w},\boldsymbol{z}) = 0 \\ & h^{p}(\boldsymbol{w},\boldsymbol{z}) \le 0 \\ & f(\boldsymbol{w},\boldsymbol{x}) = 0 \\ & \boldsymbol{w} \in \mathbf{W} \subset \mathbb{R}^{n}_{+}, \boldsymbol{x} \in \mathbf{X} \subset \mathbb{R}^{n}_{+}, \boldsymbol{y} \in \mathbf{Y} \subset \mathbb{R}^{n}_{+}, \boldsymbol{z} \in \mathbf{Z} \subset \mathbb{R}^{n}_{+} \end{aligned} \tag{P1}$$

where vector $\boldsymbol{x} = (F_i^H, TIN_i^H, TOUT_i^H$: all $i \in \mathbf{I}$; $F_j^C, TIN_j^C, TOUT_j^C$: all $j \in \mathbf{J})$ represents the linking variables in the HENS module; $\boldsymbol{w}$ represents the linking variables in the process synthesis module, which includes temperatures and molar flow rates of stream entering HENS, and temperatures and heat duties of some processing units. Vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ represent all other

variables in the HENS and process synthesis modules, respectively. Note that $\boldsymbol{y}$ contains both discrete (e.g., $V_{i,j,k}$) and continuous (e.g., $Q_{i,j,k}$) variables. Similarly, variables in vector $\boldsymbol{z}$ can be either discrete (e.g., unit selection binaries in a process superstructure) or continuous (e.g., unit pressure and sizes). Equations $g(\boldsymbol{x},\boldsymbol{y}) = 0$ and $h(\boldsymbol{x},\boldsymbol{y}) \leq 0$ are constraints in the HENS module (i.e., Eqs. (4.1) – (4.84)); equations $g^p(\boldsymbol{w},\boldsymbol{z}) = 0$ and $h^p(\boldsymbol{w},\boldsymbol{z}) \leq 0$ represent the equality and inequality constraints, respectively, in the process synthesis module (e.g., mass and energy balances); $f(\boldsymbol{w},\boldsymbol{x})$ are constraints that match the linking variables from two modules. For example, if stream $s1$ from the process synthesis module becomes stream $i1$ in the HENS module, then a constraint will be introduced to relate the molar flow rate of component $l \in \mathbf{L}$ in $s1$ ($F_{s1,l}$) to the heat capacity flow rate of $i1$ (F_{i1}^H):

$$F_{i1}^H = \sum_l (F_{s1,l} \cdot Cp_{s1,l}) \tag{4.85}$$

where $Cp_{s1,l}$ is the molar heat capacity of component l in stream $s1$. $\Phi(\boldsymbol{x},\boldsymbol{y})$ is the cost of the HEN (Eq. (4.66)); and $\Phi^p(\boldsymbol{w},\boldsymbol{z})$ is the cost function of the process synthesis module. It typically includes cost of materials, annualized capital cost of processing units, and revenue.

Please note that the classic superstructure-based HENS approaches (e.g. Yee and Grossmann[34]) can be modified to integrate with process synthesis. However, our approach is conceptually different – rather than starting from an equipment superstructure, we build the network upon a transshipment-based framework. However, we note that if sufficient number of grid points are given, and the resulting problems are solved to optimality, then the two approaches would yield the same solutions.

4.6 Illustrative Example

We consider a modification of the process studied in Kong et al.,[105] shown in Figure 4-14. It involves the production of component D from reactants A and B via intermediate C. In the first reaction (RXN1), C is produced in an isothermal (400K) continuous stirred tank reactor (CSTR1). The mixture from the outlet of CSTR1 containing unreacted A and B is separated in a flash tank (SEP1). The raw materials from the top of SEP1 are recycled while the intermediate C is sent to a continuous stirred tank reactor (CSTR2) that operates isothermally (340K) for the production of D. The final product is then purified using two separators (SEP2 and SEP3), and the unreacted C is recycled. We fix the feed flow rate of reactants ($F_A = 1\ kmol/s$ and $F_B = 2\ kmol/s$) and allow the production rate of final product D to vary. The objective is to maximize profit, which includes cost of materials, annualized capital cost, utility cost, heat exchanger cost, and revenue.

As shown in Figure 4-14, streams 3 and 8 are hot and streams 2 and 9 are cold. In addition, the heat of the endothermic reaction in CSTR2 is significant and thus an isothermal cold stream (r1) is introduced in the HENS module. We assume no stream splitting and $\overline{N}_{i,j}^{EX} = 1$ for all $i \in \mathbf{I}$ and $j \in \mathbf{J}$.

The resulting model consists of 1660 constraints and 1035 variables (256 binaries). The best possible solution is found with 6 grid points. The optimization problem is solved to within 4.5% optimality after 3 hours, with an objective function value of \$61.8 MM/yr. The final product D is produced at a rate of 0.965 kmol/s, leading to an overall yield of 96.5% with respect to limiting reactant B. Figure 4-14 and Figure 4-15 show the heat exchanger network configuration along with the optimal process flowsheet. The HEN consists of 6 heat exchanger units, with a total area of 673 m^2. Note that since cold stream 2 exchanges heat with hot utility (i.e., steam) at two

consecutive intervals (see Figure 4-15), only one heat exchanger is introduced (Figure 4-14). The process requires 34 MW external heating and 9.6 MW external cooling, leading to a total utility cost of $2.9 MM/yr.

Interestingly, we observe that hot stream 8 and cold stream 2 could potentially exchange heat to further reduce utility consumption. However, doing so requires at least one additional exchanger unit that will increase the total area due to a small driving force, which out-weights the reduction in utility cost. This example thus shows how the proposed approach considers the trade-off between capital and operating costs when performing simultaneous process and heat exchanger network synthesis. This particular solution will likely be missed by previous methods that do not consider area cost.

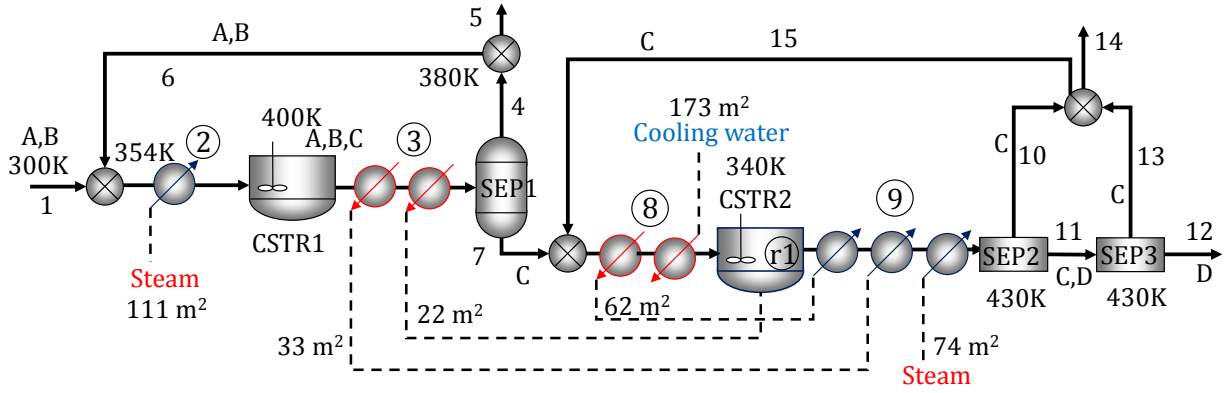


Figure 4-14. Optimal process flowsheet.

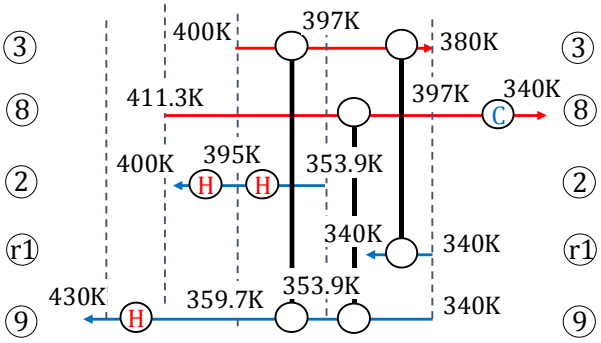


Figure 4-15. Optimal HEN configuration.

4.7 Conclusions

We proposed a mathematical programming model for simultaneous process and heat exchanger network synthesis. This transshipment-based model yields a realistic HEN and calculates an accurate total cost, while considering variable stream temperatures and flow rates. The heat cascade is built upon a dynamic temperature grid so that stream heat duties are accurately calculated at each interval. By allowing both hot and cold streams to cascade heat, non-vertical heat exchange is considered. To correctly identify the number of units, we introduce binary variables to detect heat exchange across consecutive intervals and keep track of heat capacity flow rates when splitting is allowed. Tightening constraints and symmetry breaking techniques were proposed to enhance the solution process, and the impact of the number of grid points on the accuracy and tractability of the proposed models was discussed.

To our knowledge, the proposed model is the first approach in the literature to address, simultaneously, the two major challenges towards the simultaneous process (i.e., reactor and separation network) and heat exchanger network synthesis. First, it yields an actual heat exchanger network, rather than heat and area targets, and thus accurately accounts for the tradeoff between operational and capital costs. Second, it accounts for process unit heating and cooling loads, modeled as streams; variable stream temperatures and flowrates; and process streams that cannot a priori be classified as hot or cold. Therefore, it can be seamlessly integrated with a process synthesis model, potentially leading to solutions that cannot be obtained using existing methods.

Nevertheless, the proposed model is computationally expensive. Therefore, a possible future research direction is the development of computationally better models that can account for the same set of features, as well as the development of additional problem-specific solution methods.

Chapter 5
On the Derivation of Continuous Piecewise Linear Approximating Functions[7]

5.1 Introduction and Motivation

Continuous piecewise linear (PWL) functions have been widely used to fit discrete data points or approximate nonlinear functions. Many papers have focused on how to model continuous piecewise linear functions and how to incorporate these functions into mixed-integer programing (MIP) models [106-111], and its application to mixed-integer nonlinear programming (MINLP) and global optimization [112-118]. Piecewise linear approximations have been used in a wide range of applications including supply network [119], gas/electricity network [120,121], process and systems engineering [118,122], and clustering and classification problems [123-125].

Despite their pervasive application, there has been limited research on how to obtain these piecewise linear functions. For example, how many segments are needed, given an error tolerance, to approximate data; or, given a fixed number of segments, where to place the break points so that the fitting error is minimized. Traditionally, the number and locations of break points are first obtained by heuristics [126-129], and then an unconstrained optimization problem is solved to minimize an error of interest [130,131]. When necessary, constraints can be added, leading to constrained optimization problems.

[7] The contents of this chapter appear in Kong and Maravelias, *INFORMS J.O.C.* Submitted.

Some optimization-based approaches have been proposed to simultaneously determine the location/number of break points and the piecewise linear functions. In terms of approximating continuous functions, Rosen and Pardalos [132] proposed a method to find the minimum number of break points with a given error tolerance assuming the break points are equally spaced; Strikholm [133] proposed a sequential method for finding the number of break points in a piecewise liner function; Rebennack and Kallrath [134] proposed methods to obtain the optimal number and location of break points for univariate functions, and the extensions to bivariate and multivariate functions [135]. In terms of fitting discrete data points, Toriello and Vielma [136] discussed MIP formulations for PWL fitting, focusing on the application to convex functions; and Yang et al. [125] proposed a discontinuous PWL approximation approach that yields the optimal break point locations.

The problem we are interested in is fitting a finite set of data points using a continuous piecewise linear function. From a modeling standpoint, the work by Yang et al. [125] is most related to our problem of interest. However, the method by Yang et. al. [125] does not lead to continuous PWL functions. Given the number of segments of the PWL function, the break point locations should be optimized to obtain a PWL function that minimizes fitting error. If an error tolerance (δ) is specified (i.e., the absolute error at each data point does not exceed δ), the number of break points or number of PWL function segments required should also be a decision variable. The data set (**I**) may come from experiments, input and output of a black box function, or simulation. We assume that the data set has only one independent variable ($x \in \mathbb{R}$) and one output ($y \in \mathbb{R}$), and thus data points are represented by $(x_i, y_i) \in \mathbb{R} \times \mathbb{R}$ for $i \in \mathbf{I}$.

The main novelty of our approach is that it handles any finite one-dimensional data set with continuous piecewise linear functions. In particular, the continuity condition is enforced via a set of linear constraints as opposed to nonlinear in the conventional approach. Therefore, the proposed

fitting models are linear, resulting in two main advantages. First, they remain tractable for relatively large instances, which can result either from data sets with many data points or data sets that require PWL approximations with many segments (e.g., to fit thermodynamic properties with high accuracy in chemical engineering applications). Second, linear models allow us to solve fitting problems fast, which often necessary in the real-time applications where online measurements are used to update equipment models; e.g., in central HVAC optimization, chiller and cooling tower models have to be updated in real time [137]. Moreover, while our models are tailored for fitting discrete data points, they can be extended to approximate univariate functions.

5.2 Fitting Data Points

5.2.1 Problem Definition

Given a finite set of data points $(x_i, y_i) \in \mathbb{R} \times \mathbb{R}$, for $i \in \mathbf{I}$, the goal is to find a continuous piecewise linear function that best fits the data set. In general, the fitting error is measured by an l_q-norm of interest, or $\|\cdot\|_q$. Therefore, the objective function has the following general form:

$$\min \sum_{i\in\mathbf{I}} \|f(x_i) - y_i\|_q \tag{5.1}$$

which is the summation of absolute error between the approximation ($f(x_i)$) and the y-coordinate of the data point (y_i) raised to power q. In this work, we mainly focus on $q \in \{1,2\}$.

Without a loss of generality, we assume that the data set is located in the first quadrant ($0 \le x_i, 0 \le y_i, \forall i \in \mathbf{I}$). We further assume that the x-values of the data points are unique and ordered ($x_i < x_{i+1}, \forall i \in \mathbf{I}$). A set of ordered break points X_k^B, $k \in \mathbf{K}$, is introduced to partition the x-domain into intervals $\mathbb{D}_k = [X_{k-1}^B, X_k^B],\ \forall k \in \mathbf{K}^I = \mathbf{K}\backslash\{0\}$ (see Figure 5-1). The ordered data set is partitioned, by the break points, into ordered subsets, $\mathbf{I}_k$, and $i \in \mathbf{I}_k$ if $x_i \in \mathbb{D}_k$. Note that the membership of

$\mathbf{I}_k$ is unknown a priori, since X_k^B are decision variables. The PWL function segment in each interval, $f^k: \mathbb{D}_k \rightarrow \mathbb{R}$, can be expressed as $f^k(x) = A_k x + B_k$, in which A_k and B_k are variables that represent the slope and intercept of the straight line on which segment k lies. For simplicity, we directly denote A_k and B_k as the slope and intercept of segment k. Therefore, f^k approximates the y-coordinate of data point i if $x_i \in \mathbb{D}_k$ (i.e., $i \in \mathbf{I}_k$).

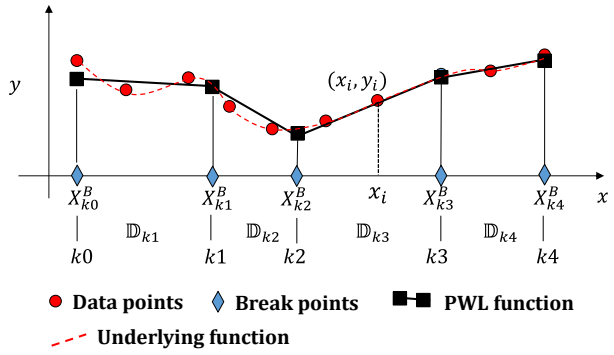


Figure 5-1. Fitting a set of data points with continuous piecewise linear functions.

In the next subsections, we present three mixed-integer programming (MIP) models. In the first two, the number of break points is assumed to be given, and the objective function is error minimization. In the third model, we minimize the number of break points/segments required for a given error tolerance. The key sets, subsets, parameters, and variables are defined as follows:

5.2.2 A Nonlinear Approach

We first present an MINLP model (M0) that is based on known approaches, which will be used as a benchmark, and will be compared against the proposed linear model presented in the next subsections.

First, the set of break points is ordered:

$$X_{k-1}^B \leq X_k^B, k \in \mathbf{K}^I \tag{5.2}$$

We introduce binary variables $Z_{i,k}$, which assign $i \in \mathbf{I}$ to one segment and enforce $x_i \in \mathbb{D}_k$ when $Z_{i,k} = 1$:

$$\sum_{k \in \mathbf{K}^I} Z_{i,k} = 1, i \in \mathbf{I} \tag{5.3}$$

$$X^B_{k-1} - \gamma^L_i \left(1 - Z_{i,k}\right) \leq x_i \leq X^B_k + \gamma^U_i \left(1 - Z_{i,k}\right), i \in \mathbf{I}, k \in \mathbf{K}^I \tag{5.4}$$

where $\gamma^L_i = x_{i=|\mathbf{I}|} - x_i$ and $\gamma^U_i = x_i - x_{i=1}$ are (nonnegative) big M parameters. Through Eqs. (5.2) – (5.4), the membership of $\mathbf{I}_k$ is implicitly determined so that the correct segment is used to approximate a data point. We introduce disaggregated variable $Y_{i,k}$ to represent the approximation of $i \in \mathbf{I}$ at $k \in \mathbf{K}^I$:

$$Y_{i,k} = x_i A_k + B_k, i \in \mathbf{I},\ k \in \mathbf{K}^I \tag{5.5}$$

Since $Y_{i,k}$ represents the approximation of point i if and only if $x_i \in \mathbb{D}_k$ (i.e., $Z_{i,k} = 1$), the absolute error at each data point (E_i) is subject to:

$$E_i \geq Y_{i,k} - y_i - \mu^L_i \left(1 - Z_{i,k}\right), i \in \mathbf{I},\ k \in \mathbf{K}^I \tag{5.6}$$

$$E_i \geq y_i - Y_{i,k} - \mu^U_i \left(1 - Z_{i,k}\right), i \in \mathbf{I},\ k \in \mathbf{K}^I \tag{5.7}$$

where μ^L_i and μ^U_i are sufficiently large positive numbers. However, tight parameters can be calculated from variable bounds. We also show that Eqs. (5.5) – (5.7) can be alternatively formulated using big-M constraints. Note that constraints similar to Eqs. (5.2) – (5.7) are presented in Yang et al. [125].

Finally, the following constraint enforces continuity by requiring the PWL function to be continuous at every break point:

$$X^B_k A_{k+1} + B_{k+1} = X^B_k A_k + B_k, k \in \mathbf{K}^{IM} \tag{5.8}$$

which is similar to constraint (2.4) in Toriello and Vielma [136]. While Eq. (5.8) is straightforward to implement, the bilinear terms ($X_k^B A_k$) and ($X_k^B A_{k+1}$) lead to a nonconvex MINLP:

$$\min \sum_{i\in \mathbf{I}} (E_i)^q \qquad \text{(M0)}$$

$$s.t.\ \text{Eqs.}\,(5.2) - (5.8)$$

Therefore, there are clear advantages if we can replace Eq. (5.8) with a set of linear constraints that enforce continuity.

5.2.3 A Linear Approach – Theoretical Results

In this section, we discuss some theoretical results that are necessary to develop the proposed approach that enforces the continuity of a PWL function. First, we introduce $\mathbf{I}_k^L$ to denote the last element (data point) in $\mathbf{I}_k$. If $i \in \mathbf{I}_k^L$, then $i+1$ is the first element in $\mathbf{I}_{k+1}$. For $k \in \mathbf{K}^{IM}$ and $i \in \mathbf{I}_k^L$, we define a point $\alpha_k = (x_i, Y_i)$ such that $Y_i = f^k(x_i)$ and point $\beta_k = (x_{i+1}, Y_{i+1})$ such that $Y_{i+1} = f^{k+1}(x_{i+1})$. We also define point $\alpha'_k = (x_i, Y'_i)$ such that $Y'_i = f^{k+1}(x_i)$, and point $\beta'_k = (x_{i+1}, Y'_{i+1})$ such that $Y'_{i+1} = f^k(x_{i+1})$, as shown in Figure 5-2.

Proposition 1. *For $k \in \mathbf{K}^{IM}$ and $i \in \mathbf{I}_k^L$, when f^k and f^{k+1} have a unique point of intersection $(\tilde{X}_k^B, \tilde{Y}_k^B)$, $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1}) \geq 0$ if and only if $x_i \leq \tilde{X}_k^B \leq x_{i+1}$.*

Proof of Proposition 1 can be found in the Appendix of [138]. According to Proposition 1, if $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1}) \geq 0$, then $x_i \leq \tilde{X}_k^B \leq x_{i+1}$ for all $i \in \mathbf{I}_k^L$, and $(\tilde{X}_k^B, \tilde{Y}_k^B)$ satisfies both f^k and f^{k+1}. In other words, $\tilde{X}_k^B$ satisfies Eq. (5.8), and thus it represents the location of a break point so that the piecewise linear function is continuous across break point k.

Theorem 1. *If $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1})$ is non-negative for all $k \in \boldsymbol{K}^{IM}$ and $i \in \boldsymbol{I}_k^L$, then the piecewise linear function is continuous.*

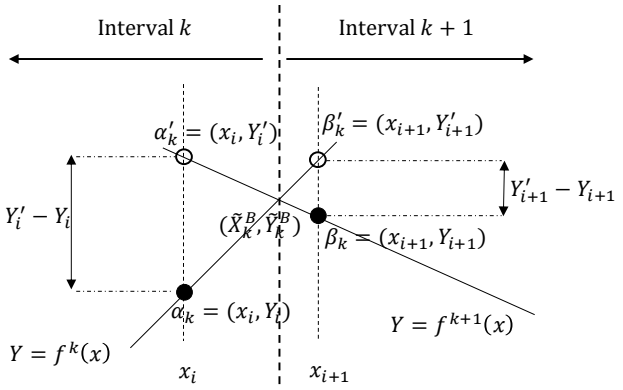


Figure 5-2. Point of intersection of two linear functions.

Proof. For intervals $k \in \mathbf{K}^{IM}$, we have shown that if f^k and f^{k+1} have one intersecting point and $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1}) \geq 0$, then the piecewise linear function is continuous. If $f^k = f^{k+1}$ (have infinitely many points of intersection), then $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1}) = 0$. In this case, the piecewise linear function is also continuous across break point k. If f^k and f^{k+1} have no point of intersection (i.e., $A_k = A_{k+1}$ and $B_k \neq B_{k+1}$), then $Y'_{i+1} - Y_{i+1} = B_k - B_{k+1} = -(Y'_i - Y_i) \neq 0$, which implies that $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1}) < 0$. In this case, the piecewise linear function is clearly discontinuous. Thus, regardless of the number of intersecting points, as long as $(Y'_i - Y_i)(Y'_{i+1} - Y_{i+1}) \geq 0$ is true for all $i \in \mathbf{I}_k^L$ and $k \in \mathbf{K}^{IM}$, the piecewise linear function is continuous. □

Next, we discuss calculation of bounds on the Y variables in Theorem 1 because they are critical to the development of a mixed-integer formulation, discussed in the next subsection, based on Theorem 1. Since $Y_i = A_k x_i + B_k$ and $Y'_i = A_{k+1} x_i + B_{k+1}$, we can bound Y_i and Y'_i (and Y_{i+1} and Y'_{i+1}) if we can bound A_k and B_k.

First, tight bounds on the slope, A_k, can be derived based on, Θ_i, which is the slope of the line connecting adjacent data points i and $i+1$. Therefore, $\overline{A} = \max_{i \in \mathbf{I}} \Theta_i$ and $\underline{A} = \min_{i \in \mathbf{I}} \Theta_i$ are the largest and smallest slope, respectively, among all lines connecting any two adjacent data points.

Given any two data points G and H with distinct x-values (assume that $x^G < x^H$), we define subset $\mathbf{I}^M = \{i: x^G \leq x_i < x^H\}$. From geometry, we have:

$$y^H - y^G = \sum_{i \in \mathbf{I}^M} \Theta_i \Delta x_i \tag{5.9}$$

where $\Delta x_i = x_{i+1} - x_i > 0$. By divide both sides by ($x^H - x^G = \sum_{i \in \mathbf{I}^M} \Delta x_i$), we obtain:

$$slope(GH) = \frac{y^H - y^G}{x^H - x^G} = \frac{\sum_{i \in \mathbf{I}^M} \Theta_i \Delta x_i}{\sum_{i \in \mathbf{I}^M} \Delta x_i} = \sum_{i \in \mathbf{I}^M} \Theta_i \lambda_i \tag{5.10}$$

where $\lambda_i = \Delta x_i / \sum_{i \in \mathbf{I}^M} \Delta x_i$ so that $0 \leq \lambda_i \leq 1$ and $\sum_{i \in \mathbf{I}^M} \lambda_i = 1$. Therefore,

$$\underline{A} \leq \min_{i \in \mathbf{I}^M} \Theta_i \leq slope(GH) \leq \max_{i \in \mathbf{I}^M} \Theta_i \leq \overline{A} \tag{5.11}$$

According to Eq. (5.11), the slope of the line connecting any two data points is bounded by $\underline{A}$ and $\overline{A}$.

Proposition 2. *$\overline{A}$ and $\underline{A}$ are valid upper and lower bounds on A_k, $k \in \boldsymbol{K}^I$.*

The proof of Proposition 2 requires the following Lemmas, whose proofs are given in the Appendix of [138].

Lemma 1. Let point O be the intersection of f^k and a line segment connecting two adjacent data points (x^E, y^E) and (x^F, y^F) with $x^E < x^F$. If $A_k > \overline{A}$, then $y^E \geq A_k x^E + B_k$ and $y^F \leq A_k x^F + B_k$; while if $A_k < \underline{A}$, then $y^E \leq A_k x^E + B_k$ and $y^F \geq A_k x^F + B_k$.

Lemma 2. Let point O be the intersection of f^k and a line segment connecting two adjacent data points. If $A_k > \overline{A}$, then

$$\begin{cases} y_i < A_k x_i + B_k & \forall i: x_i < x^O \\ y_i > A_k x_i + B_k & \forall i: x_i > x^O \end{cases}$$

and if $A_k < \underline{A}$, then

$$\begin{cases} y_i > A_k x_i + B_k & \forall i: x_i < x^O \\ y_i < A_k x_i + B_k & \forall i: x_i > x^O \end{cases}$$

Using the results of Lemma 1 and Lemma2, we prove Proposition 2.

Proof. Assume there exists a segment κ with $A_\kappa > \overline{A}$ or $A_\kappa < \underline{A}$ (the approximating function, f^κ, is shown as line CD in Figure 5-3). f^κ either intersects with the line segment connecting two adjacent data points (Figure 5-3A and C) or passes through a data point (Figure 5-3B and D). We define the intersection as point O. In both cases, from Lemma 2, $f^\kappa = A_\kappa x + B_\kappa$ (i.e., line CD) underestimates the data points on one side of O and overestimates the data points on the other side of O.

Centered at point O, we can rotate f^κ (i.e., line CD) clockwise if $A_\kappa > \overline{A}$, or counterclockwise if $A_\kappa < \underline{A}$ until the first data point (P) lies on f^κ to obtain a new segment $\bar{f}^k$ (i.e., line $C'D'$). The new segment has the following properties:

(1) $\bar{f}^k$ leads to smaller fitting error than f^κ does for $i \in \mathbf{I}_\kappa$ (black dots in Figure 5-3)

(2) The slope of $\bar{f}^k$ (i.e., $C'D'$) is bounded by $\underline{A}$ and $\overline{A}$

Property (1) is clearly true because the rotation reduces the over- and under-estimation for all data points. Since the rotation stops when the first data point lies on the line, the fitting error at every data point in $\mathbf{I}_\kappa$ is strictly reduced (except point O in Figure 5-3B and D). Property (2) is also true because there exists a data point Q, which is adjacent to P, such that the slope of line PQ is at least as large as the slope of $C'D'$ (see Figure 5-3A and B), or at least as small as the slope of $C'D'$ (see Figure 5-3C and D).

Assume that the PWL function remains unchanged for segments other than κ (e.g., lines DD' and CC' in Figure 5-3). To retain continuity, $\mathbf{I}_\kappa$ (black dots in Figure 5-3) must be updated to $\mathbf{I}'_\kappa = \mathbf{I}_\kappa \cup$

$\mathbf{I}_\kappa^N$, where $\mathbf{I}_\kappa^N$ is the set of new data points in segment κ (blue dots). This is because by rotating clockwise, $x^{C'} < x^C$ while $x^{D'} > x^D$, which means that interval $\mathbb{D}_\kappa$ is expanded to $\mathbb{D}'_\kappa$. To ensure continuity, data points that originally belong to other segments are now included in segment κ ($\mathbf{I}_\kappa^N$) and are approximated by $\bar{f}^\kappa$. From Figure 5-3, it is obvious that $\bar{f}^\kappa$ approximates $\mathbf{I}_\kappa^N$ better than the original function segments (DD' and CC') do. This is because the interior of triangles ODD' and OCC' contains no data point, and the slopes of CC' and DD' are no greater than the slope of $C'D'$ (A'_κ) if A'_κ is positive (Figure 5-3A and B) and no less than A'_κ if A'_κ is negative (Figure 5-3C and D). Therefore, $\bar{f}^\kappa$ (i.e., $C'D'$) fits points $i \in \mathbf{I}'_\kappa$ better than f^κ (i.e., CD) does. Note that for all other data points (i.e., red dots) the fitting error remains unchanged.

Therefore, for a segment with $A_\kappa > \overline{A}$ or $A_\kappa < \underline{A}$, we can always find another segment with $\underline{A} \leq A'_\kappa \leq \overline{A}$ such that the fitting error is smaller, which implies that in an optimal solution, A_k, $k \in \mathbf{K}^I$, is bounded by $\overline{A}$ and $\underline{A}$. □

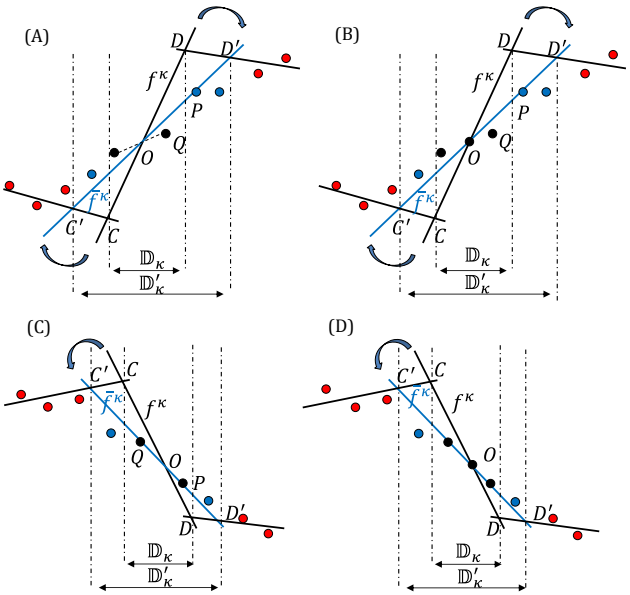


Figure 5-3. Examples of PWL function segment CD replaced by a less "steep" $C'D'$.

Bounds on intercept ($\overline{B}$ and $\underline{B}$) can be calculated, based on $\overline{A}$ and $\underline{A}$, as follows:

$$\overline{B} = y_{i=|\mathbf{I}|} - \underline{A}x_{i=|\mathbf{I}|} \tag{5.12}$$

$$\underline{B} = y_{i=|\mathbf{I}|} - \overline{A}x_{i=|\mathbf{I}|} \tag{5.13}$$

The validity of $\overline{B}$ and $\underline{B}$ can be demonstrated easily using a similar argument as in the proof of Proposition 2. As shown in Figure 5-4, any solution involving segments with $B_k > \overline{B}$ or $B_k < \underline{B}$ (dashed lines) can be represented alternatively by $\underline{B} \leq B_k' \leq \overline{B}$ (black solid lines) such that the objective function value (i.e. fitting error) is at least as good.

Based on the bounds on the A and B variables, we calculate lower ($\underline{Y_i}$) and upper ($\overline{Y_i}$) bounds on the Y variables are bounded as follows:

$$\overline{Y_i} = x_i\overline{A} + \overline{B}, i \in \mathbf{I} \tag{5.14}$$

$$\underline{Y_i} = x_i\underline{A} + \underline{B}, i \in \mathbf{I} \tag{5.15}$$

Note that the bounds introduced in this subsection might not be the tightest possible. Nevertheless, they are essential to the development of the MIP formulation presented next.

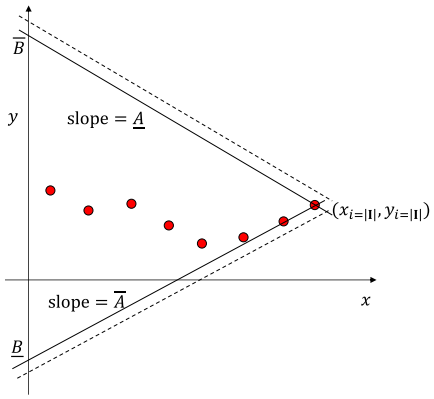


Figure 5-4. An example of finding $\overline{B}$ and $\underline{B}$.

5.2.4 Linear Models

According to the theoretical result presented in the previous subsection, the continuity can be guaranteed if the following disjunction is satisfied:

$$\begin{bmatrix} 0 \le Y_i' - Y_i \le \nu_i^U \\ 0 \le Y_{i+1}' - Y_{i+1} \le \nu_{i+1}^U \end{bmatrix} \vee \begin{bmatrix} \nu_i^L \le Y_i' - Y_i \le 0 \\ \nu_{i+1}^L \le Y_{i+1}' - Y_{i+1} \le 0 \end{bmatrix}, i \in \mathbf{I}_k^L, k \in \mathbf{K}^{IM} \tag{5.16}$$

where $\nu_i^U = \overline{Y_i} - \underline{Y_i}$ and $\nu_i^L = \underline{Y_i} - \overline{Y_i}$ are upper and lower bounds on $Y_i' - Y_i$.

The goal is to represent the disjunction with a set of mixed-integer linear constraints. The main challenge is that we cannot define $\mathbf{I}_k^L$ a priori because of the unknown locations of break points. To address this, we introduce binary variables $Z_{i,k}^F$ and $Z_{i,k}^L$, which are equal to one if i is the first/last data point in segment k. The relationship between $Z_{i,k}^F / Z_{i,k}^L$ and $Z_{i,k}$ is enforced by:

$$Z_{i,k} = Z_{i-1,k} + Z_{i,k}^F - Z_{i-1,k}^L, i \in \mathbf{I}, k \in \mathbf{K}^I \tag{5.17}$$

Each segment must have a first and a last point:

$$\sum_i Z_{i,k}^F = 1, k \in \mathbf{K}^I \tag{5.18}$$

$$\sum_i Z_{i,k}^L = 1, k \in \mathbf{K}^I \tag{5.19}$$

The ordering of segments is enforced through:

$$\sum_{i' \le i} Z_{i',k}^F \ge \sum_{i' \le i} Z_{i',k+1}^F, i \in \mathbf{I}, k \in \mathbf{K}^I \tag{5.20}$$

$$\sum_{i' \le i} Z_{i',k}^L \ge \sum_{i' \le i} Z_{i',k+1}^L, i \in \mathbf{I}, k \in \mathbf{K}^I \tag{5.21}$$

The following two constraints ensure that $Z_{i,k}^F$ and $Z_{i,k}^L$ are always zero when the corresponding $Z_{i,k}$ is zero.

$$Z_{i,k}^F \le Z_{i,k}, i \in \mathbf{I}, k \in \mathbf{K}^I \tag{5.22}$$

$$Z_{i,k}^{L} \le Z_{i,k}, i \in \mathbf{I}, k \in \mathbf{K}^{I} \tag{5.23}$$

The membership information of $\mathbf{I}_k^L$ is dynamically determined by binary variables $Z_{i,k}^{L}$ through Eqs. (5.17) – (5.23). Note that Eq. (5.17) itself does not prevent $Z_{i,k}^{F} = Z_{i-1,k}^{L} = 1$ from occurring. However, according to Eqs. (5.18) – (5.19) and (5.22) – (5.23), every interval must contain at least one data point ($\sum_i Z_{i,k} \ge 1 \, \forall k \in \mathbf{K}^{I}$), which implies that no interval may contain all the data points when two or more intervals are introduced. Therefore, Eqs. (5.17) – (5.23) ensure that $Z_{i,k}^{F} = Z_{i-1,k}^{L} = 1$ cannot occur.

Using $Z_{i,k}^{L}$, we can rewrite the disjunction (Eq. (5.16)) as follows:

$$\left[\begin{bmatrix} 0 \le Y_i' - Y_i \le \nu_i^U \\ 0 \le Y_{i+1}' - Y_{i+1} \le \nu_{i+1}^U \end{bmatrix}^{Z_{i,k}^{L}} \vee \begin{bmatrix} \nu_i^L \le Y_i' - Y_i \le 0 \\ \nu_{i+1}^L \le Y_{i+1}' - Y_{i+1} \le 0 \end{bmatrix}\right] \vee \neg Z_{i,k}^{L}, i \in \mathbf{I}, k \in \mathbf{K}^{IM} \tag{5.24}$$

Eq. (5.24) enforces that when $Z_{i,k}^{L}$ is true, $(Y_i' - Y_i)$ and $(Y_{i+1}' - Y_{i+1})$ are either both nonnegative or both nonpositive; while when $Z_{i,k}^{L}$ is false, no constraint is enforced on $(Y_i' - Y_i)$ or $(Y_{i+1}' - Y_{i+1})$. Since the polyhedra in the disjunction are bounded, Eq. (5.24) is MIP representable [139,140] and can be reformulated into the following mixed-integer linear constraints:

$$x_i A_{k+1} + B_{k+1} - (x_i A_k + B_k) = P_{i,k}^{+} - P_{i,k}^{-}, i \in \mathbf{I}, \; k \in \mathbf{K}^{IM} \tag{5.25}$$

$$x_{i+1} A_k + B_k - (x_{i+1} A_{k+1} + B_{k+1}) = Q_{i+1,k+1}^{+} - Q_{i+1,k+1}^{-}, i \in \mathbf{I}, \; k \in \mathbf{K}^{IM} \tag{5.26}$$

$$P_{i,k}^{+} \le \nu_i^U \left(1 - U_{i,k}\right), i \in \mathbf{I}, \; k \in \mathbf{K}^{IM} \tag{5.27}$$

$$Q_{i+1,k+1}^{+} \le -\nu_i^L \left(1 - U_{i,k}\right), i \in \mathbf{I}, \; k \in \mathbf{K}^{IM} \tag{5.28}$$

$$P_{i,k}^{-} \le \nu_i^U \left(1 - V_{i,k}\right), i \in \mathbf{I}, k \in \mathbf{K}^{IM} \tag{5.29}$$

$$Q^{-}_{i+1,k+1} \leq -v^{L}_{i}\left(1 - V_{i,k}\right), i \in \mathbf{I},\ k \in \mathbf{K}^{IM} \tag{5.30}$$

$$U_{i,k} + V_{i,k} = Z^{L}_{i,k}, i \in \mathbf{I},\ k \in \mathbf{K}^{IM} \tag{5.31}$$

where $P^{+}_{i,k}$, $P^{-}_{i,k}$, $Q^{+}_{i,k}$, and $Q^{-}_{i,k}$ are nonnegative slack variables that are (de)activated by binary variables $U_{i,k}$ and $V_{i,k}$. As shown in Figure 5-5, if $Z^{L}_{i,k} = 1$, then either $U_{i,k}$ or $V_{i,k}$ is equal to one while the other is equal to zero, which enforces either $P^{+}_{i,k}/Q^{+}_{i+1,k+1}$ or $P^{-}_{i,k}/Q^{-}_{i+1,k+1}$ to be zero. Thus, the right-hand-side of Eqs. (5.25) and (5.26) is either nonnegative or nonpositive. If $Z^{L}_{i,k} = 0$, then both $U_{i,k}$ and $V_{i,k}$ are equal to zero, and thus slack variables are not constrained via Eqs. (5.27) – (5.30), thereby relaxing Eqs. (5.25) and (5.26).

The proposed model can be summarized as:

$$\min \textstyle\sum_{i \in \mathbf{I}} (E_i)^{q} \tag{M1}$$

$$s.t.\ \text{Eqs.}\,(5.3), (5.5) - (5.7), (5.17) - (5.23), (5.25) - (5.31)$$

Compared to M0, Eqs. (5.2) and (5.4) are replaced by Eqs. (5.17) – (5.23), and nonlinear Eq. (5.8) is replaced by linear Eqs. (5.25) – (5.31). Therefore, when $q = 1$, the objective function and constraints are linear, leading to an MILP model. When $q = 2$, the convex quadratic objective function and linear constraints result in mixed-integer quadratic convex programming (MIQCP) model.

It is worth noting that while the number of break points/segments must be specified, no constraint is imposed on the location of break points. The break point locations are implicitly constrained via Eqs. (5.25) – (5.31) and can be calculated after the optimization problem has been solved. To avoid confusion, we use lower-case x^{B}_{k} to represent the coordinates of break points that are obtained via post-processing. Also, note that Eqs. (5.17) – (5.23) not only specify the membership information

($\mathbf{I}_k^L$), together with Eq. (5.3), but they also implicitly determine the membership of $\mathbf{I}_k$, which is achieved through Eqs. (5.2) – (5.4) using variable X_k^B. One difference between M0 and M1 is that M0 allows some $\mathbf{I}_k$ to be empty while M1 does not. Nevertheless, if needed, solutions with empty $\mathbf{I}_k$ can be easily generated post optimization.

In both M0 and M1, any solution that involves an "empty" segment (Figure 5-6A) can be equivalently, in terms of objective function value, represented by another solution in which each segment contains at least one data point (see Figure 5-6B,C,D). While M0 allows segments to contain no data points, M1 requires each segment to have at least one data point. Therefore, all four solutions in Figure 5-4 are admitted by M0, and all but (A) are admitted by M1.

Interestingly, while all four solutions have the same objective function value, the one in Figure 5-6B appears to be intuitively the best fit. While M1 would not necessarily yield the solution shown in (B) over the ones shown in (C) or (D), we can post-process (C) or (D) to obtain (B). This post-processing step might be required whenever there are segments with only one data point.

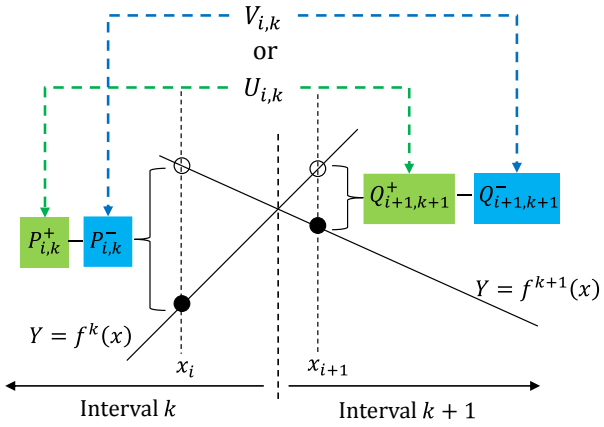


Figure 5-5. An illustration of the mixed-integer reformulation.

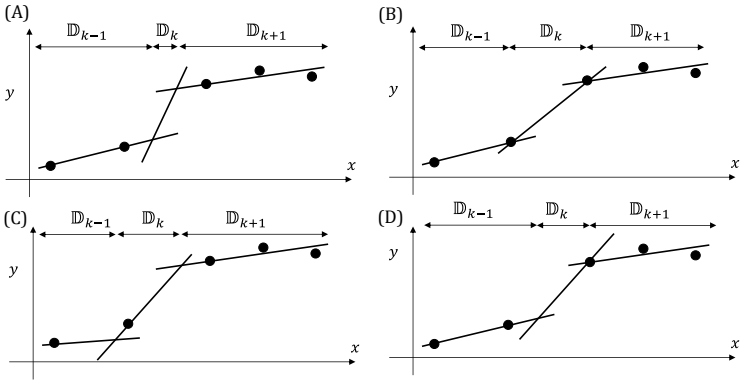


Figure 5-6. An example of fitting 5 data points with a PWL function with 3 segments.

5.2.5 Minimum Number of Break Points

In this section, we show that, with minor modifications, our model can be used to find the minimum number of break points/segments required to ensure that the absolute error at each data point (E_i) is within a user-defined threshold (δ):

$$E_i \leq \delta, i \in \mathbf{I} \tag{5.32}$$

There are, in general, two types of approaches to solve this problem. One approach is to solve M1 (with Eq. (5.32)) successively with increasing number of break points until the problem becomes feasible. While detecting infeasibility can be fast, many iterations might be needed to find the first feasible solution.

Alternatively, we can use a modified model, denoted as M2, to find the minimum number of break points for a given δ, in one step. We define a set $\mathbf{K}$ with $|\mathbf{K}|$ sufficiently large so that in the optimal solution $k \leq |\mathbf{K}|$ points will be required to satisfy the tolerance.

We introduce a binary variable, W_k, which is one if and only if $k \in \mathbf{K}^I$ is active/selected. Eqs. (5.18) and (5.19) are modified as follows:

$$\sum_i Z^F_{i,k} = W_k, k \in \mathbf{K}^I \tag{5.18b}$$

$$\sum_i Z^L_{i,k} = W_k, k \in \mathbf{K}^I \quad (5.19b)$$

to deactivate $Z^F_{i,k}$ and $Z^L_{i,k}$ for all "inactive" segments.

Furthermore, we add the following constraint to reduce symmetric solutions:

$$W_{k+1} \leq W_k, k \in \mathbf{K}^I \quad (5.33)$$

Finally, the objective function becomes:

$$\min \sum_{k \in \mathbf{K}^I} kW_k \quad (5.34)$$

where the multiplication with "k" helps improve the computational performance. Besides these modifications, M2 utilizes same constraints as M1. M2 consists of Eqs. (5.3), (5.5) – (5.7), (5.17), (5.18b) – (5.19b) (instead of (5.18) and (5.19)), (5.20) – (5.23), (5.25) – (5.31). Since the objective function and constraints are all linear, M2 is an MILP model.

5.3 Approximating Univariate Functions

While the models in Section 5.2 are for fitting one-dimensional discrete data points, they can be extended to approximate univariate functions. One problem of interest is to obtain a continuous PWL function which does not deviate more than a user-defined tolerance, δ, from the function of interest over a compact interval; hence forth we call such PWL approximation a "δ-approximator". For a given univariate function, $\phi(x)$, over a finite interval $\mathbb{D}$, we minimize the number of segments needed to obtain a δ-approximator. We assume that $\phi(x)$ is continuously differentiable over $\mathbb{D}$, and we can a priori obtain all the local extrema in $\mathbb{D}$.

One of the biggest challenges in finding a δ-approximator is to ensure that the error does not exceed δ over a continuum, which leads to semi-infinite programming (SIP) problems [141,142]. Generalizing and extending the idea of Rebennack and Kallrath [134], we develop an iterative

procedure, which involves (1) evaluating the univariate function at discrete locations to obtain a set of data points, (2) finding the minimum number of break points by solving an MILP model with a given δ-tolerance, and (3) checking if a δ-approximator is truly obtained (thereafter referred to as δ-test). If the δ-test is not satisfied, then we insert new data points and repeat.

First, in terms of generating a set of points at which the function is evaluated, we adopt a modified equidistant discretization approach that exploits the nonlinear function to be approximated. Specifically, we find all the local extrema in $\mathbb{D}$, and divide $\mathbb{D}$ into sections $\mathbb{D}_s = [x_s^{lo}, x_s^{up}]$ for $s \in \mathbf{S}$, where x_s^{lo} and x_s^{up} are the x-locations of the pair of extrema that defines $\mathbb{D}_s$. For $s \in \mathbf{S}$, we introduce x_j^s for $j \in \mathbf{J}_s = \{1, 2, \dots, \theta_s\}$, where θ_s is the number of elements in this set. Note that θ_s is not only a function of the δ tolerance, but also affected by the first and second derivative of ϕ in section s. However, since θ_s will be updated in our iterative procedure (discussed later), it is recommended to use small θ_s initially and let the procedure find the optimal θ_s for $s \in \mathbf{S}$.

Using equidistant discretization, we calculate x_j^s as follows:

$$x_j^s = x_s^{lo} + (j-1)\left(x_s^{up} - x_s^{lo}\right)/(\theta_s - 1), j \in \mathbf{J}_s, s \in \mathbf{S} \tag{5.35}$$

As shown in Figure 5-7, $x_j^s \ \forall j \in \mathbf{J}_s, s \in \mathbf{S}$ are mapped onto x_i for $i \in \mathbf{I}$ in the correct order. At the boundaries of two sections, the last point in the previous section has the same x as the first point in the next section, and thus they are mapped to the same x_i. Therefore, the number of elements in set $\mathbf{I}$ is $|\mathbf{I}| = \sum_{s \in \mathbf{S}} \theta_s - |\mathbf{S}| + 1$.

Given (x_i, y_i) for $i \in \mathbf{I}$, we could solve M2 to find the δ-approximator. However, unlike fitting a finite data set, here the approximation must be valid throughout $\mathbb{D}$. We note that the approximation errors at the break points are likely to be relatively large due to the change of slope (see Figure 5-8A). Since errors at the break points are not considered in M2, it is likely that the δ-test fails at

the break points. To address this, we introduce a modified model, denoted as M3, to estimate the minimum number of break points for a given δ. Compared to M2, the additional assumption in M3 is that break points (x_k^B) must coincide with the x-coordinate of the last data point in k or the first data point in $k+1$. Note that with this additional assumption, Theorem 1 still holds, and the only modification required is to replace Eqs. (5.27) – (5.30) with the following constraints:

$$P_{i,k}^{+} \le v_i^U\left(1-U_{i,k}\right), i \in \mathbf{I},\ k \in \mathbf{K}^{IM} \tag{5.27b}$$

$$Q_{i+1,k+1}^{+} \le v_i^L\left(1-V_{i,k}\right), i \in \mathbf{I},\ k \in \mathbf{K}^{IM} \tag{5.28b}$$

$$P_{i,k}^{-} \le v_i^U\left(1-U_{i,k}\right), i \in \mathbf{I}, k \in \mathbf{K}^{IM} \tag{5.29b}$$

$$Q_{i+1,k+1}^{-} \le v_i^L\left(1-V_{i,k}\right), i \in \mathbf{I},\ k \in \mathbf{K}^{IM} \tag{5.30b}$$

which enforce that when i is the last point in k, either $P_{i,k}^{+}/P_{i,k}^{-}$ or $Q_{i+1,k+1}^{+}/Q_{i+1,k+1}^{-}$ must be zero.

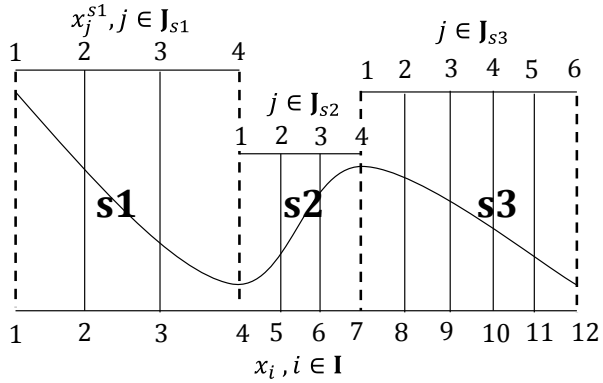


Figure 5-7. An example of how the data set is generated.

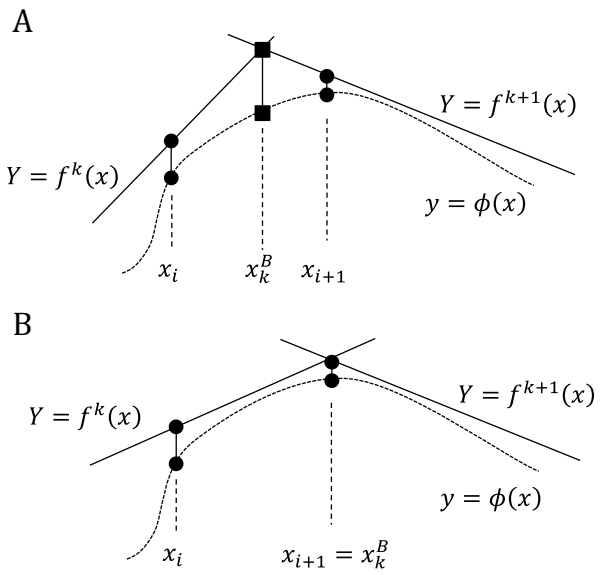


Figure 5-8. Deviation from the original function (dashed lines) at various locations.

By enforcing x_k^B to coincide with an x_i, we lose some degree of freedom in terms of finding potentially better approximations, however, the approximation at x_k^B is guaranteed to be within the δ-tolerance, which makes finding a δ-approximator more likely.

Next, we solve the following global optimization problems to check if a δ-approximator is obtained:

$$\varepsilon_k = \max_{\mathrm{x}\in\widetilde{\mathbb{D}}_k} |\phi(x) - f^k(x)|, k \in \mathbf{K}^I \tag{5.36}$$

where $f^k(x) = A_k x + B_k$ is the PWL function segment obtained by solving M3, and $\widetilde{\mathbb{D}}_k = [x_{k-1}^B, x_k^B]$. Although we have to solve $|\mathbf{K}^I|$ global optimization problems, we are interested in obtaining approximations of relatively "easy" $\phi(x)$ to be integrated in larger optimization problems. Thus, Eq. (5.36) is generally tractable. If $\varepsilon_k \leq \delta$ for all $k \in \mathbf{K}^I$, then a δ-approximator is obtained. Otherwise, we modify **I** by inserting additional data points.

To limit the size of the model at each iteration, we only insert new data points to the sections where the δ-test failed. We define subset $\mathbf{K}_s^I \subset \mathbf{K}^I$ such that $k \in \mathbf{K}_s^I$ if $\widetilde{\mathbb{D}}_k \cap \mathbb{D}_s \neq \emptyset$. We also define

subset $\mathbf{S}^U \subset \mathbf{S}$ such that $s \in \mathbf{S}^U$ if any $k \in \mathbf{K}_s^I$ fails the δ-test. Then, for $s \in \mathbf{S}^U$, we update the number of data points $\theta_s \leftarrow 2\theta_s - 1$. Note that we can use different multipliers to update θ_s, as long as they are integers that are greater than or equal to 2. It is also possible to use different multipliers for different $s \in \mathbf{S}^U$ based on the extent of violation of the δ tolerance. Nevertheless, using 2 as a multiplier is, in general, sufficient. For clarity we use superscript "(n)" to denote the sets/parameters/variables at the nth iteration. Thus, at iteration n, let $\boldsymbol{x}_s^{(n)} = \{x_j^s | j \in \mathbf{J}_s^{(n)}, s \in \mathbf{S}^{U(n)}\}$. If x_j^s are calculated from Eq. (5.35) and $\theta_s^{(n+1)} = 2\theta_s^{(n)} - 1$ for $s \in \mathbf{S}^U$, then $\boldsymbol{x}_s^{(n)} \subset \boldsymbol{x}_s^{(n+1)}$ for $s \in \mathbf{S}$. In other words, the set of data points from the previous literation is contained in the data set of the next iteration. Therefore, if z^* is the optimal objective of M3 in the current iteration, then z^* is a valid lower bound on the objective of M3 in the next iteration.

Proposition 3. If $z^{*(n)}$ is the optimal objective of M3 at iteration n, and $z^{(n+1)}$ is the objective value of any feasible solution of M3 at iteration $n+1$, then $z^{*(n)} \leq z^{(n+1)}$.

Proof. The proof is by contradiction. Assume there exists $z^{(n+1)} < z^{*(n)}$ such that M3 is feasible at the $(n+1)$th iteration, which means that the δ-tolerance is satisfied at every data point. Since the set of data points in iteration $n+1$ contains the data set in iteration n, the δ-tolerance is also satisfied for every data point in the nth iteration. It means that there exists a solution at the nth iteration whose objective function value is strictly smaller than $z^{*(n)}$, contradicting the fact that $z^{*(n)}$ is optimal. □

The procedure for finding the minimum number of break points for a univariate function is summarized in Algorithm 1. It is worth noting that while Algorithm 1 will terminate within a finite number of iterations, it does not guarantee that the obtained PWL function satisfies the δ-test. If the violation is small, then a work-around is to rerun the algorithm with a slightly stricter tolerance.

By doing so, the solutions obtained might be conservative (i.e., overestimating the number of segments required), but the original δ tolerance may be satisfied.

Algorithm 1. Finding a δ-approximator for a univariate function

0: Given nonlinear function ϕ: $\mathbb{D} \to \mathbb{R}$, all the local extrema ($\boldsymbol{x}^E$) within $\mathbb{D}$, and tolerance δ
1: based on $\boldsymbol{x}^E$, divide $\mathbb{D}$ into sections $\mathbb{D}_s$ for $s \in \mathbf{S}$, and compute x_j^s according to Eq. (5.35)
2: **while** resource limit is not reached
3: map x_j^s, $j \in \mathbf{J}$ and $s \in \mathbf{S}$ to x_i for $i \in \mathbf{I}$, and compute $y_i = \phi(x_i)$ for $i \in \mathbf{I}$.
4: solve M3 to obtain z^*, the optimal objective function value
5: solve Eq. (5.36) for all $k \in \mathbf{K}^I$ (i.e., δ-test)
6: **if** $\varepsilon_k \leq \delta$ for all $k \in \mathbf{K}^I$
7: a δ-approximator with z^* segments is found. STOP.
8: **else**
9: update the lower bound of z as z^* in the next iteration
10: for $s \in \mathbf{S}^U$, set $\theta_s \leftarrow 2\theta_s - 1$, and update x_j^s for all $s \in \mathbf{S}^U$
11: **end if**
12: **end while**

5.4 Illustrative Example

we compare the computational performance of the proposed linear model (M1) against the nonlinear benchmark model (M0) using 17 different instances. Each instance is a unique combination of different data sets and number of break points. The data sets are chosen from a wide range of fields: from physical sciences and engineering to social sciences. Therefore, the curvature of the underlying functions is substantially different, and so are the number of data points in each set. We use $q = 2$ and we set a CPU time limit of 1800 seconds. M1 is solved with CPLEX, while M0 is solved using three state-of-the-art global optimization solvers (BARON, ANTIGONE, and SCIP). For each instance, only the best solution out of the three is used for comparison. The computational performance is shown in Figure 5-9 using a performance chart. Clearly, compared to M0, M1 is significantly faster and can solve much more instances. We also solve M1 with a single sufficiently large big-M parameter (10^3) instead of using tight variable bounds such as $\underline{A}$

and $\overline{A}$ and tight big-M parameters as discussed in Section 5.2. Computational results imply that the proposed bounds and parameters improve the solution performance significantly.

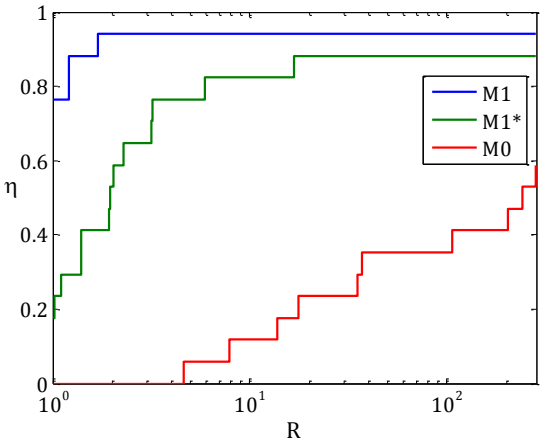


Figure 5-9. Performance chart comparing M0 (red line), M1 with tight big-M parameters and tight variable bounds (blue line), and M1 with a large big-M parameter (green line, M1*).

5.5 Conclusions

In this chapter, we first developed a series of theoretical results that allow us to formulate the problem of fitting a PWL function to data points as a mixed-integer lineal model. Specifically, we derived a condition for the PWL function to be continuous and proposed bounds on variables such as A_k, B_k, and Y_i, allowing us to reformulate the nonlinear continuity constraint through a set of mixed integer linear constraints. Second, we developed three MILP and MIQCP models: (1) model (M1) with convex objective function and linear constraints for error minimization with given number of break points; (2) MILP model (M2) that allows finding the minimum number of break points required for a given error tolerance; and (3) MILP model (M3) that minimizes the number of break points for a given error limit. We showed that, compared against previously proposed approaches that utilize nonlinear constraints to enforce continuity, our models find solutions that are at least as good and prove optimality significantly faster. Finally, we proposed an iterative

procedure to approximate continuous univariate functions with a PWL function having the minimum number of segments given a bound on the pointwise approximation error.

Chapter 6
McCabe-Thiele-inspired Math Program for Distillation Column Design[8]

6.1 Motivation

Distillation is arguably the most important unit operation in the separation subsystem of chemical processes. The design and optimization of distillation columns have drawn considerable attention in the chemical engineering literature for more than one hundred years. A simple yet accurate distillation model is crucial for higher-level modeling such as distillation sequencing [59,143-150], separation network synthesis [151,152], and general process synthesis [90,153,154]. Methods for modeling distillation columns are discussed in many textbooks [31,155,156] and review papers [157-159].

As previously mentioned, there are three types of approaches, namely, shortcut, rigorous, and graphical methods. While equation-based shortcut methods such as the Underwood equations are readily applicable to mathematical programming-based models, they are based on various assumptions and simplifications such as the approximation of the VLE with constant relative volatility across the entire column. Moreover, shortcut methods usually focus on the targeting of either the energy demand (e.g., minimum reflux ratio) or the capital cost (e.g., minimum number of trays), but overlook the energy-capital tradeoff and the effect of column operations (e.g. reflux ratio) on design decisions (e.g. number of trays). On the other hand, rigorous approaches rely on minimum assumptions and detail property models to model the distillation column rigorously. However, they can be computationally intractable due to the use of complex thermodynamic

[8] The contents of this chapter appear in Kong and Maravelias, *AIChE J.,* 2019.

models and the VLE calculation at every tray. Finally, while graphical methods have some important limitations, such as limited degrees of freedom, they also have two important advantages: (1) they are intuitive and offer significant insights, and (2) "complex thermodynamics" can be readily represented graphically. As a result, they are still used nowadays for conceptual design or for preliminary analysis for a more rigorous design. The natural question then becomes: can we develop flexible mathematical-programming-based approaches while maintaining the advantages of graphical methods? In fact, extracting the concepts from graphical approaches to develop flexible optimization models has been successfully implemented in other areas. For example, in heat integration, the composite curves, originally introduced in pinch analysis, can be represented mathematically in an optimization model for energy [48,55] and heat exchanger area targeting [97,160,161]. However, the same has not been done for distillation column design and optimization. Therefore, the goal of this work is to propose a mixed-integer nonlinear programming (MINLP) model for distillation column design based on the concepts and equations underpinning the McCabe-Thiele method as well as ideas used in rigorous methods.

6.2 Derivation of McCabe-Thiele Method

The first scientific method for binary distillation, proposed by Sorel [162], utilizes material and energy balances as well as equilibrium relationships. Subsequently, Lewis [163] simplified the calculation by eliminating energy balances. Based on the equations from Lewis' method, McCabe and Thiele developed their graphical method. The major assumption made in all these methods is that the molar liquid and vapor flows are constant in the rectifying section (section of column above the feed) as well as in the stripping section (section of column below the feed). This

assumption is referred to as the *constant-molar overflow* assumption. The assumption holds if the following conditions are true:

(1) Heat losses from the column are small

(2) The molar latent heat of vaporization of the two components are not substantially different, and

(3) The boiling point of liquid does not change substantially across the column

The first condition is generally valid since most columns are well-insulated; the second and third conditions can be justified because, in many cases, we are interested in separating molecules with similar physical properties (otherwise, the separation would be easy and distillation is not necessary for separating the mixture). According to McCabe and Thiele, the violations to these conditions are usually small and tend to balance each other out. Our approach inherently adopts the same assumptions. In a later section, we will show that the proposed model can be modified to account for situations where some of these conditions are not valid.

Under the constant-molar overflow assumption, the flow pattern inside a column is represented in Figure 6-1. Specifically, V_1 and L_1 represent the constant vapor and liquid flow rates in the rectifying section, while V_2 and L_2 represent the vapor and liquid flow rates in the stripping section. Since a binary mixture is assumed, the composition at each tray can be readily represented by the mole fraction of the relatively light component (hereon denoted as composition). At the nth tray, Y_n and X_n represent the vapor and liquid composition, respectively.

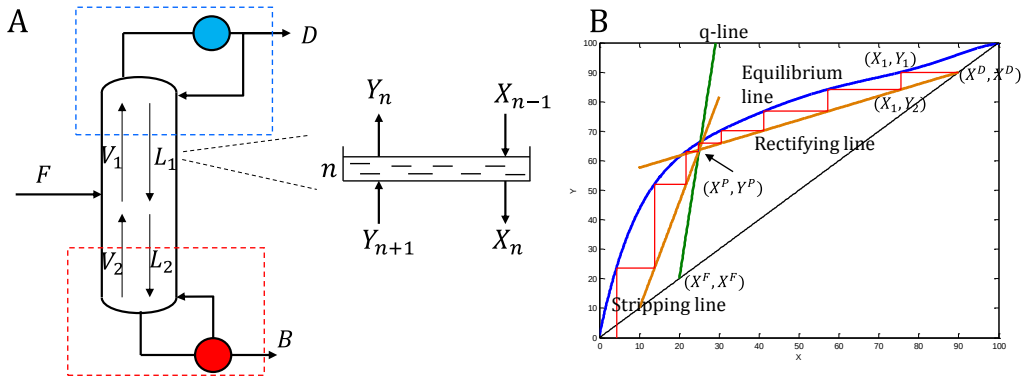


Figure 6-1. (A) A distillation column for separating a binary mixture; (B) a typical McCabe- Thiele diagram.

The McCabe-Thiele graphical method is derived from the following material balances introduced by Sorel [162]. First, overall and component material balances are written around the column,

$$F = D + B \tag{6.1}$$

$$FX^F = DX^D + BX^B \tag{6.2}$$

where F, D, and B are molar flow rate of the feed, distillate, and bottom liquid stream, and X^F, X^D, and X^B are mole fraction of the light component in the corresponding stream.

With the constant-molar overflow assumption, the liquid and vapor flow rates remain constant from one tray to the next in each column section. Therefore, in the rectifying section, an overall material balance is introduced to relate the liquid and vapor flow rates with the distillate flow rate:

$$V_1 = L_1 + D \tag{6.3}$$

Then component material balances are written over an envelope for the condenser and tray 1 to n for $n \in \mathbf{N}^R$ (blue box in Figure 6-1A),

$$V_1 Y_{n+1} = L_1 X_n + DX^D \qquad n \in \mathbf{N}^R \tag{6.4}$$

where $\mathbf{N}^R$ is the set of trays in the rectifying section.

In the stripping section, an overall material balance (red box in Figure 6-1A) relates the liquid and vapor flow rates to the bottom liquid flow rate and component material balances are written from the tray $n \in \mathbf{N}^S$ to the reboiler of the column:

$$V_2 = L_2 - B \tag{6.5}$$

$$V_2 Y_{n+1} = L_2 X_n - B X^B \qquad n \in \mathbf{N}^S \tag{6.6}$$

where $\mathbf{N}^S$ is the set of trays in the stripping section.

Finally, at the intersection of the rectifying and stripping sections where the feed is introduced, the corresponding liquid and vapor compositions (X^P and Y^P) are determined via material balances. Eq. (6.7) corresponds to the blue dashed box in Figure 6-2 and Eq. (6.8) corresponds to the red dashed box,

$$V_1 Y^P = L_1 X^P + D X^D \tag{6.7}$$

$$V_2 Y^P = L_2 X^P - B X^B \tag{6.8}$$

which represent the component material balances written around the rectifying section and around the stripping section of the column.

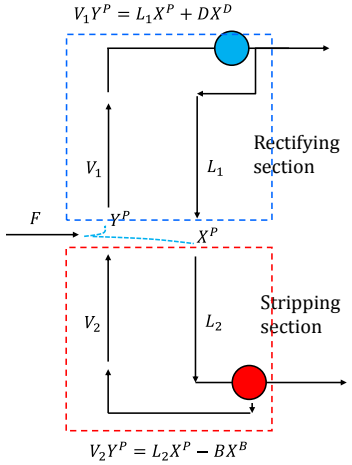


Figure 6-2. Material balances from feed to top and bottom of the distillation column.

McCabe and Thiele assume that the flow rate and composition of feed, distillate, and bottom liquid are given a priori. Thus, the rectifying operating line can be derived from Eqs. (6.3) and (6.4), and the stripping operating line can be derived from Eqs. (6.5) and (6.6). These straight lines relate the composition of vapor entering and liquid leaving a tray (X_n and Y_{n+1}). The equilibrium line (blue line in Figure 6-1B), on the other hand, specifies the composition relationship between the liquid and vapor leaving a tray (X_n and Y_n). Therefore, given a composition at the top ($Y_1 = X^D$), the liquid and vapor composition of each tray can be determined by "stepping down" from the first tray as illustrated in Figure 6-1B. Due to the discrete nature of the procedure, one cannot guarantee that the bottom composition obtained graphically will match the specification (i.e. X^B), which means that some iterations are needed. However, we will show that these iterations can be avoided in the proposed method. The q-line (green line in Figure 6-1B) that connects the feed to the intersection of the two operating lines can be derived from Eqs. (6.7) – (6.8). For detailed derivation, readers are referred to various textbooks or the original paper by McCabe and Thiele [75]. As the foundations of the McCabe-Thiele graphical method, Eqs. (6.1) – (6.8) are critical in the derivation of the proposed model for simple columns, which will be introduced next.

6.3 Proposed Model

The proposed model is comprised of (1) a set of mixed-integer constraints that specify the column sections and determine the location of the feed tray; (2) material balances similar to those used to derive the McCabe-Thiele method, and (3) equilibrium relationship that relates the liquid and vapor composition at each tray. In this section, we focus on simple distillation columns with one feed and two product streams, while the design of complex columns with multiple feeds and side streams will be discussed later.

6.3.1 Column Sections

In our model, unless specified, the number of trays is a decision variable. Therefore, similar to the rigorous approach[71], we define a set **N** that contains sufficiently many elements. A subset of **N** will be selected by optimization to represent the trays in the column. For the selected trays, we also need to determine which section of the column they belong to since some material balances (e.g., Eqs. (6.4) and (6.6)) require the membership of $\mathbf{N}^R$ and $\mathbf{N}^S$ to be known. However, $\mathbf{N}^R$ and $\mathbf{N}^S$ depend on the location of the feed tray, which cannot be identified until the model is solved. Therefore, we introduce sections $k \in \mathbf{K} = \{1,2,3\}$ in which "1" and "2" represent the rectifying and stripping sections, respectively, while "3" is an auxiliary section for the unselected trays. In particular, the first element in the auxiliary section represents the reboiler, which is modeled as an equilibrium tray. We introduce a binary variable $U_{k,n}$, which is one if tray n belongs to section k. A tray should belong to only one section,

$$\sum_k U_{k,n} = 1 \qquad n \in \mathbf{N} \qquad (6.9)$$

In our convention, the tray number increases from top to bottom of the column, and thus in set **N**, the first r elements represent trays in the rectifying section, the next s elements are trays in the stripping section, and the last t elements represent trays that are not selected (Figure 6-3). Note that in the McCabe-Thiele method, the feed location is implicitly determined via material balances, and thus the exact feed location should be somewhere between the last tray in the rectifying section and the first in the stripping section. Therefore, in our model, while the "optimal feed tray" refers to the first tray in the stripping section, the actual feed location can be on or above that tray.

Since X^P and Y^P represent the liquid and vapor composition, respectively, at the exact feed location, they constrain the liquid and vapor composition of trays in each column section:

$$X^P - (1 - U_{1,n}) \leq X_n \leq X^P + (1 - U_{2,n}) \qquad n \in \mathbf{N} \qquad (6.10)$$

$$Y^P - \left(1 - U_{1,n}\right) \leq Y_{n+1} \leq Y^P + (1 - U_{2,n}) \qquad n \in \mathbf{N} \qquad (6.11)$$

For example, if tray n' belongs to the rectifying section, $U_{1,n'} = 1$, then Eq. (6.11) enforces that $X_{n'} \geq X^P$ while Eq. (6.12) enforces that $Y_{n'+1} \geq Y^P$. This can be seen from the McCabe-Thiele diagram (Figure 6-1B) in which point P is always on the lower left of the intersecting points of the red line segments and the rectifying line.

The vapor and liquid composition for trays in the stripping section should be no smaller than the bottom liquid composition (i.e., X^B),

$$X^B \leq X_n + \left(1 - U_{2,n}\right) \qquad n \in \mathbf{N} \qquad (6.12)$$

$$Y^B \leq X_n + \left(1 - U_{2,n}\right) \qquad n \in \mathbf{N} \qquad (6.13)$$

For trays that are not selected ($U_{3,n} = 1$), we enforce the liquid composition to be equal to X^B:

$$X_n - \left(1 - U_{3,n}\right) \leq X^B \leq X_n + \left(1 - U_{3,n}\right) \qquad n \in \mathbf{N} \qquad (6.14)$$

which, physically, means that the liquid composition in the reboiler should be equal to X^B. The vapor composition of the first tray is equal to the distillate composition,

$$Y_1 = X^D \qquad (6.15)$$

Through Eqs. (6.14) and (6.15), the proposed model ensures that the vapor composition of the first tray and the liquid composition of the reboiler are equal to the top and bottom product specifications (i.e. X^D and X^B), which is another clear advantage over its graphical counterpart where an iterative procedure might be needed to ensure the consistency of compositions between the solution and the specifications.

The composition profile must follow:

$$X_n \geq X_{n+1} \qquad n \in \mathbf{N} \qquad (6.16)$$

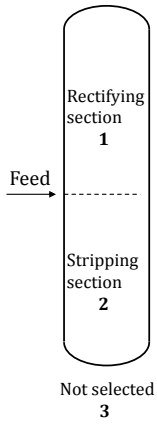
Although not necessary, a similar constraint can be written for the vapor phase as well.

6.3.2 Material Balances

The proposed model includes the material balances introduced in the previous section with some modifications. The membership of $\mathbf{N}^R$ and $\mathbf{N}^S$ is implicitly determined using binary variables, and thus Eqs. (6.4) and (6.6) can be reformulated as follows:

$$V_1 Y_{n+1} - L_1 X_n - DX^D = S^P_{1,n} - S^N_{1,n} \qquad n \in \mathbf{N} \qquad (6.4b)$$

$$V_2 Y_{n+1} - L_2 X_n + BX^B = S^P_{2,n} - S^N_{2,n} \qquad n \in \mathbf{N} \qquad (6.6b)$$



$U_{1,n}$	$U_{2,n}$	U_{3n}
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
0	0	1
0	0	1

Figure 6-3. Illustrative example of binary variable $U_{k,n}$.

$$S^P_{k,n} \leq \mu_k (1 - U_{k,n}) \qquad k \in \{1,2\}, n \in \mathbf{N} \qquad (6.17)$$

$$S^N_{k,n} \leq \mu_k (1 - U_{k,n}) \qquad k \in \{1,2\}, n \in \mathbf{N} \qquad (6.18)$$

where $S^P_{k,n}$ and $S^N_{k,n}$ are nonnegative slack variables that are set to zero when binaries $U_{k,n} = 1$, and μ_k is a sufficiently large positive number for each k. When tray n is in the rectifying section

($U_{1,n} = 1$), both $S_{1,n}^{P}$ and $S_{1,n}^{N}$ are equal to zero and thus the right-hand-side (RHS) of Eq. (6.4b) is zero, thereby enforcing the material balance in Eq. (6.4). Similarly, when $U_{2,n} = 1$, the right-hand-side of Eq. (6.6b) is zero so that the material balance in the stripping section is enforced.

Under the constant-molar overflow assumption, the difference of flow rate between the rectifying and stripping section is due to the feed,

$$L_2 - L_1 = qF \tag{6.19}$$

where q is a parameter that specifies the feed condition, similar to the q in the McCabe-Thiele method that determines the slope of the "q-line". Specifically, when the temperature of the feed is between its dew point and bubble point, q is the liquid mole fraction of the feed; when the feed temperature is above the dew point or below the bubble point, q is equal to the heat required to vaporize one mole of feed over the molar latent heat of vaporization. For example, when the feed is saturated liquid ($q = 1$), according to Eq. (6.19), the liquid flow rate in the stripping section is the summation of the liquid flow rate in the rectifying section and the feed; whereas when the feed is a superheated vapor whose temperature is above its dew point ($q < 0$), some liquid from the stripping section will be vaporized and thus $L_2 < L_1$.

Conceptually, if "q" fraction of the feed joins the down-flowing liquid, then the remaining "$1 - q$" portion of the feed will be added to the ascending vapor:

$$V_1 - V_2 = (1 - q)F \tag{6.19b}$$

However, Eq. (6.19b) is redundant since it can be easily derived from Eqs. (6.1), (6.3), (6.5) and (6.19).

6.3.3 VLE Model

Since we assume that the liquid and vapor at each tray are in equilibrium, X_n and Y_n satisfy function $Y = f^E(X)$, which is the mathematical equivalence of the equilibrium line in the graphical approach. There are many ways to obtain this function. For ideal systems, it can be directly derived from the constant relative volatility: $Y = \alpha X/(1 + (\alpha - 1)X)$, where α is the relative volatility of the mixture.

For non-ideal systems, more accurate equilibrium relationship can be obtained from the K values that relate the liquid and vapor composition at each tray, which are calculated from thermodynamic models such as the Margules [164] and Van Laar equations [165], or other equations of state. However, using these highly nonlinear and nonconvex thermodynamic equations significantly affects the tractability and robustness of the resulting optimization models. Accordingly, we model the VLE relationship with continuous piecewise linear approximating (PWLA) functions, which eliminate all the nonlinearities in terms of thermodynamics.

For a given mixture, a set of VLE data is obtained experimentally or from rigorous thermodynamic models such as the UNIFAC method. Then, the data points are fitted by a continuous PWLA function via a mixed-integer linear programming (MILP) model [138], which finds the minimum number of segments necessary to satisfy a given error tolerance. Interestingly, many state-of-the-art VLE models are constructed from fitting experimental data points. However, highly nonlinear and complex functions (e.g. cubic functions and logarithmic functions) are needed to capture the non-ideal behaviors.

Here we can develop our own VLE model for the mixture of interest using a PWLA function to directly approximate experimentally obtained data points. This allows us to model any mixture

with arbitrary non-ideality (i.e. any "shape" of VLE line) using a PLWA function with a few segments. The difficulty of our approximation would not change significantly from one mixture to another.

Using the surrogate thermodynamic model, we can balance accuracy with complexity. For example, we can fit the thermodynamic data set to an arbitrary accuracy using the fewest number of function segments possible. In the fitting model, we can introduce constraints to enforce thermodynamics-based constraints. For example, a constraint is added to enforce positive slope of each segment because the VLE curve must have strictly positive slope [166]. For zeotropic distillation, the PWLA function must pass through points (0,0) and (1,1), which is enforced via additional constraints in the fitting model. For azeotropic systems, the PWLA function is constrained to pass through the azeotrope and point (0,0) or (1,1).

Moreover, care is taken when approximating the VLE near the azeoptrope as small difference in approximation can lead to very different solutions. To illustrate, four experimentally obtained data points near the azeotrope of water-ethanol system [167] are plotted in Figure 6-4. As shown in the figure, the underlying VLE function appears to be tangent to the 45-degree line near the azeotrope and thus the linear approximating function (blue line in Figure 6-4) would easily lead to a significant underestimation of the number of trays and overestimation of the reflux ratio. To resolve this, we enforce a stricter error tolerance for the data points near the azeotrope. The red line in Figure 6-4 is obtained by enforcing zero error for the first two data points (a1 and a2) below the azeotrope. Compared to the blue line, the red line better represents the VLE near the azeotrope, which is crucial for an accurate solution. However, note that doing this might increase the number of segments needed and thus lead to larger optimization models.

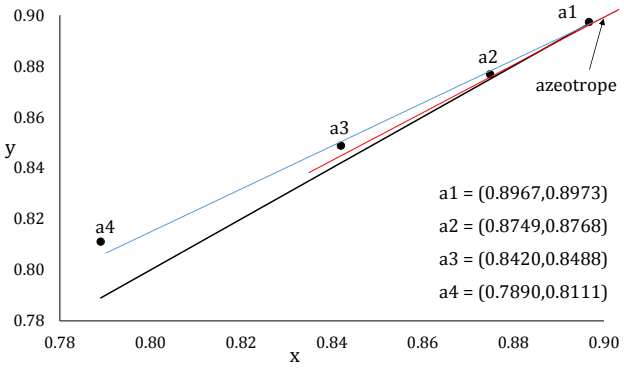


Figure 6-4. Ethanol-water VLE at 50.66 kPa. The black line represents the 45-degree line.

Once the PWLA function is determined offline, it is modeled using a set of mixed-integer linear constraints and integrated into the distillation model. While there are different approaches to model a PWLA function [136], we present the following formulation as an example.

Given a set of PWLA function segments $j \in \mathbf{J}$, we introduce a binary variable $Z_{j,n}$, which is equal to one if and only if tray n belongs to segment j. Each tray must belong to one segment:

$$\sum_j Z_{j,n} = 1 \qquad n \in \mathbf{N} \qquad (6.20)$$

We introduce a new continuous variable $X^D_{j,n}$, which is the disaggregation of X_n into segments:

$$\sum_j X^D_{j,n} = X_n \qquad n \in \mathbf{N} \qquad (6.21)$$

where $X^D_{j,n}$ is subject to the following constraint:

$$x^{lo}_j Z_{j,n} \le X^D_{j,n} \le x^{up}_j Z_{j,n} \qquad j \in \mathbf{J}, n \in \mathbf{N} \qquad (6.22)$$

and x^{lo}_j and x^{up}_j are lower and upper x limits of segment j that are determined by the fitting problem. Let m_j and b_j be the slope and intercept of j also obtained from the fitting problem, then Y_n is related to X_n as follows:

$$Y_n = \sum_j (m_j X^D_{j,n} + b_j Z_{j,n}) \qquad n \in \mathbf{N} \qquad (6.23)$$

6.3.4 Objective Functions

A wide range of objective functions can be chosen. In general, the objective is to minimize a cost of interest (Ψ), which can include contributions from the number of trays, vapor and liquid flow rates, reboiler and condenser duty and their capital cost. Therefore, the proposed model (𝕄1) for simple distillation columns is as follows:

$$\min \Psi \qquad \text{(M1)}$$

$$s.t. \quad \text{Eqs.}\,(6.1) - (6.23)$$

Our model provides a highly flexible approach to design and optimize a distillation column because some specifications can be defined as variables to be optimized. It is completely equation oriented (no need to define input/output variables), allowing us to go beyond the graphical design method to formulate, for example, an "operational" problem where the number of trays is fixed and the product purity/recovery are optimized. Unlike its graphical counterpart, the decision variables include not only column design specifications (e.g. number of trays) and operating conditions (e.g. reflux ratio and heat duty), but also flowrate and composition of streams entering (e.g., feed) and leaving (e.g., top and bottom) the column, enabling the proposed model to be used as a sub-model for optimization-based process synthesis. In addition, the proposed model-based approach can handle distillation optimization with many degrees of freedom, which is difficult for other approaches such as selectivity analysis.

6.3.5 Remarks

The material balances introduced are intended for a conventional column with a total condenser at the top, a reboiler at the bottom, and a feed introduced somewhere in the column. However, these material balances can be trivially modified to account for "nonconventional" columns such as a

column with a partial condenser or a feed directly introduced to the bottom of column. For example, when a partial condenser is used, it can be simply approximated as an additional equilibrium tray so that Y_1 and X_1 represent the vapor and liquid composition of the condenser.

The proposed model can predict the minimum number of trays (N_{min}), which corresponds to a column operating at total reflux. To obtain N_{min}, $\mathbb{M}1$ is solved with an objective of minimizing number of trays. In addition, the feed flow rate is set to equal to zero and both V_1 and L_1 are set to be equal to an arbitrary nonzero value. Since F, D, and B are all zero, the liquid and vapor flow rates are constant across the column ($V_1 = V_2 = L_1 = L_2$). N_{min} can be used to estimate the size of the number of elements in **N** in model $\mathbb{M}1$.

The computational performance of $\mathbb{M}1$ can be improved by first obtaining the minimum reflux ratio, R_{min}. If the PWLA function of the VLE line is concave (see Figure 6-5A), then the minimum reflux ratio corresponds to the rectifying line passing the intersecting point of the VLE line and the q-line (X^{PC}, Y^{PC}). In this case, R_{min} can be precalculated if X^D and X^F are given. However, if the PWLA function is not concave (e.g. for azeotropic mixtures), then it is possible that the rectifying line is tangent with the VLE line somewhere else (Figure 6-5B). To find R_{min}, we introduce a new optimization model, $\mathbb{M}2$, as follows:

$$\min R \qquad \text{(M2)}$$

$$s.t. \quad f^E(X^{PC}) \geq \frac{R}{R+1}X + \frac{X^D}{R+1}$$

$$f^E(X^{PC}) = \frac{q}{q-1}X^{PC} - \frac{X^F}{q-1}$$

$$m_j x_j^{up} + b_j \geq \frac{R}{R+1}x_j^{up} + \frac{X^D}{R+1} \qquad j \in \mathbf{J}^P$$

where $\mathbf{J}^P = \{j | x_j^{up} > X^{PC}, j \in \mathbf{J}\}$. $\mathbb{M}2$ yields the minimum reflux ratio that corresponds to the true pinch point of the PWLA function and the rectifying line.

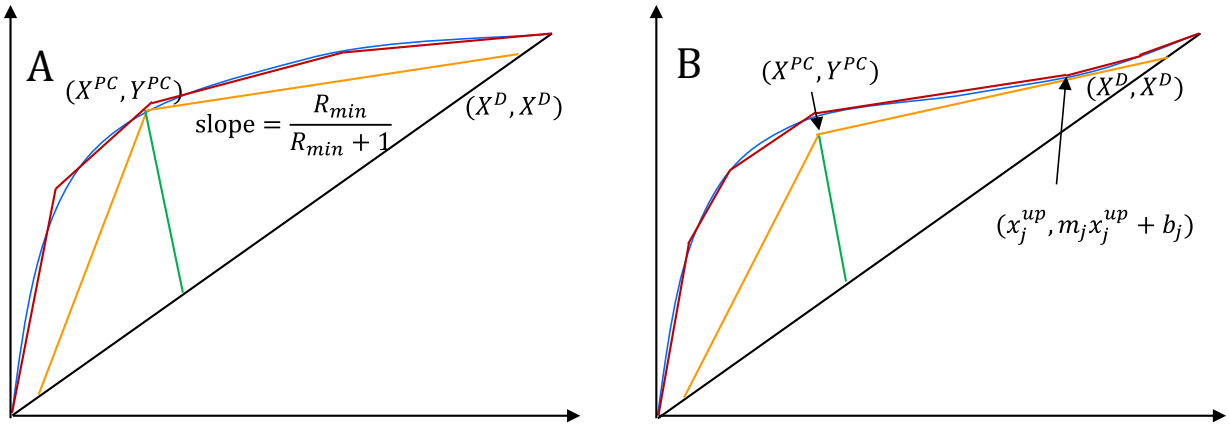


Figure 6-5. Distillation column operation corresponds to minimum reflux ratio.

Finally, to account for the fact that vapor and liquid are not completely in equilibrium, we can model tray efficiency using various tray efficiency calculation methods. For instance, we add the Murphree efficiency [168] for the vapor phase:

$$\eta = (Y_n - Y_{n+1})/(Y_n^* - Y_{n+1}) \qquad n \in \mathbf{N} \qquad (6.24)$$

where $Y_n^* = f^E(X_n)$ is the vapor composition that would be in equilibrium with the liquid composition leaving tray n. Y_n and Y_{n-1} are the average actual vapor composition leaving trays n and $n-1$, respectively.

6.4 Complex Column

For a simple column with one feed, the McCabe-Thiele diagram includes a rectifying operating line and a stripping operating line. For distillation columns with multiple feeds and side streams, one additional operating line will be added for each additional stream. Similarly, in the proposed approach, additional feeds and side streams lead to additional column sections. A column with p

feeds and q side streams result in $p + q + 1$ sections (see Figure 6-6). The boundary of two neighboring sections is defined by either a feed or a side stream.

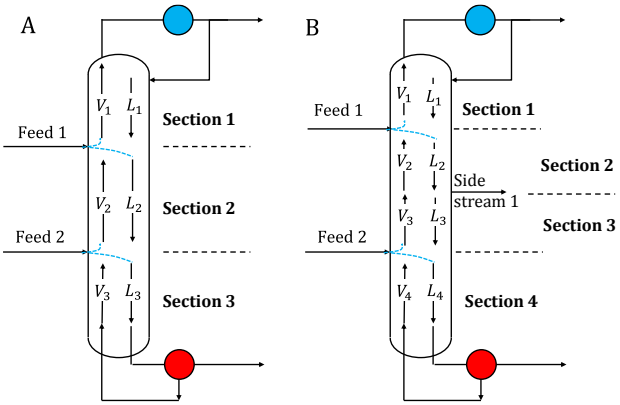


Figure 6-6. (A) A three-section column with two feeds; (B) a four-section column with two feeds and one side stream.

For columns with multiple feeds and side streams, an additional challenge comes from the sequencing of feeds and side streams from top to bottom. The optimal sequence depends not only on the compositions and quality of the feeds and side streams, but also on other optimization decisions such as the reflux ratio. For instance, in Figure 6-7, feed F1 is saturated vapor with $X_1^F = 0.6$, and feed F2 is saturated liquid with $X_2^F = 0.4$. At large reflux and reboil ratios, the operating lines are close to the 45-degree line and F1 should be placed above F2; while at smaller reflux and reboil ratios, the operating lines move towards the VLE line and F1 should be placed below F2. Therefore, it is important to develop an optimization model that can determine the optimal sequence of the feeds and side streams based on their composition, quality, and other optimization decisions.

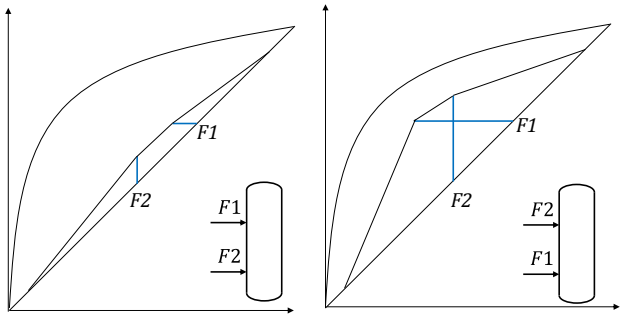


Figure 6-7. Two possible column designs with the same feed specifications.

The model in this subsection is an extension and generalization of $\mathbb{M}1$ to consider (1) multiple feeds and side streams, and (2) unknown sequencing of feeds and side streams. Given a column with p feeds and q side streams, we define the set of feeds as $l \in \mathbf{L} = \{1,2, ...,$ and the set of side streams as $m \in \mathbf{M} = \{1,2, \dots, q\}$. Therefore, the number of elements in set **K** is $\tau = p + q + 2$. A set **N** is introduced to represent the tray candidates and Eq. (6.9) is included to specify the relationship between section k and tray n. Eq. (6.16) is also included to enforce the composition relationship. Since the boundaries of the first $p + q$ column sections are defined by feeds or side streams, it is critical to assign feeds/side streams to the proper boundaries so that the correct material balances can be applied. We introduce section boundaries $k \in \mathbf{K}^P = \{1,2, \dots, \tau - 2\}$ that are defined by either feeds or side streams.

New binary variables, $Z^F_{l,k}$ and $Z^S_{m,k}$, are introduced so that $Z^F_{l,k} = 1$ if feed l is assigned to boundary k and $Z^S_{m,k} = 1$ if side stream m is assigned to boundary k. Each feed and side stream can only be assigned to one boundary:

$$\sum_{k \in \mathbf{K}^P} Z^F_{l,k} = 1 \qquad l \in \mathbf{L} \qquad (6.25)$$

$$\sum_{k \in \mathbf{K}^P} Z^S_{m,k} = 1 \qquad m \in \mathbf{M} \qquad (6.26)$$

The section boundaries are defined by either a feed or a side stream:

$$\sum_l Z^F_{l,k} + \sum_m Z^S_{m,k} = 1 \qquad k \in \mathbf{K}^P \qquad (6.27)$$

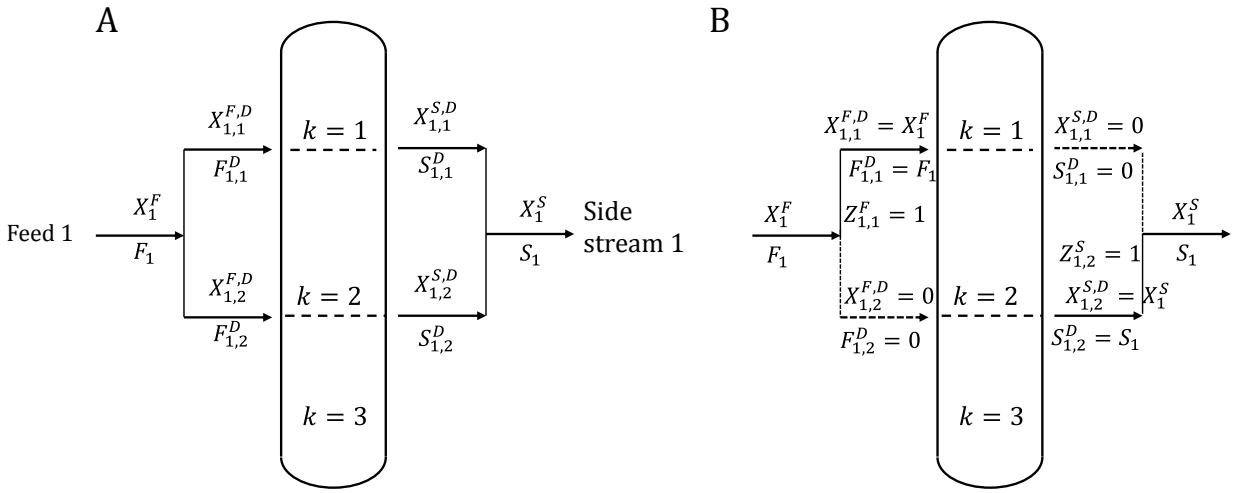


Figure 6-8. An example of a distillation column with one feed and one side stream with unknown sequence.

Then, the feed flow rates (F_l) and compositions (X^F_l) are disaggregated and (de)activated by $Z^F_{l,k}$:

$$F_l = \sum_{k \in \mathbf{K}^P} F^D_{l,k} \qquad l \in \mathbf{L} \qquad (6.28)$$

$$X^F_l = \sum_{k \in \mathbf{K}^P} X^{F,D}_{l,k} \qquad l \in \mathbf{L} \qquad (6.29)$$

$$F^D_{l,k} \leq \gamma_l Z^F_{l,k} \qquad l \in \mathbf{L},\ k \in \mathbf{K}^P \qquad (6.30)$$

$$X^{F,D}_{l,k} \leq Z^F_{l,k} \qquad l \in \mathbf{L}, k \in \mathbf{K}^P \qquad (6.31)$$

where $F^D_{l,k}$ and $X^{F,D}_{l,k}$ are disaggregated feed flow rate and composition, respectively. Parameter γ_l is an upper bound on the feed flow rate. Similarly, the side stream flow rate (S_m) and composition (X^S_m and Y^S_m) are also disaggregated and (de)activated by $Z^S_{m,k}$,

$$S_m = \sum_{k \in \mathbf{K}^P} S^D_{m,k} \qquad m \in \mathbf{M} \qquad (6.32)$$

$$X^S_m = \sum_{k \in \mathbf{K}^P} X^{S,D}_{m,k} \qquad m \in \mathbf{M} \qquad (6.33)$$

$$Y^S_m = \sum_{k \in \mathbf{K}^P} Y^{S,D}_{m,k} \qquad m \in \mathbf{M} \qquad (6.34)$$

$$S^D_{m,k} \leq \gamma_m Z^S_{m,k} \qquad m \in \mathbf{M}, k \in \mathbf{K}^P \qquad (6.35)$$

$$X^{S,D}_{m,k} \leq Z^S_{m,k} \qquad m \in \mathbf{M}, k \in \mathbf{K}^P \qquad (6.36)$$

$$Y^{S,D}_{m,k} \leq Z^S_{m,k} \qquad m \in \mathbf{M}, k \in \mathbf{K}^P \qquad (6.37)$$

where $S^D_{m,k}$, $X^{S,D}_{m,k}$, and $Y^{S,D}_{m,k}$ are disaggregated side stream flow rate, liquid composition, and vapor composition, respectively. Figure 6-8 provides an example of mapping one feed and one side stream to two boundaries via variable disaggregation. In the example, feed 1 is mapped to boundary 1 ($Z^F_{1,1} = 1$), and thus, according to Eqs. (6.28) – (6.31), $F^D_{1,1} = F_1$ and $X^{F,D}_{1,1} = X^F_1$. On the other hand, side stream 1 is mapped to boundary 2 ($Z^S_{1,2} = 1$) so that, from Eqs. (6.32) – (6.37), $S^D_{1,2} = S_1$ and $X^{S,D}_{1,2} = X^S_1$. In this way, the variables from the feeds and side streams are connected to the disaggregated variables based on the binaries. The disaggregated variables are used in the material balances that will be introduced later.

Without loss of generality, we assume that a side stream is a saturated liquid stream drawn directly from a tray. Therefore, X^S_m represents the actual composition of the side stream while Y^S_m is the vapor composition that is in equilibrium with the liquid. In the case that saturated vapor is drawn, Y^S_m will represent the side stream composition while X^S_m is the composition of the liquid in equilibrium.

We introduce a new binary variable, $V_{m,n} = 1$ if side stream m is drawn from tray n. Each side stream should be matched with only one tray:

$$\sum_n V_{m,n} = 1 \qquad m \in \mathbf{M} \qquad (6.38)$$

The liquid and vapor concentration of side stream m should match the liquid and vapor composition of the tray from which the side stream is drawn. In other words, variables X_m^S and Y_m^S should be equal to X_n and Y_{n+1}, respectively, if $V_{m,n} = 1$:

$$X_n - (1 - V_{m,n}) \leq X_m^S \leq X_n + (1 - V_{m,n}) \qquad m \in \mathbf{M}, n \in \mathbf{N} \qquad (6.39)$$

$$Y_{n+1} - (1 - V_{m,n}) \leq Y_m^S \leq Y_{n+1} + (1 - V_{m,n}) \qquad m \in \mathbf{M}, n \in \mathbf{N} \qquad (6.40)$$

Eqs. (6.10) – (6.14) are generalized, as follows, to enforce appropriate relationships between tray compositions and boundary compositions:

$$X_k^P - (1 - U_{k,n}) \leq X_n \leq X_{k-1}^P + (1 - U_{k,n}) \qquad k \in \mathbf{K}\backslash\{\tau\}, n \in \mathbf{N} \qquad (6.41)$$

$$Y_k^P - (1 - U_{k,n}) \leq Y_{n+1} \leq Y_{k-1}^P + (1 - U_{k,n}) \qquad k \in \mathbf{K}\backslash\{\tau\}, n \in \mathbf{N}\backslash\{|\mathbf{N}|\} \qquad (6.42)$$

$$X^B - (1 - U_{\tau,n}) \leq X_n \leq X^B + (1 - U_{\tau,n}) \qquad n \in \mathbf{N} \qquad (6.43)$$

If a feed is assigned to boundary k, then the composition at the boundary, X_k^P and Y_k^P, will be implicitly constrained via material balances that will be shown later (i.e., Eqs. (6.52) and (6.53)); While if a side stream is assigned to boundary k ($Z_{m,k}^S = 1$), then X_k^P and Y_k^P should represent the liquid and vapor composition of the tray from which the side stream is drawn:

$$X_{m,k}^{S,D} - (1 - Z_{m,k}^S) \leq X_k^P \leq X_{m,k}^{S,D} + (1 - Z_{m,k}^S) \qquad m \in \mathbf{M}, k \in \mathbf{K}^P \qquad (6.44)$$

$$Y_{m,k}^{S,D} - (1 - Z_{m,k}^S) \leq Y_k^P \leq Y_{m,k}^{S,D} + (1 - Z_{m,k}^S) \qquad m \in \mathbf{M}, k \in \mathbf{K}^P \qquad (6.45)$$

For example, when $Z_{m,k}^S = 1$, Eqs. (6.44) and (6.45) enforce $X_k^P = X_{m,k}^{S,D}$ and $Y_k^P = Y_{m,k}^{S,D}$. When combined with Eqs. (6.32) – (6.37), $X_k^P = X_{m,k}^{S,D} = X_m^S$ and $Y_k^P = Y_{m,k}^{S,D} = Y_m^S$, thus the composition of the boundary is equal to the composition of the side stream that is assigned to the boundary.

When combined with Eqs. (6.41) and (6.42), X_k^P and Y_k^P are equal to the composition of the tray where the side stream is drawn.

Therefore, Eqs. (6.25) – (6.45) are used to assign the feeds and side streams to section boundaries based on composition relationship, which is enforced via variable disaggregation and mixed-integer constraints.

Material balances from $\mathbb{M}1$ are included with some modifications. Specifically, the material balance written around the entire column (Eq. (6.1)) is modified to include the multiple feeds and side streams:

$$\sum_l F_l = D + B + \sum_m S_m \qquad (\mathbf{6}.46)$$

At the section boundaries, the liquid and/or vapor flow rates change due to the introduction of a feed or a side stream (see Figure 6-9). Although it is unknown a priori which feed or side stream should be assigned to boundary k, the summation of the disaggregated variables ensures that only the correct feed/side stream is included in the following material balances:

$$L_{k+1} - L_k = \sum_l \left(q_l F_{l,k}^D\right) - \sum_m S_{m,k}^D \qquad k \in \mathbf{K}^P \qquad (\mathbf{6}.47)$$

$$V_k - V_{k+1} = \sum_l (1 - q_l) F_{l,k}^D \qquad k \in \mathbf{K}^P \qquad (\mathbf{6}.48)$$

Since we assume that side streams are saturated liquid, the term $\sum_m S_{m,k}^D$ only appears in Eq. (6.47); while if a side stream is drawn from the vapor phase of a tray, then this term will appear in Eq. (6.48) instead. For the $\tau - 1$th section, the liquid and vapor flow rates are connected to the bottom liquid flow rate:

$$L_k = V_k + B \qquad k = \tau - 1 \qquad (6.49)$$

The component material balance for the entire column includes contribution from the multiple feeds and side streams:

$$\sum_l F_l X_l^F = DX^D + BX^B + \sum_m S_m X_m^S \tag{6.50}$$

Component material balance for each tray now includes the disaggregated feeds/side streams:

$$V_k Y_{n+1} + \sum_l \sum_{k' \le k-1} F_{l,k'}^D X_{l,k'}^{F,D} - L_k X_n - DX^D - \sum_m \sum_{k' \le k-1} S_{m,k'}^D X_{m,k'}^{S,D} = S_{k,n}^P - S_{k,n}^N$$

$$k \in \mathbf{K}^P, n \in \mathbf{N} \tag{6.51}$$

where slack variables $S_{k,n}^P$ and $S_{k,n}^N$ are (de)activated by Eqs. (6.17) – (6.18).

The term $\sum_l \sum_{k' \le k-1} F_{l,k'}^D X_{l,k'}^{F,D}$ represents the contributions from all the feeds that are above boundary k, while $\sum_j \sum_{k' \le k-1} S_{m,k'}^D X_{m,k'}^{S,D}$ represents the summation of flow for all the side streams that are above k.

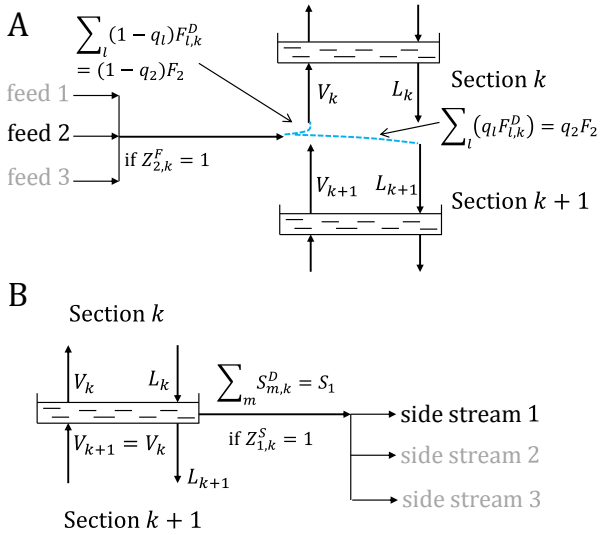


Figure 6-9. Illustrative examples of material balances at boundary k.

Finally, the component material balances at the boundaries are written as follows:

$$V_k Y_k^P + \sum_l \sum_{k' \le k-1} F_{l,k'}^D X_{l,k'}^{F,D} = L_k X_k^P + DX^D + \sum_m \sum_{k' \le k-1} S_{m,k'}^D X_{m,k'}^{S,D} \quad k \in \mathbf{K}^P \tag{6.52}$$

$$V_{k+1}Y_k^P - \sum_l \sum_{k' \geq k+1} F_{l,k'}^D X_{l,k'}^{F,D} = L_{k+1}X_k^P - BX^B + \sum_m \sum_{k' \geq k+1} S_{m,k'}^D X_{m,k'}^{S,D} \quad k \in \mathbf{K}^P \qquad (6.53)$$

where the summations of disaggregated variables are used so that only the appropriate flows are included.

The model for complex distillation columns with unknown sequencing ($\mathbb{M}3$) is as follows:

$$\min \Psi$$
$$s.t. \text{ Eqs. } (6.9) - (6.10), (6.15) - (6.16), (6.20) - (6.23), (6.25) - (6.53) \qquad \text{(M3)}$$

6.5 Extensions

6.5.1 Non-constant Molar Overflow

Since the material balances in the proposed model are derived from Sorel's method, our approach is subject to the constant-molar overflow assumption. As illustrated previously, several factors can contribute to the violation of this assumption, and one of them is the different molar latent heat of evaporation between the two components.

To address this, Peters [169] introduced the concept of "latent heat units", which is a correction of the component flow rate in terms of their molar latent heat of evaporation. Based on Peters' approach, we introduce a parameter p_i to represent the ratio of the molar latent heat between components $i \in \{lk, hk\}$ and the heavy component, hk, which can be used to convert the molar flow rate to latent-heat-based flow rate at various locations. For example, the feed flow rate and composition can be converted as follows:

$$\tilde{F}_i^C = p_i F_i^C \qquad i \in \{lk, hk\} \qquad (6.54)$$

$$\tilde{F} = \sum_i \tilde{F}_i^C \qquad (6.55)$$

$$\tilde{X}^F = \tilde{F}_{lk}^C / \tilde{F} \qquad (6.56)$$

where F_i^C is the molar component flow rate of the feed, and $\tilde{F}_i^C$, $\tilde{F}$ and $\tilde{X}^F$ are the component flow, total flow, and composition in latent heat units. Similar modifications are made at the top and bottom of the column,

$$\tilde{D}_i^C = p_i D_i^C \qquad i \in \{lk, hk\} \tag{6.57}$$

$$\tilde{D} = \sum_i \tilde{D}_i^C \tag{6.58}$$

$$\tilde{X}^D = \tilde{D}_{lk}^C / \tilde{D} \tag{6.59}$$

$$\tilde{B}_i^C = p_i B_i^C \qquad i \in \{lk, hk\} \tag{6.60}$$

$$\tilde{B} = \sum_i \tilde{B}_i^C \tag{6.61}$$

$$\tilde{X}^B = \tilde{B}_{lk}^C / \tilde{B} \tag{6.62}$$

McCabe and Thiele pointed out that if latent heat units are chosen, they must also be used to express the composition in each tray as well as to construct the VLE line. Therefore, the VLE data set must be modified as follows:

$$\tilde{\boldsymbol{x}} = \frac{\boldsymbol{x}p}{\boldsymbol{x}p + (1 - \boldsymbol{x})} \tag{6.63}$$

$$\tilde{\boldsymbol{y}} = \frac{\boldsymbol{y}p}{\boldsymbol{y}p + (1 - \boldsymbol{y})} \tag{6.64}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the vectors of x and y coordinates of the original VLE data set, and $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{y}}$ are the coordinates in latent heat units. Then, $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{y}}$ are used to find the approximating function of VLE in latent heat units.

Finally, variables in latent heat units will substitute the original variables in the equations in previous sections. As shown in Figure 6-10A, since the modification is done in an outer layer, there is no need to modify any constraint.

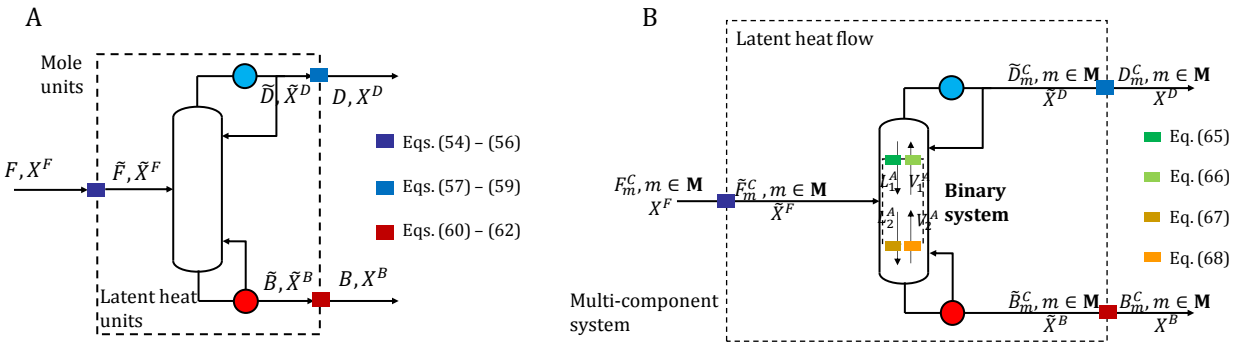


Figure 6-10. (A) The conversion from mole units to latent heat units, and (B) multicomponent distillation with non-constant-molar overflow.

6.5.2 Multicomponent Distillation

For multicomponent distillations, Hengstebeck [80] proposed a simple graphical method that is an extension to the McCabe-Thiele's method, which is also applicable to our approach with the following assumptions:

- The light and heavy key components must be specified.
- The separation is sharp (i.e., the two key components are adjacent in terms of volatility).
- The flows of heavy non-keys are negligible in the rectifying section and the light non-keys are negligible in the stripping section.

Let $i \in \mathbf{I} = \{l1, l2, \ldots, lk, hk, h1, h2, \ldots\}$ be the set of components in the mixture in the order of decreasing volatility, $\mathbf{I}^{lt} = \{l1, l2, \ldots\} \subset \mathbf{I}$ be the subset of components that are more volatile (lighter) than the light key, and $\mathbf{I}^{hv} = \{h1, h2, \ldots\} \subset \mathbf{I}$ be the subset of components that are less volatile (heavier) than the heavy key.

Hengstebeck recognized that, in actual column operation, components lighter than the light key have approximately constant flow rate in most of the rectifying section, while components heavier than the heavy key also have constant flow rate in most of the stripping section. Accordingly, we

introduce variables L_k^A and V_k^A to represent the constant liquid and vapor flow rate for the light and heavy keys combined, which are determined as follows [80]:

$$L_1^A = L_1 - \sum_{i \in \mathbf{I}^{lt}} L_{1,i}^C \tag{6.65}$$

$$V_1^A = V_1 - \sum_{i \in \mathbf{I}^{lt}} V_{1,i}^C \tag{6.66}$$

$$L_2^A = L_2 - \sum_{i \in \mathbf{I}^{hv}} L_{2,i}^C \tag{6.67}$$

$$V_2^A = V_2 - \sum_{i \in \mathbf{I}^{hv}} V_{2,i}^C \tag{6.68}$$

where $L_{k,i}^C$ and $V_{k,i}^C$ are the flow rate of non-keys i in section k, determined as follows:

$$L_{1,i}^C = \frac{D_i^C}{\alpha_i - 1} \qquad i \in \mathbf{I}^{lt} \tag{6.69}$$

$$V_{1,i}^C = L_{1,i}^C + D_i^C \qquad i \in \mathbf{I}^{lt} \tag{6.70}$$

$$V_{2,i}^C = \frac{\alpha_i B_i^C}{\alpha_{lk} - \alpha_i} \qquad i \in \mathbf{I}^{hv} \tag{6.71}$$

$$L_{2,i}^C = V_{1,i}^C + B_i^C \qquad i \in \mathbf{I}^{hv} \tag{6.72}$$

Variables D_i^C and B_i^C are the component flow rate of the distillate and the bottom liquid, and α_i is the relative volatility of component i with respect to the heavy key averaged across the column.

With modified flow rates (L_k^A and V_k^A) that are "independent" of the flow rate of non-keys, the multicomponent distillation is reduced back to a binary distillation problem so that the constraints from the previous sections can be applied with minor modifications. Specifically, Eqs. (6.4b) and (6.6b) are replaced by:

$$V_1^A Y_{n+1} - L_1^A X_n - (V_1^A - L_1^A) X^D = S_1^P - S_1^N \qquad n \in \mathbf{N} \tag{6.4c}$$

$$V_2^A Y_{n+1} - L_2^A X_n + (L_2^A - V_2^A) X^B = S_2^P - S_2^N \qquad n \in \mathbf{N} \tag{6.6c}$$

In addition, Eqs. (6.7), (6.8), and (6.19) are modified in a similar way:

$$V_1^A Y^P = L_1^A X^P + (V_1^A - L_1^A) X^D \tag{6.7b}$$

$$V_2^A Y^P = L_2^A X^P - (L_2^A - V_2^A) X^B \tag{6.8b}$$

$$L_2^A - L_1^A = q \sum_{i \in \{lk,hk\}} F_i^C \tag{6.19b}$$

The extension to multicomponent distillations can be combined with the extension to non-constant-molar overflow. This is because the adjustment of flow rates to latent heat units is done in the outer most layer. Therefore, there is no need to modify the constraints further, besides substituting the molar-flow-based variables with latent-heat-based ones (Figure 6-10B).

6.6 Solution Methods

The MINLP models in this work are nonconvex even after relaxing the integrality constraints because of the bilinear terms in the material balances (e.g. Eqs. (6.52) and (6.53)). Therefore, global optimization solvers are needed to find the globally optimal solution. We develop several solution strategies to aid the solution of these complex models.

First, we rescale variables to improve numerical stability. The proposed models mainly include two types of continuous variables: (1) flow variables such as B, D, F_l, V_k, and L_k, and (2) composition variables such as X^B,X^D, and Y_k^P. While the composition variables are naturally bounded between 0 and 1, the scale of the flow variables depends on the problem specifications. A good strategy is to rescale the sum of flow rate of all the feeds to 1 (i.e. $\sum_l F_l = 1$). After rescaling, external flow rates including B, D, and $S_m, m \in \mathbf{M}$ will be bounded by 1. With a moderate reflux ratio, the internal flow rates (L_k and V_k) will also have the same or at least similar order of magnitude. A well-scaled model is generally beneficial in terms of computation.

In addition, we introduce redundant constraints to strengthen the relaxation of the proposed models. Specifically, we take advantage of the graphical origin of the models. In the McCabe and Thiele diagram, points $\boldsymbol{x}_k^P = (X_k^P, Y_k^P)$, $k \in \mathbf{K}^P$ that represent the intersection of two operating lines must lie within the region defined by the VLE line and the 45-degree line. Similarly, in our model, $\boldsymbol{x}_k^P$ must be contained within the polygon ($\mathcal{P}$) formed by the PWLA function and the 45-degree line (see Figure 6-11). In other words, $\boldsymbol{x}_k^P, k \in \mathbf{K}^P$ can be expressed as a convex combination of the vertices of the convex-hull of $\mathcal{P}$:

$$\boldsymbol{x}_k^P = \sum_s \lambda_{s,k} \boldsymbol{x}_s \qquad k \in \mathbf{K}^P \tag{6.73}$$

$$\sum_s \lambda_{s,k} = 1 \qquad k \in \mathbf{K}^P \tag{6.74}$$

where vectors $\boldsymbol{x}_s, s \in \mathbf{S}$ represent the coordinates of the vertices of conv($\mathcal{P}$), $\lambda_{s,k} \in [0,1]$, $k \in \mathbf{K}^P$ are coefficients whose sum must be equal to 1. Eqs. (6.73) and (6.74) are valid even when $\mathcal{P}$ is nonconvex (e.g., azeotropic systems). Note that Eqs. (6.73) – (6.74) are not necessary for completeness since (X_k^P, Y_k^P) are implicitly constrained to be within the convex-hull of $\mathcal{P}$ via material balances such as Eqs. (6.52) and (6.53). Nevertheless, these redundant constraints lead to a tighter relaxed space and have found to speed up the solution process. Similar redundant constraints can be written for points (X_n, Y_{n+1}), $n \in \mathbf{N}$, which also lie within conv($\mathcal{P}$).

Another simple but useful redundant constraint is introduced for the vapor and liquid composition at each tray (Y_n and X_n). In any binary distillation system, point (X_n, Y_n) should be "above" the 45-degree line for all $n \in \mathbf{N}$,

$$Y_n \geq X_n \qquad n \in \mathbf{N} \tag{6.75}$$

which is also valid for azeotropic systems.

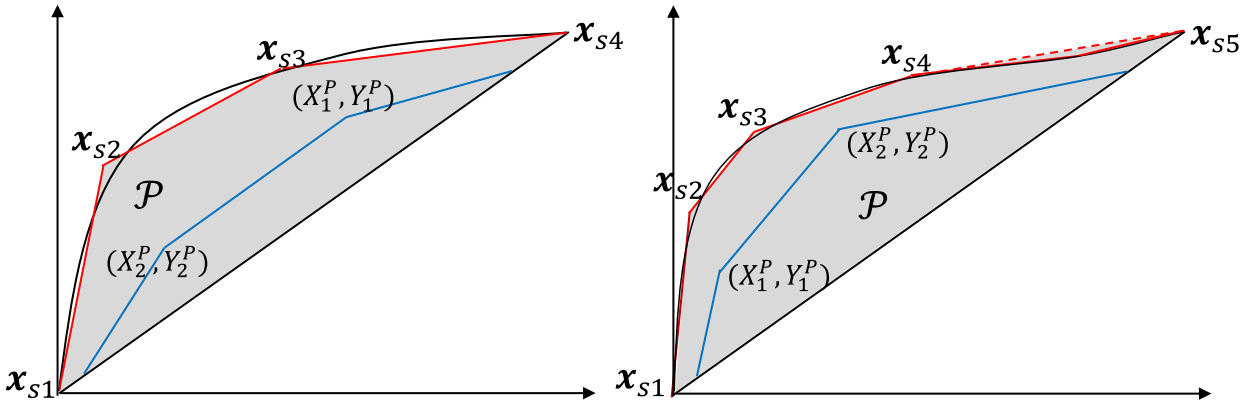


Figure 6-11. Polygon $\mathcal{P}$ and its convex-hull formed by the piecewise linear function and the 45 degree line.

6.7 Examples

We study different distillation systems to demonstrate the capabilities and advantages of the proposed approach. For brevity, we show one of them, which is an industrial-relevant air separation example that involves designing both simple and complex columns.

A typical cryogenic air separation process uses a heat-integrated double-column configuration (Figure 6-12A). The high pressure column (HPC) takes saturated air at the bottom (AIR-F) and produces a liquid stream enriched in oxygen (CLOX), which is fed to the low pressure column (LPC) after expansion. The top product of HPC (LIN-R) containing high purity nitrogen is expanded and fed to the top of the LPC as a liquid reflux. Gaseous nitrogen (GAN) and liquid oxygen (LOX) are drawn from the top and bottom, respectively, from the LPC as products. The LPC also contains a liquid nitrogen waste stream (LIN-W) and an additional air feed stream (AIR-E). The HPC can be treated as a rectifying column with one feed, while the LPC is a column with multiple feeds and side streams. In this example, we study the HPC and LPC separately.

We assume that the air feed is a pseudo-binary mixture containing nitrogen and oxygen. Other minor components such as argon are lumped together with oxygen as a pseudo-component, which is a common practice when modeling distillation columns in air separation processes.

The VLEs of nitrogen and oxygen at the HPC operating pressure (6bar) and LPC operating pressure (2bar) are obtained from ASPEN Plus (v8.6) using UNIQUAC method. The VLE data sets are then fitted using continuous piecewise linear functions with 5 segments.

In the HPC, the AIR-F stream is saturated vapor ($q = 0$) that contains 78 mol% of nitrogen. The feed flow rate is 1.00 kmol/hr. The bottom liquid stream (CLOX) is in equilibrium with the feed. Since there is no reboiler, the vapor boil-up is provided by the feed to the last tray. The material balances in our model is general enough to consider a feed to the bottom of the column. The purity of LIN-R should be at least 99.99 mol% and the reflux ratio is bounded between 0.1 and 10. We estimate that the maximum number of trays is 40. We solve $\mathbb{M}1$ minimizing $\sum_n (U_{1,n} + U_{2,n}) + 100R$. For the LPC, the specifications are summarized in [170]. The upper limit on the number of trays is 30. While the sequence of the feeds and the side stream can be determined a priori, we choose not to specify them, letting the optimization determine the optimal sequencing of these streams. We solve $\mathbb{M}3$ to minimize the number of trays.

Both models are solved, separately, to global optimality using BARON (17.10.16) [96]. In the optimal solutions, the HPC has 27 trays with the AIR-F stream fed to the last tray (Figure 6-13A) and a reflux ratio of 0.957. The distillate flow rate (D =0.511 kmol/hr) and composition ($X^D = 0.9999$) are determined. Similarly, the bottom liquid flow rate ($B = 0.489$ kmol/hr) and composition ($X^B = 0.571$) are also determined.

For the LPC, the optimal design has 24 trays plus a reboiler. The side stream (LIN-W) with a 0.06 flow rate is drawn from the 13th tray with a composition of 0.904. The AIR-E stream is fed to above the 16th tray while the CLOX is fed to above the 18th tray. The optimal design of the LPC is represented graphically in Figure 6-13B.

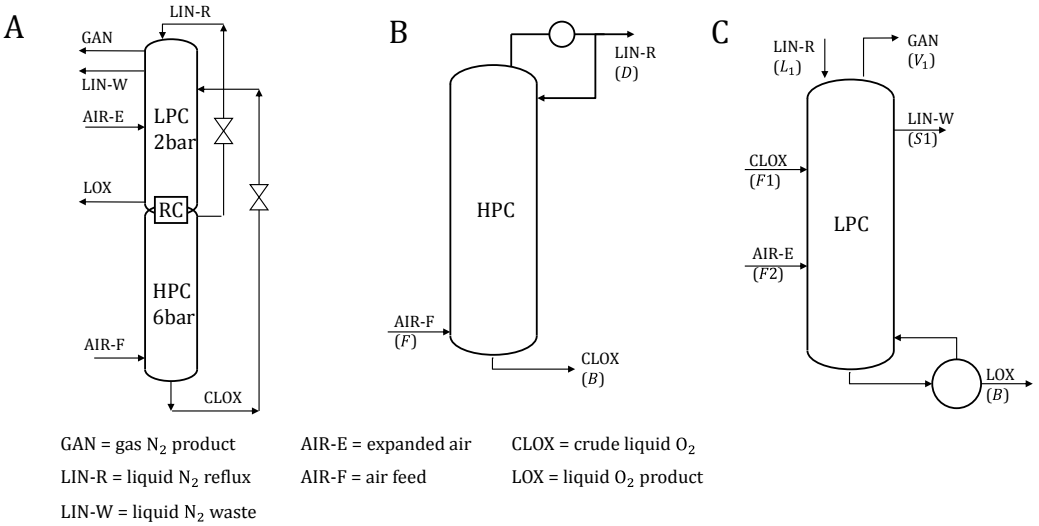


Figure 6-12. (A) A typical double-column configuration in air separation, (B) the high pressure column modeled as a simple column and (C) the low pressure column modeled as a complex column.

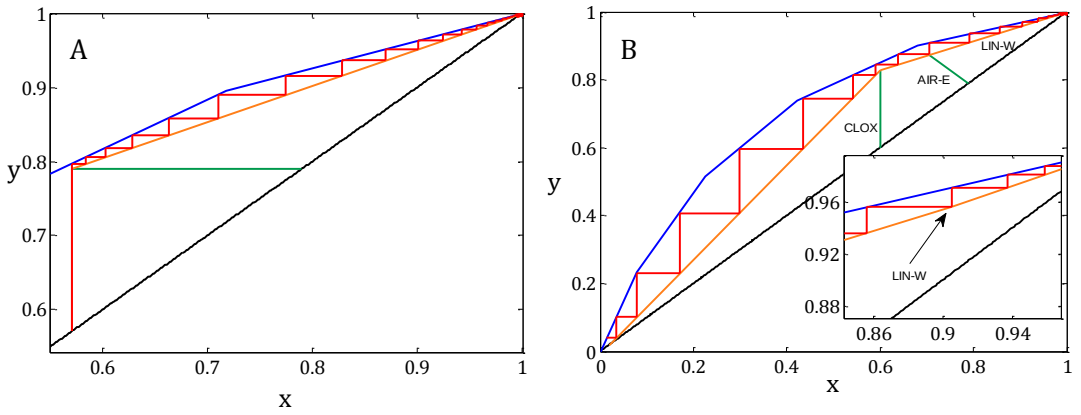


Figure 6-13. Graphical representation of the optimal designs of (A) the HPC and (B) the LPC.

6.8 Conclusions

We introduced a mathematical-programming model for simple and complex distillation column design and optimization. The model is inspired by the classical McCabe-Thiele graphical method. Similar to the Sorel's equations on which the McCabe-Thiele method is based, we wrote overall and component material balances at different locations of the distillation column. We introduce binary variables and mixed-integer constraints to determine the number of trays and the optimal feed trays. The complex VLE relationship was approximated via inexpensive continuous

piecewise linear functions, which, surprisingly, can be more accurate than EOS when experimental data is available. Moreover, we showed that only a few segments are needed for a PWLA function to represent the VLE accurately. Compared to the original graphical approach, our optimization model is more flexible and can handle multiple degrees of freedom. On the other hand, our model is more tractable than the rigorous approaches as the complicated physical property and VLE calculations are greatly simplified. The model can be conveniently modified to account for nonconventional columns such as columns with partial condenser or reflux and/or boil-up provided by external streams, thereby facilitates its integration with other unit models in process synthesis.

Similar to its graphical counterpart, the proposed approach, even with the aforementioned extension, cannot handle arbitrary multicomponent separations with nonconsecutive keys or multicomponent azeotropic systems. Nevertheless, it is readily applicable to industrial-relevant processes such as cryogenic air separation and hydrogen/carbon monoxide separation.

Chapter 7
Distillation Column Modeling for Superstructure-based Process Synthesis[9]

7.1 Motivation

In chemical and petrochemical industries, distillation columns are the most commonly used units to separate a multicomponent mixture. The vapor flowrate in a column is directly related to the reboiler and condenser duties and the column diameter. Therefore, finding the minimum vapor flowrate (V^{min}) is useful to understand the energy demand and capital cost of the separation. Among the many methods for calculating V^{min}, the Underwood method [63,171] is the most widely used.

To use the Underwood method, the light and heavy key components must be specified. For a binary mixture, the more volatile component is, by definition, the light key, and the less volatile one is the heavy key. For a multicomponent mixture, the designers will have to choose the key components based on the goal of the separation. Defining key components is crucial because the number and values of the active roots (ϕ) in the Underwood equations are closely related to the choice of the key components.

Underwood showed that the number of active roots is equal to the number of distributed components (i.e. components more volatile than the heavy key but less volatile than the light key) plus one. Each active underwood root is bounded by the relative volatility of two components with

[9] The contents of this chapter appear in Kong and Maravelias, *AIChE J.* submitted.

adjacent volatility. For example, consider the separation of a four-component mixture ABCD, in which the volatility of components decreases in successive order (i.e., A is the most volatile component and D the least volatile component). In this paper, components are always ordered from high to low volatility. If A is the light key, D is the heavy key, and B and C are distributed components, then there are three active Underwood roots, which are bounded by the relative volatility of the key and distributed components:

$$\alpha_D < \phi_3 < \alpha_C < \phi_2 < \alpha_B < \phi_1 < \alpha_A \tag{7.1}$$

where α_i is the relative volatility, defined as K_i/K^0, where K_i and K^0 are Henry's law constants for component i and a reference component (component D in this example). Clearly, the use of Underwood method requires the a priori definition of the key and distributed components so that correct number of active roots and bounds on the roots can be determined.

However, incorporating the Underwood equations in a math-programming-based *optimization* problem is not always easy, especially when the component flowrates of the feed stream are variables. If the optimization problem contains a feasible solution where the feed flowrates of some components are equal to zero, then defining and bounding the active roots cannot be done prior to optimization. In the previous example, when the feed flowrate of B is equal to zero, Eq. (7.1) is not valid, as there exists only one distributed component (C). The *two* active roots should be bounded as follows:

$$\alpha_D < \phi_2 < \alpha_C < \phi_1 < \alpha_A \tag{7.2}$$

The possibility of zero-flow can also cause difficulties in the definition of key components. Consider a prefractionator in which light and heavy keys are the most and least volatile components in the mixture, respectively. For a mixture of ABCD, if the feed flowrates of all the

components are nonzero, then A is the light key and D is the heavy key. The three active roots are bounded as in Eq. (7.1); while if the feed flowrate of A is zero, then B becomes the light key and thus the two active roots are bounded as follows:

$$\alpha_D < \phi_2 < \alpha_C < \phi_1 < \alpha_B \tag{7.3}$$

Therefore, the number of roots and their bounds cannot be determined a priori when the component flowrates can be equal to zero. Zero component flows into a column, or a network of distillation units, can occur when the reactor and separation (distillation) networks are synthesized simultaneously (see **Figure 7-1**). For example, consider the synthesis of a process for the valorization of lignin (a polymeric component of biomass), for which a wide range of depolymerization technologies are available. Thus, depending on the selection of the depolymerization reactions, different bioproducts will be produced, which means that some of the flows of the postulated components from the reactor network to the distillation network will be zero.

In cases where the distillation column is modeled so that it can carry out multiple separation tasks [172], the key components cannot be determined prior to optimization. Consider a distillation column that can either separate A or C from a ternary mixture ABC. If A is separated, then A is the light key and B is the heavy key, and thus the active Underwood root is bounded by the relative volatility of A and B: $\alpha_B < \phi < \alpha_A$. If C is separated, then B is the light key and C is the heavy key, and thus $\alpha_C < \phi < \alpha_B$. Therefore, even when the feed flowrates are nonzero for all the components in the feed, the Underwood equations are still challenging to use due to the unknown key components.

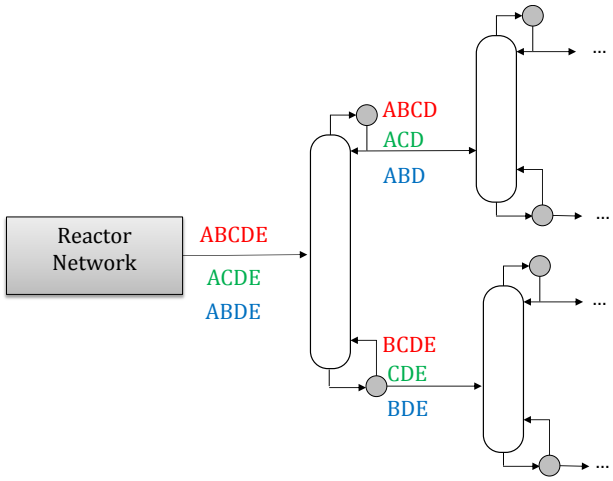


Figure 7-1. A five-component mixture (ABCDE) to be separated using distillation columns.

Accordingly, in this chapter, we introduce an Underwood-based approach for calculating minimum vapor flowrate, which is tailored to be used in an optimization model in which some components may have zero flowrate or the key components cannot be determined prior to optimization.

7.2 Minimum Vapor Flowrate

7.2.1 The Underwood Method

Given a set of components $i \in \mathbf{I}$ that are ordered from high to low volatility ($\alpha_i > \alpha_{i+1}$) including n distributed components ($0 \leq n \leq |\mathbf{I}| - 2$), Underwood suggested that there exist $n + 1$ active roots ϕ_r, $r \in \mathbf{R} = \{1, \dots, n+1\}$, which are bounded by relative volatilities as follows:

$$\alpha_{i=h-r+1} < \phi_r < \alpha_{i=h-r} \qquad r \in \mathbf{R}, i \in \mathbf{I}^C \qquad (7.4)$$

where the heavy key is the hth component in set $\mathbf{I}$, and $\mathbf{I}^C \subset \mathbf{I}$ is an ordered subset containing only the key and distributed components. In our convention, α_i is the relative volatility of component i with respect to the heaviest component, which is not necessarily the heavy key. For instance, if $\mathbf{I} = \{\overline{\text{A}}, \text{B}, \text{C}, \underline{\text{D}}, \text{E}\}$ and $\overline{\text{A}}$ is the light key and $\underline{\text{D}}$ is the heavy key, then $\mathbf{I}^C = \{\overline{\text{A}}, \text{B}, \text{C}, \underline{\text{D}}\}$ and Eq. (7.4)

is equivalent to Eq. (7.1). Note that for the remainder of this paper, we use over bar to represent the light key and under bar for the heavy key.

The Underwood equations for calculating minimum vapor flowrate can be written as follows:

$$\sum_i \frac{\alpha_i F_i}{\alpha_i - \phi_r} = (1-q)\sum_i F_i \qquad r \in \mathbf{R} \tag{7.5}$$

$$\sum_i \frac{\alpha_i D_i}{\alpha_i - \phi_r} = V_1^{min} \qquad r \in \mathbf{R} \tag{7.6}$$

$$\sum_i \frac{\alpha_i B_i}{\alpha_i - \phi_r} = -V_2^{min} \qquad r \in \mathbf{R} \tag{7.7}$$

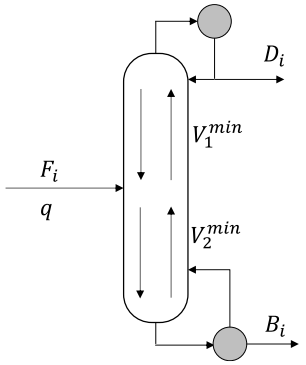


Figure 7-2. A distillation column operating at minimum vapor flowrate.

where F_i, D_i, and B_i are the component flowrates of the feed, the top, and the bottom product streams, respectively; V_1^{min} and V_2^{min} are the minimum vapor flow rate in the rectifying and stripping sections, respectively, and q is the quality of the feed (see **Figure 7-2**). In this paper, we assume that the feed is always at its bubble point ($q = 1$), and thus $V_1^{min} = V_2^{min}$. Note that the Underwood equations can be equivalently written in the composition space. For instance, Eq. (7.5) can be written as

$$\sum_i \frac{\alpha_i x_i^F}{\alpha_i - \phi_r} = 1 - q \tag{7.5b}$$

where X_i^F is the mole fraction of i in the feed. To use the Underwood equations, the recoveries of the key components ($\gamma_i = D_i/F_i, i \in \{LK, HK\}$) must be specified. In addition, any component that is heavier than the heavy key does not appear in the top product stream and any component that is lighter than the light key does not appear in the bottom stream, which is commonly referred to as sharp split.

7.2.2 Components with Zero Flowrate

Unlike in the original approach where the set of roots is defined over components in **R**, here we redefine the set of roots in the "**I**" *space* as: $\phi_i, \forall i \in \mathbf{I}^R \subset \mathbf{I}$. Subset $\mathbf{I}^R = \{i \in \mathbf{I} | lk \leq i \leq hk - 1\}$, in which lk represents the order of the light key and hk represents the order of the heavy key in the ordered set **I**. In this paper, we are using "i" interchangeably to denote an element in the set (e.g. B or C) or the order of that element (e.g. 2 or 3) in the ordered set.

Using this definition, the Underwood equations can be re-written as follows:

$$\sum_i \frac{\alpha_i F_i}{\alpha_i - \phi_{i'}} = (1-q)\sum_i F_i \qquad i' \in \mathbf{I}^R \qquad (7.8)$$

$$\sum_i \frac{\alpha_i D_i}{\alpha_i - \phi_{i'}} = V_1^{min} \qquad i' \in \mathbf{I}^R \qquad (7.9)$$

$$\sum_i \frac{\alpha_i B_i}{\alpha_i - \phi_{i'}} = -V_2^{min} \qquad i' \in \mathbf{I}^R \qquad (7.10)$$

Variables ϕ_i are bounded by the relative volatility of the light and heavy keys ($\alpha_{HK} < \phi_i < \alpha_{LK}, \forall i \in \mathbf{I}^R$). We introduce a nonnegative variable, Δ_i, to represent the difference between ϕ_i and ϕ_{i+1}:

$$\phi_i - \phi_{i+1} = \Delta_i \qquad i \in \mathbf{I}^{RD} \qquad (7.11)$$

where $\mathbf{I}^{RD} = \{i \in \mathbf{I} | lk \leq i \leq hk - 2\}$. Without a loss of generality, here we assume that the roots are ordered from large to small values (i.e. $\phi_i \geq \phi_{i+1}, i \in \mathbf{I}^{RD}$).

If the key components are defined and all the distributed components have nonzero flowrates, then the following constraint is enforced:

$$\Delta_i \geq \epsilon \qquad i \in \mathbf{I}^{RD} \tag{7.12}$$

where ϵ is a small positive number. Eqs. (7.11) and (7.12) enforce that there are $n + 1$ distinct ϕ_i. When combined with the Underwood equations (Eqs. (7.8) – (7.10)), the $n + 1$ distinct roots become the active roots that are bounded by the relative volatilities as in Eq. (7.4) (see **Figure 7-3**). Therefore, at this point, our approach (Eqs. (7.8) – (7.12)) is equivalent to the original Underwood method (Eqs. (7.4) – (7.7)).

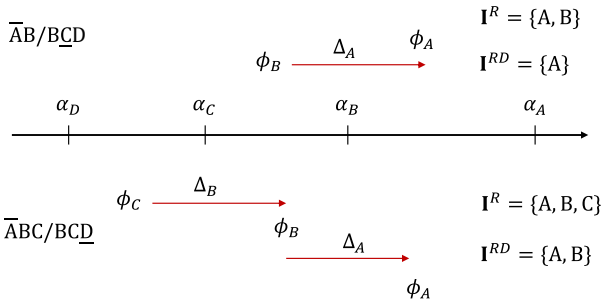


Figure 7-3. Relationship between α and ϕ in two different separation tasks.

Now, let us assume that the key components are defined but some *distributed* components have zero flowrate. Enforcing Eqs. (7.4) – (7.7) or Eqs. (7.8) – (7.12) will lead to infeasibility because the number of active roots is less than n. To address this, we reformulate Eq. (7.12) by first introducing a binary variable X_i, which is equal to 1 only if a distinct root is necessary. Variable Δ_i is strictly positive if and only if $X_i = 1$:

$$\epsilon X_i \leq \Delta_i \qquad i \in \mathbf{I}^{RD} \tag{7.12b}$$

$$\Delta_i \leq (\alpha_{LK} - \alpha_{HK})X_i \qquad i \in \mathbf{I}^{RD} \tag{7.13}$$

For example, if $\mathbf{I} = \{\overline{\text{A}}, \text{B}, \text{C}, \underline{\text{D}}\}$, then $\mathbf{I}^R = \{\text{A}, \text{B}, \text{C}\}$ and $\mathbf{I}^{RD} = \{\text{A}, \text{B}\}$. When $F_B = 0$ (see **Figure 7-4**), we only need to enforce $X_A = 0$ and $X_B = 1$ so that $\Delta_A = 0$ and $\Delta_B > 0$, and thus $\phi_A = \phi_B > \phi_C$ according to Eq. (7.11). When combined with Eqs. (7.8) – (7.10), $\alpha_D < \phi_C < \alpha_C < \phi_B = \phi_A < \alpha_A$ is enforced.

The values of X_i are related to the component flowrates. First, we need to determine which, if any, distributed components have zero flowrate. To do that, we introduce a binary variable, Y_i, which is equal to zero if and only if the flowrate of i is zero:

$$Y_i f_i^{LO} \leq F_i \leq Y_i f_i^{UP} \qquad i \in \mathbf{I} \tag{7.14}$$

where f_i^{UP} is a valid upper bound on component flowrate, and f_i^{LO} is a small positive number to indicate the existence of component i in the mixture. The activation and deactivation of the Underwood roots are directly related to the existence of the distributed components:

$$X_i = Y_{i+1} \qquad i \in \mathbf{I}^{RD} \tag{7.15}$$

When a distributed component does not exist (i.e. $F_{i+1} = Y_{i+1} = 0$), then we systematically enforce $X_i = \Delta_i = 0$ so that $\phi_{i+1} = \phi_i$. Eqs. (7.14) – (7.15) provide an interface between the flowrate of the distributed components and the Underwood equations so that the active roots can be implicitly constrained. In the previous example where $\mathbf{I} = \{\overline{\text{A}}, \text{B}, \text{C}, \underline{\text{D}}\}$ and $F_B = 0$, Eq. (7.14) enforces that $Y_B = 0$ and $Y_C = 1$. Then, Eq. (7.15) correctly enforces that $X_A = 0$ and $X_B = 1$ (**Figure 7-4**).

When distribution components can have zero flowrate, the following constraint is used to ensure that no ϕ_i should be equal to α_i:

$$(\phi_{i'} - \alpha_i)^2 \geq \epsilon' \qquad i' \in \mathbf{I}^R, i \in \mathbf{I} \qquad (7.16)$$

where ϵ' is a small positive number. Eq. (7.16) ensures that any $\phi_{i'}$ must be at least $\sqrt{\epsilon'}$ away from any α_i. This is critical because if $F_i = 0$ and $\phi_{i'} - \alpha_i = 0$ can be true simultaneously, then the Underwood equations are numerically unstable.

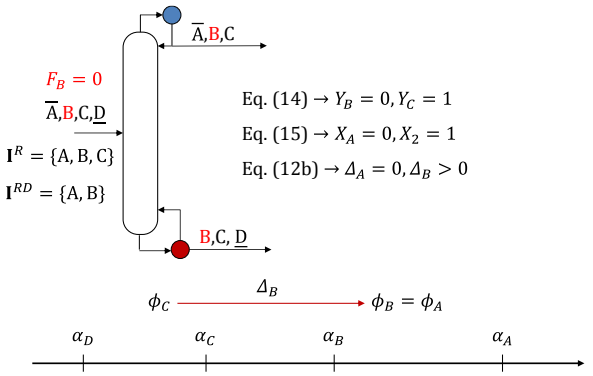


Figure 7-4. Separation of ABCD when the flowrate of B is equal to zero.

7.2.3 Unknown Key Components

We consider a more general case where the key components are unknown prior to optimization. We introduce a pair of binary variables (Y_i^{LK} and Y_i^{HK}) to represent the choice of the key components. If $Y_i^{LK} = 1$, then i is the light key; while if $Y_i^{HK} = 1$, then i is the heavy key. In this work, however, we simply assume they are given.

All possibly active underwood roots must be bounded by the relative volatility of the key components:

$$\sum_i \alpha_i Y_i^{HK} + \sqrt{\epsilon'} \leq \phi_{i'} \leq \sum_i \alpha_i Y_i^{LK} - \sqrt{\epsilon'} \qquad i' \in \mathbf{I}^R \qquad (7.17)$$

When the key components are unknown a priori, the set of distributed components is also unknown. Therefore, in this subsection, we define $\mathbf{I}^R = \{i \in \mathbf{I} | i \leq |\mathbf{I}| - 1\}$ and $\mathbf{I}^{RD} = \{i \in \mathbf{I} | i \leq |\mathbf{I}| - 2\}$ to account for all the possible active Underwood roots for any choice of key components.

We introduce $Z_i \in \{0,1\}$, which is equal to one if and only if i is a distributed component:

$$Z_i = \sum_{i'' \le i-1} Y_{i''}^{LK} - \sum_{i' \le i} Y_{i'}^{HK} \qquad i \in \mathbf{I} \tag{7.18}$$

To determine if a distinct root is necessary, Eq. (7.15) can be rewritten as follows:

$$X_i = Z_{i+1} Y_{i+1} \qquad i \in \mathbf{I}^{RD} \tag{7.15b}$$

which can be easily reformulated into linear equations. The logic that Eq. (7.15b) enforces is: the $i+1$th root is active ($X_i = 1$) if and only if component $i+1$ is a distributed component ($Z_{i+1} = 1$) and the component flowrate is nonzero ($Y_{i+1} = 1$).

To illustrate, consider $i \in \mathbf{I} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{F}\}$ without specifying the key components. If the process synthesis model yields $Y_B^{LK} = Y_F^{HK} = 1$ and $F_D = 0$, then according to Eq. (7.17), ϕ_i will be bounded in $[\alpha_F, \alpha_B]$. At the same time, Eq. (7.14) relates the component flowrates with the corresponding Y_i ($Y_A = Y_B = Y_C = Y_E = 1$ and $Y_D = 0$) and Eq. (7.18) yields the set of distributed components ($Z_C = Z_D = Z_E = 1$). Eq. (7.15b) determines the values of binary variables X_i ($X_B = X_D = 1$ and $X_A = X_C = 0$), which are used to (de)activate Δ_i in Eqs. (7.12b) and (7.13). Values of these variables are tabulated in Table 1. When combined with Eq. (7.11), the active (i.e. distinct) roots are determined (ϕ_E, ϕ_C, ϕ_B). As shown in **Figure 7-5**, ϕ_D becomes equal to ϕ_C because distributed component D has zero flowrate and $\phi_A = \phi_B$ because A is neither a key nor a distributed component. From Eqs. (7.8) – (7.18), the number of distinct Underwood roots and their values can be determined when the key components are unknown prior to optimization.

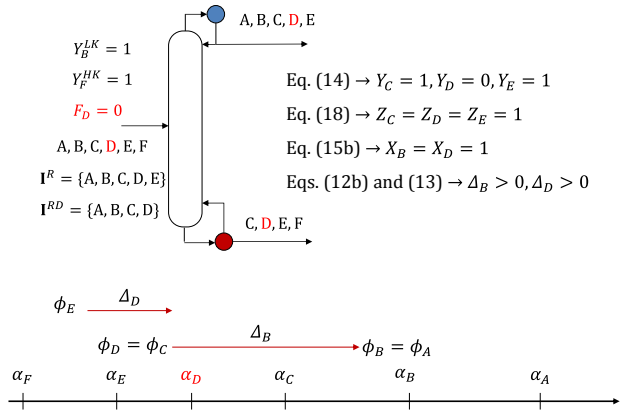


Figure 7-5. Separation of ABCDEF without specifying the key components.

Table 7-1. Values of key variables for instance shown in **Figure 7-5**.

Variables	A	B	C	D	E	F
Y_i^{LK}	0	1	0	0	0	0
Y_i^{HK}	0	0	0	0	0	1
Y_i	1	1	1	0	1	1
Z_i	0	0	1	1	1	0
X_i	0	1	0	1	-	-
Δ_i	0	>0	0	>0	-	-

7.3 Extensions

7.3.1 Integration with Process Synthesis

The proposed approach calculates the minimum vapor flowrate for arbitrary feed flowrates and unknown key components. The actual vapor flowrate can be estimated as 1.2 times the minimum vapor flowrate [173]. The vapor flowrate can then be used to calculate the condenser and reboiler duty and the column area. Together with the (minimum) number of trays, which can be estimated by Fenske equation, the design and cost of the column can be obtained from a system of mixed-integer nonlinear equations. Therefore, the proposed approach can be readily used as a submodule for distillation column design and be integrated into a process optimization model.

In superstructure-based process synthesis, the flowrate of the effluent of the reactor superstructure (F_i^E) becomes the feed flowrate (F_i) of the first separation unit (column) in the separation

(distillation) superstructure. Therefore, F_i^E and F_i are the linking variables that couple the reactor and distillation superstructures. Without a loss of generality, we assume that there is one stream coupling the two superstructures. Whether F_i^E is equal to zero or not often depends on some discrete decisions such as the selection of reactors. In this case, f_i^{LO} in Eq. (7.14) can be calculated from the lower bounds of the reactors that have component i in the effluent. For example, in **Figure 7-6**, component B only appears in the effluent of reactor R1, and thus f_B^{LO} is equal to the lower bound on the flow of B in the effluent of R1.

In the distillation superstructure, two types of information, for each distillation column, must be either specified by the user or determined via optimization. First, the key components for each column are represented via binaries Y_i^{LK} and Y_i^{HK}. The values of Y_i^{LK} and Y_i^{HK} depend not only on the goal of the separation (e.g. direct versus indirect split), but also the existence of components in the feed stream. While how these binaries are determined is not of our concern here, there should be exactly one light key and one heavy key:

$$\sum_i Y_i^{LK} = 1 \tag{7.19}$$

$$\sum_i Y_i^{HK} = 1 \tag{7.20}$$

When the split is sharp and the recovery of the key components given, the following constraints are included:

$$D_i \geq F_i \gamma_{LK} - \delta_i^{UP}(1 - Y_i^{LK}) \qquad i \in \mathbf{I} \tag{7.21}$$

$$B_i \geq F_i(1 - \gamma_{HK}) - \beta_i^{UP}(1 - Y_i^{HK}) \qquad i \in \mathbf{I} \tag{7.22}$$

where δ_i^{UP} and β_i^{UP} are upper bounds on the component flowrate at the top and bottom, respectively. They can be calculated based on upper bounds on the feed flowrates (f_i^{UP}) and the recoveries as follows:

$$\delta_i^{UP} = f_i^{UP}\gamma_{LK} \qquad i \in \mathbf{I} \qquad (7.23)$$

$$\beta_i^{UP} = f_i^{UP}(1-\gamma_{HK}) \qquad i \in \mathbf{I} \qquad (7.24)$$

Second, the relative volatilities must be given or calculated by optimization. The volatility (e.g. K-value) of each component is a function of the temperature and pressure, which might be unknown priori to solving the optimization problem. To account for that, Raoult's Law and Antoine equation can be incorporated into the model to relate the temperature, pressure, and composition at the top and bottom of the column. Meanwhile, the K-value of each component can be determined at the top and bottom so that the averaged relative volatilities can be obtained and used in the Underwood equations.

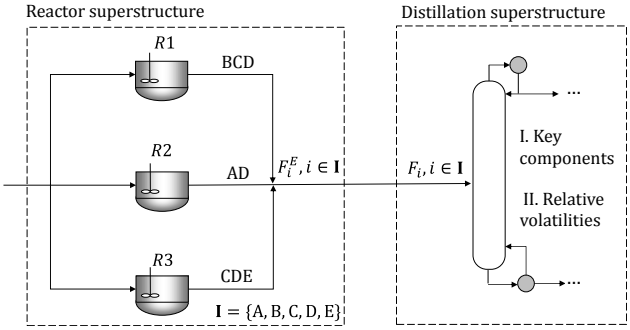


Figure 7-6. An example of separating a mixture from reactor effluents.

7.3.2 Solution Methods

When used in a math programming model, the Underwood equations (Eqs. (7.8) – (7.10)) can be numerically unstable, especially when the flowrates of some components are equal to or approach zero. Eq. (7.16) was introduced to address this issue. However, choosing the right ϵ' in Eq. (7.16)

is not simple because an ϵ' that is too large could cutoff feasible solutions while an ϵ' that is too small is not effective.

To address this, we develop solution strategies to improve the numerical stability of our approach. The first step is to introduce auxiliary variables ($U^F_{i,i'}, U^D_{i,i'}$, and $U^B_{i,i'}$) on the left-hand-side (LHS) of the Underwood equations:

$$U^F_{i,i'} = \frac{\alpha_i F_i}{\alpha_i - \phi_{i'}} \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.25}$$

$$U^D_{i,i'} = \frac{\alpha_i D_i}{\alpha_i - \phi_{i'}} \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.26}$$

$$U^B_{i,i'} = \frac{\alpha_i B_i}{\alpha_i - \phi_{i'}} \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.27}$$

representing the terms in the summations in Eqs. (7.8) – (7.10). The above equations can be rearranged as follows:

$$U^F_{i,i'}(\alpha_i - \phi_{i'}) = \alpha_i F_i \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.28}$$

$$U^D_{i,i'}(\alpha_i - \phi_{i'}) = \alpha_i D_i \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.29}$$

$$U^B_{i,i'}(\alpha_i - \phi_{i'}) = \alpha_i B_i \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.30}$$

The summations of the auxiliary variables are equal to the right-hand-side (RHS) of the Underwood equations:

$$\sum_i U^F_{i,i'} = (1-q)\sum_i F_i \qquad i' \in \mathbf{I}^R \tag{7.31}$$

$$\sum_i U^D_{i,i'} = V_1^{min} \qquad i' \in \mathbf{I}^R \tag{7.32}$$

$$\sum_i U^B_{i,i'} = -V_2^{min} \qquad i' \in \mathbf{I}^R \tag{7.33}$$

While state-of-the-art global optimization solvers reformulate constraints during preprocessing, manual reformulation brings two advantages.

First, these auxiliary variables can be bounded. For example, bounds on $U_{i,i'}^{F}$ can be calculated based on the upper bound on F_i and the minimum allowable difference between $\phi_{i'}$ and α_i. Second, if there is a possibility that $F_i = D_i = B_i = 0$ for some components, then the following constraints enforce the auxiliary variables to be equal to zero:

$$u_{i,i'}^{F,L} Y_i \leq U_{i,i'}^{F} \leq u_{i,i'}^{F,U} Y_i \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.34}$$

$$u_{i,i'}^{D,L} Y_i \leq U_{i,i'}^{D} \leq u_{i,i'}^{D,U} Y_i \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.35}$$

$$u_{i,i'}^{B,L} Y_i \leq U_{i,i'}^{B} \leq u_{i,i'}^{B,U} Y_i \qquad i \in \mathbf{I}, i' \in \mathbf{I}^R \tag{7.36}$$

where $u_{i,i'}^{F,L}$, $u_{i,i'}^{D,L}$, $u_{i,i'}^{B,L}$ are lower bounds and $u_{i,i'}^{F,U}$, $u_{i,i'}^{D,U}$, $u_{i,i'}^{B,U}$ are upper bounds on the corresponding auxiliary variables. Eqs. (7.34) – (7.36) are very important because when $F_i = D_i = B_i = 0$, Eqs. (7.28) – (7.30) only enforce $(\alpha_i - \phi_{i'}) = 0$ or $U = 0$. Without Eqs. (7.34) – (7.36), U might be mistakenly set to nonzero values.

7.4. Application

We study the initial separation of a seven-component mixture (ABCDEFG) using distillation (see **Figure 7-7**A). While the key components are unspecified, we assume that the light key is either B or C and the heavy key is either F or G. Therefore, $\mathbf{I}^R = \{\mathrm{B}, \mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{F}\}$ and $\mathbf{I}^{RD} = \{\mathrm{B}, \mathrm{C}, \mathrm{D}, \mathrm{E}\}$. In particular, we investigate a choice of key components ($Y_B^{LK} = Y_F^{HK} = 1$). The feed flowrates are also assumed given. Particularly, the feed flowrate of C is equal to zero ($F_C = 0$).

This is a typical example of unknown key components and zero component flowrate. Therefore, Eqs. (7.8) – (7.24) are included and the objective function is minimizing vapor flowrate. The resulting MINLP model contains 139 equations and 65 variables (30 binaries). It is solved to global optimality in 10 seconds. As shown in **Figure 7-7**B, there are three distinct roots (ϕ_B, ϕ_D, and ϕ_E). Since $F_C = 0$, the model enforces $Y_C = X_B = 0$ and thus $\phi_C = \phi_B$. On the other hand, since F is selected as the heavy key, the inactive ϕ_F automatically becomes equal to ϕ_E (see **Figure 7-7**C).

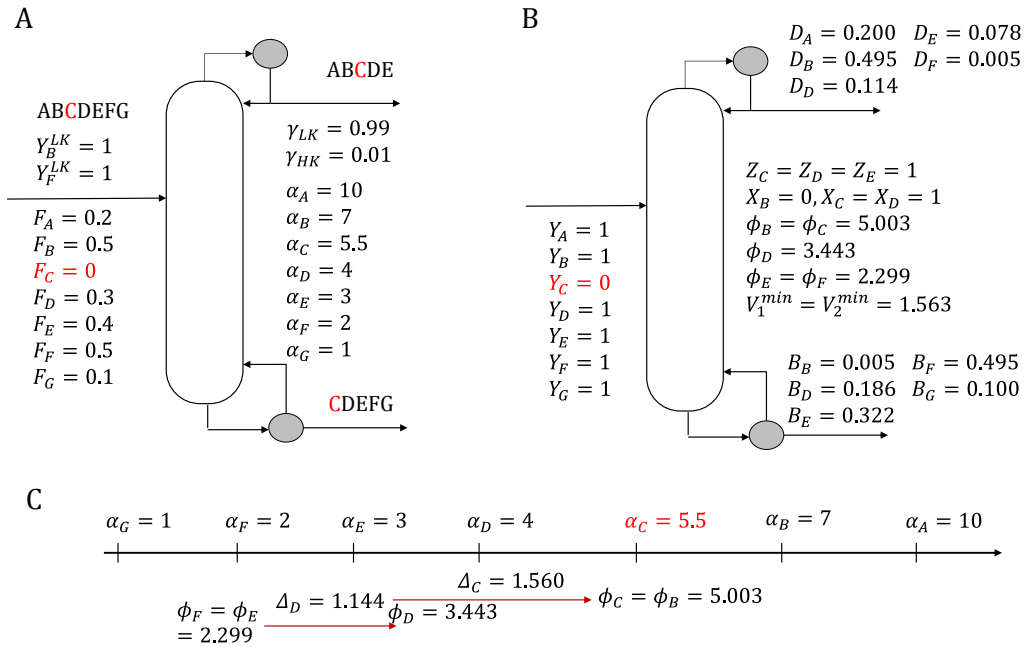


Figure 7-7. Separation of a seven-component mixture: (A) column specifications, (B) optimal solution, and (C) root locations.

7.5 Conclusions

We proposed a new approach to facilitate the use of Underwood equations for calculating minimum vapor flowrate in optimization models for process synthesis as well as distillation network synthesis. Unlike the original method, the number and values of distinct Underwood roots are implicitly determined via discrete variables and mixed integer constraints, which are based on

the selection of key components and the variable component flowrate. When a distributed component has zero flowrate, the corresponding root is deactivated. Therefore, the approach can be used when the feed flowrates and compositions are variables and when the separation task carried out by the column is unknown prior to optimization, which may occur when our approach is integrated into a process synthesis model. Furthermore, our approach can be combined with other equations such as the Fenske equation and the Raoult's Law to obtain more accurate estimation of the capital and operating cost of the column.

Chapter 8
Expanding the Scope of Distillation Network Synthesis Using Superstructure-based Methods[10]

8.1 Motivation

Despite the substantial work in the area of distillation/separation network synthesis, most of the existing literatures have focused either on the distillation network itself or the optimization of the distillation network and the heat exchanger network, which is justifiable due to the high energy demand of distillation. However, from a process synthesis standpoint, the distillation network also strongly interacts with the reactor network: the distillation network is used to separate the mixture(s) in the effluent(s) of the reactor network. All the existing distillation-sequencing approaches assume that a stream containing a *given* n-component mixture is to be separated. While some methods allow the flowrate of each component to vary in that stream, there is an implicit assumption: the flowrate of every component is nonzero (strictly positive). In other words, all the n postulated components must exist in the stream that connects the reactor and distillation network. This assumption is, however, not always valid, especially in the superstructure-based process synthesis where the reactor superstructure can produce different types of effluents, depending on the selections of reactions/reactors. For example, in a process that produces bio-renewable chemicals through valorization of lignin (a polymeric component of biomass), a wide range of depolymerization strategies are available. Therefore, different bioproducts will be produced based

[10] The contents of this chapter appear in Kong and Maravelias. *Comp. & Chem. Engr.* Submitted.

on the selection of the depolymerization reactions, which means that the flowrates of some of the postulated components will be zero.

Streams with zero flows for some of the components result in a challenge when generating and modeling the distillation superstructure. All the existing approaches assume no intermediates (i.e. states) correspond to a subset of components with nonconsecutive volatility (e.g. ABDE cannot be obtained from ABCDEF using distillation), which is true when the feed flowrates of all the postulated components are strictly positive. However, some of these intermediates should be admissible when the feed flowrates of some components are zero (e.g. ABDE *can* be obtained from ABCDEF when the flowrate of C is zero). Therefore, distillation superstructure-based methods must be generalized to consider these intermediate products/states and the separation tasks that produce them. In addition, we need to address the numerical issues arising by allowing components with zero flows.

Another challenge comes from the possibility of multiple effluent streams produced by the reactor network. Since these streams may have different compositions or even contain different sets of components, where (i.e. which distillation columns) to introduce them in the distillation network should be an optimization decision. However, all existing approaches for distillation network synthesis do not consider such systems because they assume that there is only one initial feed stream to the distillation network.

Accordingly, we propose a superstructure-based approach for near-ideal and non-azeotropic distillation sequence synthesis, which is applicable to streams with variable, including zero, component flows. The proposed approach is tailored for simultaneous reactor and distillation network synthesis where one or more streams couple the reactor and distillation superstructures.

8.2 Proposed Model

8.2.1 Superstructure Generation and Representation

Shah and Agrawal [59] proposed a matrix-based method for enumerating distillation configurations, in which the n component mixture to be separated, the intermediates, and the final products are "placed" in an $n \times n$ upper-triangular matrix (**Figure 8-1**A). In the proposed model, the distillation superstructure is constructed using a similar matrix representation.

Without loss of generality, we assume that the n postulated components to be separated, $i \in \mathbf{I} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \dots\}$, are ordered from high to low volatility. The rows and columns of the $n \times n$ matrix are defined as $j \in \mathbf{J} = \{1,2, \dots, n\}$ and $k \in \mathbf{K} = \{1,2, \dots, n\}$, respectively. In this paper, we use i, j, k interchangeably as elements or the order of the elements in the ordered sets.

In the distillation superstructure (see **Figure 8-1**B), each distillation column has one feed, one top stream, and one bottom stream. A mixer is introduced for each distillation column. The inlets of the mixer are connected with the top/bottom streams from upstream distillation columns or the initial feed stream to the superstructure (from the reactor superstructure). Without loss of generality, we assume that there are n final product streams that are enriched in one of the components. However, the proposed model can handle separating the n component mixture into m ($m < n$) final products.

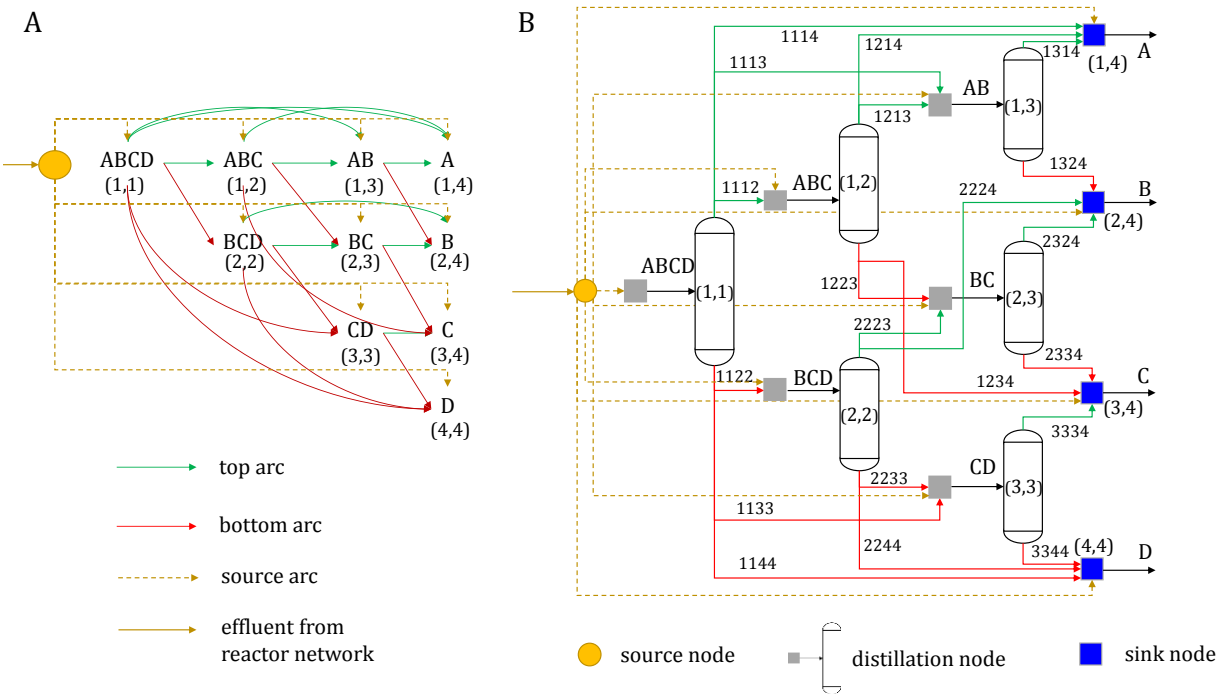


Figure 8-1. Matrix-based superstructure for separating ABCD: (A) matrix representation, (B) corresponding superstructure.

The distillation superstructure is modeled using a network representation, which is based on the matrix notation. There are three types of nodes in the network: the *source nodes* (or simply *sources*), $l \in \mathbf{L}$, representing the initial feeds to the distillation network; the *distillation nodes*, each one of which consists of a pair of a mixer and a distillation column, denoted as $(j,k) \in \mathbf{C} = \{(j,k) | j \leq k, k < n \; \forall j,k\}$; and the *sink nodes* (or simply *sinks*) which are the final products, denoted as $(j,k) \in \mathbf{M} = \{k = n, \forall j,k\}$. Without loss of generality, in this section we assume that there is one source ($\mathbf{L} = \{1\}$), while the generalization to multiple sources will be discussed later.

The connection between nodes is represented by arcs. *Source arcs* (yellow dashed line in **Figure 8-1**) are directed arcs that connect the source to the distillation nodes and sinks. Each distillation node has two types of outgoing arcs that are connected to other nodes. The set of *top arcs*, $\mathbf{A}^T$, representing the connection between the top stream of one distillation column to another distillation column or the final product, is defined as follows:

$$\mathbf{A}^T = \{(j,k,j',k') | j' = j,\ k' > k, \forall (j,k) \in \mathbf{C}, (j',k') \in \mathbf{N}\} \quad (8.1)$$

and the set of bottom arcs, $\mathbf{A}^B$, representing the bottom stream of one column connecting to another column or the final product, is defined as:

$$\mathbf{A}^B = \{(j,k,j',k') | (k'-j') = (k-j),\ k' > k, \forall (j,k) \in \mathbf{C}, (j',k') \in \mathbf{N}\} \quad (8.2)$$

where $\mathbf{N} = \mathbf{C} \cup \mathbf{M}$ is the set of *distillation network nodes* (i.e. the distillation nodes and sinks). From a matrix perspective, $\mathbf{A}^T$ includes connections between two nodes on the same row ($j = j'$) and $\mathbf{A}^B$ includes connections between two nodes on the diagonals of the matrix and its square submatrices ($k' - j' = k - j$). Note that distillation network nodes are ordered based on the matrix notation: node (j,k) is an upstream node with respect to node (j',k') if $k' > k$ and vice versa. Top and bottom arcs are directed arcs connecting an upstream node (j,k) to a downstream node (j',k'). For example, in **Figure 8-1**, green arrows represent all the arcs in $\mathbf{A}^T$ and red arrows are all the arcs in $\mathbf{A}^B$.

Similar to the state equipment network (SEN) based superstructure, the separation tasks are not unique for some distillation nodes (e.g. node (1,1) in **Figure 8-1**) and thus there might be multiple outgoing top/bottom arcs originating from a distillation node. On the other hand, some nodes (e.g. node (3,4)) have more than one incoming arc because the same intermediate/product can be produced by more than one distillation node (nodes (1,2), (2,3) and (2,3)).

For each distillation node, there is a one-to-one correspondence between its matrix indices (i.e. j and k) and the set of postulated components in its feed stream ($\mathbf{I}_{jk}^C$), as follows:

$$\mathbf{I}_{jk}^C = \{i \in \mathbf{I} | j \leq i \leq n - k + j, \forall (j,k) \in \mathbf{C}\} \quad (8.3)$$

Similarly, we can identify the lightest, $\mathbf{I}_{jk}^{L} = \{i \in \mathbf{I} | i = j, \forall (j,k) \in \mathbf{C}\}$, and the heaviest $\mathbf{I}_{jk}^{H} = \{i \in \mathbf{I} | i = n - k + j, \forall (j,k) \in \mathbf{C}\}$ components in each node. This type of correspondence is critically important in determining the potential key components for each separation (see §2.4).

The superstructure is the basis for the proposed model for simple distillation columns (i.e. columns each has exactly one condenser and one reboiler), which will be introduced in the subsequent sections. We consider sloppy split in which the key components are not necessarily consecutive in volatility.

8.2.2 Superstructure Logic and Connectivity

First, we introduce a binary variable, X_{jk}, to denote the selection of a distillation network node (the pair of distillation column and mixer or the final product). $X_{jk} = 1$ if node (j,k) is selected and $X_{jk} = 0$ otherwise. A pair of binary variables ($Y_{jkj'k'}^{T}$ and $Y_{jkj'k'}^{\mathrm{B}}$) are introduced to represent the connectivity in the superstructure. Specifically, $Y_{jkj'k'}^{T} = 1$ if and only if the outgoing top arc of distillation node (j,k) is connected to node (j',k'); while $Y_{jkj'k'}^{B} = 1$ if and only if the outgoing bottom arc of distillation node (j,k) is connected to node (j',k'). If a distillation node is selected ($X_{jk} = 1$), then it must have exactly one outgoing top arc activated in $\mathbf{A}^{T}$ and one outgoing bottom arc activated in $\mathbf{A}^{B}$; while if the node is not selected ($X_{jk} = 0$), then all its outgoing arcs are deactivated,

$$X_{jk} = \sum_{(jkj'k') \in \mathbf{A}^{T}} Y_{jkj'k'}^{T} \qquad (j,k) \in \mathbf{C} \qquad (8.4)$$

$$X_{jk} = \sum_{(jkj'k') \in \mathbf{A}^{B}} Y_{jkj'k'}^{\mathrm{B}} \qquad (j,k) \in \mathbf{C} \qquad (8.5)$$

As pointed out by Caballero, Grossmann [150], unless in some special cases, we do not need to consider a state (i.e. intermediates or products) that can be produced by more than two upstream

distillation columns. Therefore, if a distillation network node is selected, except for node (1,1), then it is connected to one or two upstream distillation nodes,

$$X_{jk} \leq \sum_{(j'k'jk)\in\mathbf{A}^T} Y^T_{j'k'jk} + \sum_{(j'k'jk)\in\mathbf{A}^B} Y^B_{j'k'jk} \qquad (j,k) \in \mathbf{N}, k > 1 \qquad (8.6)$$

$$\theta_{jk} X_{jk} \geq \sum_{(j'k'jk)\in\mathbf{A}^T} Y^T_{j'k'jk} + \sum_{(j'k'jk)\in\mathbf{A}^B} Y^B_{j'k'jk} \qquad (j,k) \in \mathbf{N}, k > 1 \qquad (8.7)$$

where $\theta_{jk} = 1$ for $\{(j,k) \in \mathbf{N} | j = k \text{ or } j = 1\}$ and $\theta_{jk} = 2$ for $\{(j,k) \in \mathbf{N} | j \neq k \text{ and } j \neq 1\}$. In other words, $\theta_{jk} = 2$ for nodes that are not located on the first row or the diagonal of the matrix, while $\theta_{jk} = 1$ otherwise. In **Figure 8-2**B, $\theta_{jk} = 2$ for nodes (2,3), (2,4), (3,4) because these three nodes each can have two incoming arcs activated, one top arc and one bottom arc, from two upstream distillation nodes. For instance, node (2,3) can be simultaneously connected to upstream nodes (1,2) *and* (2,2), while node (1,3) can only be connected to upstream node (1,1) *or* (1,2).

When the flowrate of every postulated component in the stream that couples the distillation superstructure with the reactor superstructure is strictly positive in all feasible solutions, then we know prior to optimization that (i) the source is connected to distillation node (1,1), and (ii) all the sinks must be selected ($X_{jk} = 1, \forall (j,k) \in \mathbf{M}$). However, if one or more of the postulated components in the source can have zero outgoing flow, then these two pieces of information might not be true.

First, the source should be connected to an appropriate distillation network node depending on the component flowrates in the coupling stream. For example, if the postulated components are ABCDE, but the optimization determines that the flowrates of A and E are equal to zero, then the separation should start from the distillation node that separates BCD. Therefore, the source, which contains BCD, should be connected to node (2,3) and thus node (2,3) is the most upstream distillation node. In the extreme case when only one component is present in the source, the

distillation superstructure is bypassed because no separation is needed, and the source is connected to one of the sinks. Therefore, in the context of simultaneous reactor and distillation network synthesis, we must allow the source to be connected to the appropriate node.

To address this, we introduce a binary variable, Y^0_{jk}, which is equal to one if and only if the source is connected to distillation network node (j,k). The source must be connected to exactly one distillation network node:

$$\sum_{(j,k)\in\mathbf{N}} Y^0_{jk} = 1 \tag{8.8}$$

We introduce disaggregated variables ($\acute{F}^0_{ijk}$) for the flowrate from the source (F^0_i),

$$F^0_i = \sum_{(j,k)\in\mathbf{N}} \acute{F}^0_{ijk} \qquad i \in \mathbf{I} \tag{8.9}$$

which (de)activate the corresponding Y^0_{jk}, as follows:

$$\acute{F}^0_{ijk} \le f^{UP}_i Y^0_{jk} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{N} \tag{8.10}$$

When the source is connected to node (j,k), the node must be selected,

$$X_{jk} \ge Y^0_{jk} \qquad (j,k) \in \mathbf{N} \tag{8.11}$$

and if a distillation network node is selected, then it must be connected to the source and/or upstream distillation nodes:

$$X_{jk} \le \sum_{(j'k'jk)\in\mathbf{A}^T} Y^T_{j'k'jk} + \sum_{(j'k'jk)\in\mathbf{A}^B} Y^B_{j'k'jk} + Y^0_{jk} \qquad (j,k) \in \mathbf{N} \tag{8.12}$$

It is important to note that while we do not directly impose constraints on which distillation network nodes the source can or should be connected to, the connection is implicitly constrained via material balances and constraints on product purity, which will be discussed later. To illustrate, a source with components ABCD cannot be connected to a distillation node for the separation of

AB (node (1,3) in **Figure 8-1**) because C and D cannot be directed to the sinks corresponding to C (node (3,4)) and D (node (4,4)), unless the concentrations of C and D in the source are low enough so that the purity requirement can be satisfied. If that is the case, however, our model correctly allows the source to be connected to the distillation node that separates AB.

Second, the sinks should be (de)activated based on the flowrates from the source (or simply source flowrates). If a component has zero flowrate from the source, then it does not present in the distillation superstructure and thus the corresponding sink should be deactivated. To enforce this, we introduce a binary variable, Y_i, which is equal to one if and only if the source flowrate of component i is strictly positive:

$$\delta Y_i \le F_i^0 \le f_i^{UP} Y_i \qquad i \in \mathbf{I} \tag{8.13}$$

where f_i^{UP} is the upper bound on F_i^0, and δ is a small *threshold* flow of component i. Throughout the paper, we use "zero-flow" and "$F_i^0 = 0$" to refer to the situation where $F_i^0 < \delta$. The flowrate of a component determines the selections of some distillation network nodes. The sinks are selected if and only if the corresponding $Y_i = 1$:

$$X_{jk} = Y_i \qquad (j,k) \in \mathbf{M}, i = j \tag{8.14}$$

In addition, when $Y_i = 0$, we deactivate the nodes on the row ($j = i$) and the diagonal ($k - j = n - i$) where the ith sink node is located.

$$X_{jk} \le Y_i \qquad (j,k) \in \mathbf{N}, i = j \tag{8.15}$$

$$X_{jk} \le Y_i \qquad (j,k) \in \mathbf{N}, i = n - k + j \tag{8.16}$$

To illustrate, if $Y_B = 0$ as shown in **Figure 8-2**B, then Eq. (8.15) deactivates nodes on the second row and Eq. (8.16) deactivates node (1,3) which is on the same diagonal as node (2,4). After

deactivating the nodes, the remaining distillation superstructure is equivalent to the superstructure for the separation of ACD. One feasible solution when $Y_B = 0$ is shown in **Figure 8-3**. In particular, ABC is separated into A and C without the presence of B. This separation is possible because $\mathbf{A}^T$ includes the arc that connects nodes (1,2) and (1,4) and $\mathbf{A}^B$ includes the arc that connects nodes (1,2) and (3,4). This type of separation, however, is not admissible in previous approaches.

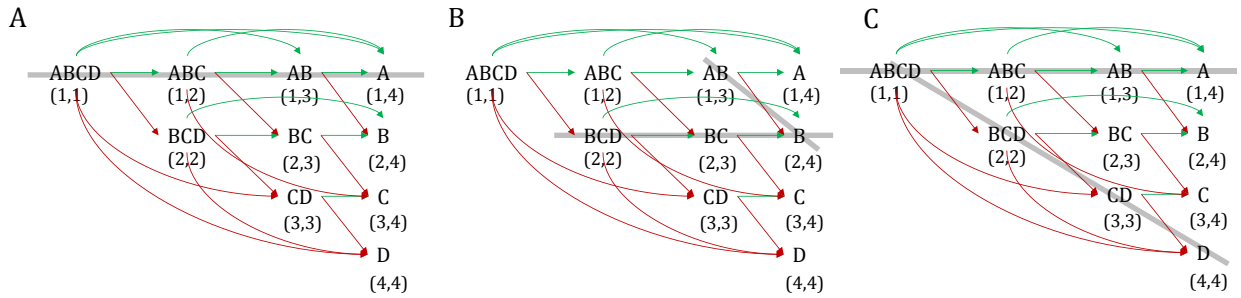


Figure 8-2. Distillation superstructure for the separation of ABCD.

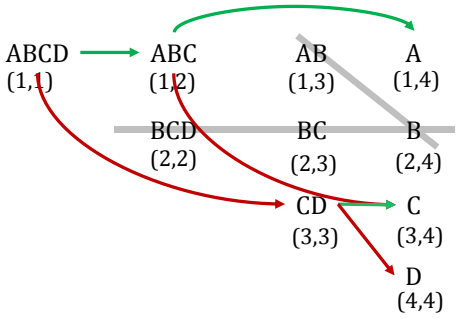


Figure 8-3. One possible configuration of the separation of ABCD with zero-flow of B.

8.2.3 Material Balances

Material balances are introduced for the distillation column in each distillation node to couple the feed flowrate (F_{ijk}) with the distillate (D_{ijk}) and bottom liquid (B_{ijk}) flowrates.

$$F_{ijk} = D_{ijk} + B_{ijk} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \qquad (8.17)$$

Internal material balances are introduced to couple the external flowrates (F_{ijk}, D_{ijk}, and B_{ijk}) to the internal liquid ($L1_{jk}$ and $L2_{jk}$) and vapor ($V1_{jk}$ and $V2_{jk}$) flowrates (see **Figure 8-4**).

$$\sum_{i\in\mathbf{I}} D_{ijk} = V1_{jk} - L1_{jk} \qquad (j,k) \in \mathbf{C} \qquad (8.18)$$

$$\sum_{i\in\mathbf{I}} B_{ijk} = L2_{jk} - V2_{jk} \qquad (j,k) \in \mathbf{C} \qquad (8.19)$$

$$\sum_{i\in\mathbf{I}} F_{ijk} = L2_{jk} - L1_{jk} \qquad (j,k) \in \mathbf{C} \qquad (8.20)$$

These material balances are enforced regardless of the selection of the distillation nodes in the final solution, although they might not be necessary (e.g. if the objective is minimizing total vapor flow). When a distillation node is not selected, all the external flowrates will be forced to be equal to zero as will be discussed later, and thus Eqs. (8.17) – (8.20) will enforce the internal flowrates to be equal to zero. Since the cost of each distillation column is directly related to the internal liquid and vapor flowrate, the distillation column in the node is deactivated if the distillation node is not selected.

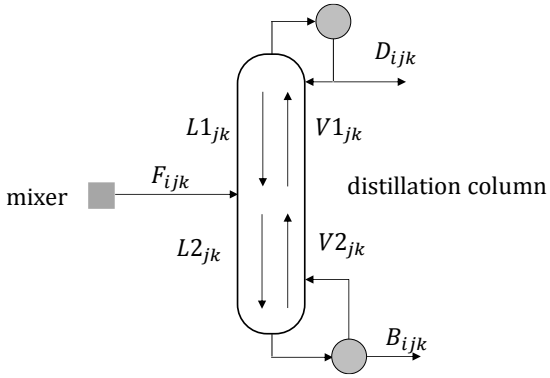


Figure 8-4. Internal and external material flows for the distillation column in node (j,k).

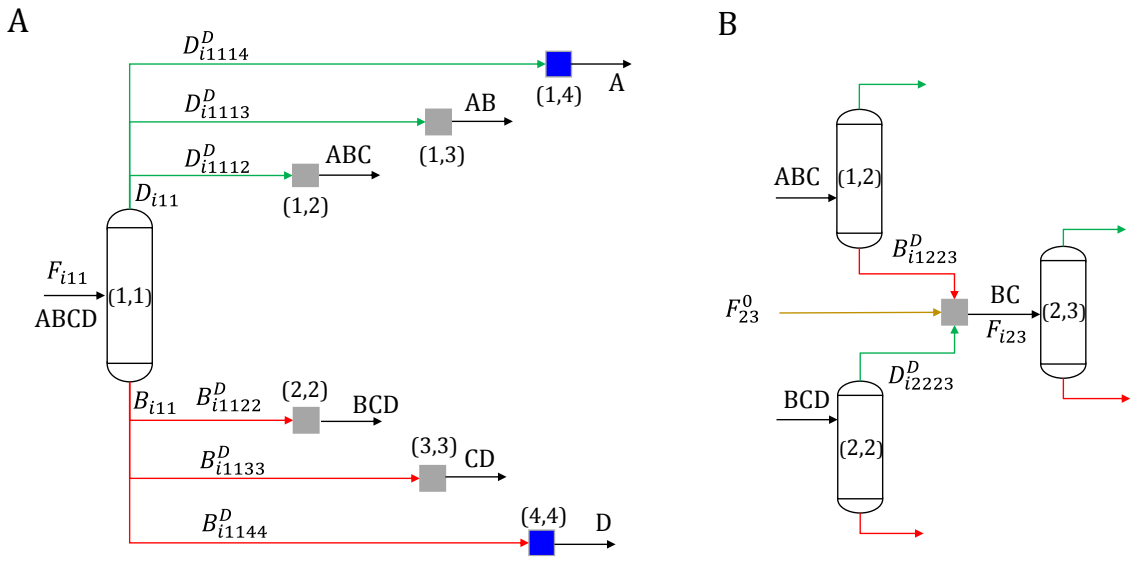


Figure 8-5. Material balances for a source with postulated components ABCD.

Material balances are also introduced to connect the flowrates between different nodes in the network. Recall (**Figure 8-1**A) that a distillation node has outgoing top and bottom arcs. Therefore, we introduce disaggregated variables ($D^D_{ij'k'jk}$ and $B^D_{ij'k'jk}$) to represent the component flows of the top and bottom arcs, which are (de)activated based on the corresponding binary variables ($Y^T_{jkj'k'}$ and $Y^B_{jkj'k'}$),

$$D_{ijk} - \delta^{UP}_{ijk}\left(1 - Y^T_{jkj'k'}\right) \le D^D_{ijkj'k'} \le D_{ijk} \qquad i \in \mathbf{I}, (jkj'k') \in \mathbf{A}^T \qquad (8.21)$$

$$D^D_{ijkj'k'} \le \delta^{UP}_{ijk} Y^T_{jkj'k'} \qquad i \in \mathbf{I}, (jkj'k') \in \mathbf{A}^T \qquad (8.22)$$

$$B_{ijk} - \beta^{UP}_{ijk}\left(1 - Y^B_{jkj'k'}\right) \le B^D_{ijkj'k'} \le B_{ijk} \qquad i \in \mathbf{I}, (jkj'k') \in \mathbf{A}^B \qquad (8.23)$$

$$B^D_{ijkj'k'} \le \beta^{UP}_{ijk} Y^B_{jkj'k'} \qquad i \in \mathbf{I}, (jkj'k') \in \mathbf{A}^B \qquad (8.24)$$

where δ^{UP}_{ijk} and β^{UP}_{ijk} are upper bounds on D_{ijk} and B_{ijk}, respectively.

From a unit-operation standpoint, the inlets of the mixer in each node are streams from upstream distillation columns or the source and the outlet is the feed to the distillation column (see **Figure**

8-5B). The final product streams are the collections of product streams from upstream distillation columns. From a network standpoint, the incoming flow along a distillation network node (F_{ijk}) is equal to the summation of flow for all the incoming top and bottom arcs and the source arc.

$$F_{ijk} = \sum_{(j'k'jk)\in\mathbf{A}^T} D^D_{ij'k'jk} + \sum_{(j'k'jk)\in\mathbf{A}^B} B^D_{ij'k'jk} + F^0_{ijk} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{N} \tag{8.25}$$

According to Eqs. (8.12) and (8.25), distillation network node (j,k) is either the one where the source is connected to ($Y^0_{jk} = 1$ and $F_{ijk} = \acute{F}^0_{ijk} = F^0_i$) or a downstream node ($Y^0_{jk} = 0$ and $\acute{F}^0_{ijk} = 0$) whose inlet flowrate is equal to the summation of disaggregated outgoing flowrates of upstream nodes. To illustrate, if node (1,1) is connected to node (1,2) via a top arc ($Y^T_{1112} = 1$) and it is connected to node (2,2) via a bottom arc ($Y^B_{1122} = 1$), then $D^D_{i1112} = D_{i11}$ and $B^D_{i1122} = B_{i11}$ according to Eqs. (8.21) – (8.24). For nodes (1,2) and (2,2), $F_{i12} = D_{i11}$ and $F_{i22} = B_{i11}$ according to Eq. (8.25).

In addition, the following constraint enforces that if a distillation node is selected, then the feed flowrate must be at least δ',

$$\sum_{i\in\mathbf{I}} F_{ijk} \geq \delta' X_{jk} \qquad (j,k) \in \mathbf{C} \tag{8.26}$$

where δ' can be estimated based on, for example, the sizing limit of a distillation column.

8.2.4 Key Components

The key components for each distillation node are determined based on the separation task that is carried out by the node, which might be unknown prior to optimization. For each distillation node, four types of information are needed to determine the key components: (1) the selection of the node (X_{jk}), (2) the selection of incoming top arcs ($Y^T_{jkj'k'}$), (3) the selection of incoming bottom arcs ($Y^B_{jkj'k'}$), and (4) the presence of components in the network (Y_i).

When the distillation node is selected, the pair of downstream nodes to which the outgoing top and bottom arcs connect determine the separation task and the key components. However, this is valid only if all the postulated components have positive flowrate from the source. Consider node (1,1) in **Figure 8-6**, which is connected to nodes (1,2) and (4,4). If the source flowrates of all the components are strictly positive, then we can identify C as the light key and E as the heavy key. If $F_C^0 = 0$, then in node (1,1), component B, instead of C, is the light key because the separation task becomes ABD/DE.

Therefore, for each feasible connection between nodes (j,k) and (j',k') via a top arc and (j'',k'') via the a bottom arc, there is a set of light key candidates ($\mathbf{I}^{LK}_{j'k'j''k''}$) and a set of heavy key candidates ($\mathbf{I}^{HK}_{j'k'j''k''}$). Note that neither $\mathbf{I}^{LK}_{j'k'j''k''}$ nor $\mathbf{I}^{HK}_{j'k'j''k''}$ is indexed over (j,k) because, in any feasible solution, given a pair of downstream nodes (j',k') and (j'',k''), the upstream node (j,k) is unique. This is because node (j,k) is on the same row as node (j',k') and the same diagonal as node (j'',k''). Sets $\mathbf{I}^{LK}_{j'k'j''k''}$ and $\mathbf{I}^{HK}_{j'k'j''k''}$ can be determined prior to optimization, as follows:

$$\mathbf{I}^{LK}_{j'k'j''k''} = \begin{cases} \{i \in \mathbf{I} | i = j' - k' + n\} & (j' - k' + n + 1 - j'' < 0) \\ \{i \in \mathbf{I} | j' \le i \le j'' - 1\} & (j' - k' + n + 1 - j'' \ge 0) \end{cases} \tag{8.27}$$

$$\mathbf{I}^{HK}_{j'k'j''k''} = \begin{cases} \{i \in \mathbf{I} | i = j'' - 1\} & (j' - k' + n + 1 - j'' < 0) \\ \{i \in \mathbf{I} | j' - k' + n + 1 \le i \le n - k'' + j''\} & (j' - k' + n + 1 - j'' \ge 0) \end{cases} \tag{8.28}$$

Consider a feasible configuration in **Figure 8-6**A where node (1,1) is connected to downstream nodes (1,2) and (4,4). Since $j' - k' + n + 1 - j'' = 1 \ge 0$, $\mathbf{I}^{LK}_{1244} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$ and $\mathbf{I}^{HK}_{1244} = \{\mathrm{E}\}$. There are three light key candidates because of the possibility of zero-flow. Specifically, if $F_C^0 > 0$, then C is the light key; if $F_C^0 = 0$ and $F_B^0 > 0$, then B is the light key; and if $F_C^0 = F_B^0 = 0$ but $F_A^0 > 0$, then A is the light key. On the other hand, E is the only heavy key candidate, regardless

of the flowrate of B, C, and D (A and E must have positive flow for node (1,1) to be selected). Consider another feasible connection of node (2,3) to nodes (2,5) and (4,5). According to Eqs. (8.27) and (8.28), since $j' - k' + n + 1 - j'' = -1 < 0$, $\mathbf{I}_{2545}^{LK} = \{\text{B}\}$ and $\mathbf{I}_{2545}^{HK} = \{\text{D}\}$. In this case, the key components are uniquely determined for distillation node (2,3).

The cases where $j' - k' + n + 1 - j'' < 0$ deserve additional comments. These configurations (i.e. connections) are valid only when one or more of the distributed components have zero flowrate. The key component candidates are unique because (i) the heaviest component among the postulated components for the top downstream node (j', k') and the lightest component among the postulated components for the bottom downstream node (j'', k'') have positive flow, and (ii) the zero-flow distributed components are not key components. To illustrate, consider in **Figure 8-6**B, node (1,1) is connected to nodes (1,4) and (4,4). Eqs. (8.27) and (8.28) yield B as the light key and D as the heavy key. The heaviest postulated component in node (1,4) (i.e. B) must have positive source flowrate because node (1,4) would otherwise be deactivated by Eq. (8.16). On the other hand, the lightest postulated component in node (4,4) (i.e. D) must have positive flow for the same reason. The distributed components that are not present in the feed (component C in this case) are not candidate key components.

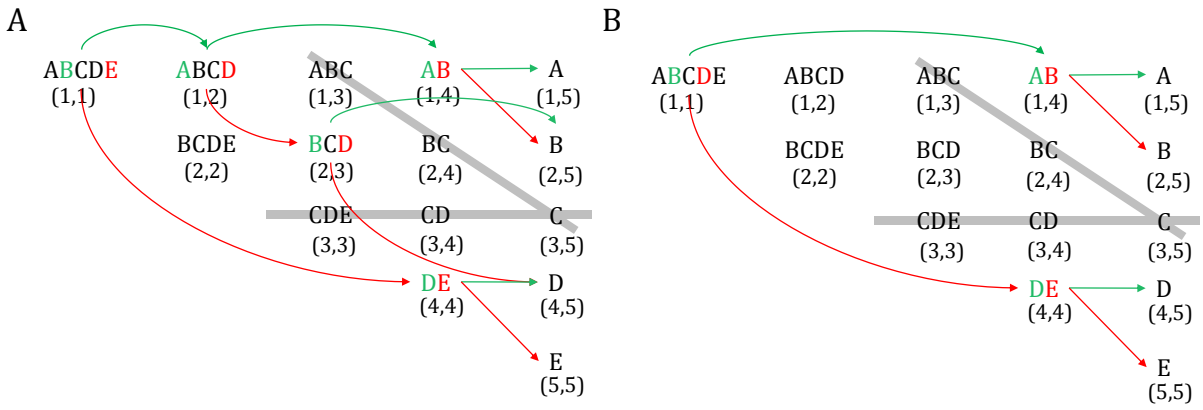


Figure 8-6. Two feasible solutions of separating a five-component mixture with zero-flow of C.

For each distillation node, we introduce a pair of binary variables, Y_{ijk}^{LK} and Y_{ijk}^{HK}, which are equal to one if i is the light/heavy key in $(j,k) \in \mathbf{C}$. Since the key components can be determined by the aforementioned four types of information, Y_{ijk}^{LK} and Y_{ijk}^{HK} are determined based on the four binary variables (X_{jk}, $Y_{jkj'k'}^{T}$, $Y_{jkj'k'}^{B}$, and Y_i). First, there are one light key and one heavy key if and only if the distillation node is selected:

$$\sum_i Y_{ijk}^{LK} = X_{jk} \qquad (j,k) \in \mathbf{C} \tag{8.29}$$

$$\sum_i Y_{ijk}^{HK} = X_{jk} \qquad (j,k) \in \mathbf{C} \tag{8.30}$$

If the distillation node is selected, then the light and heavy keys can only be selected from the key component candidate sets ($\mathbf{I}_{j'k'j''k''}^{LK}$ and $\mathbf{I}_{j'k'j''k''}^{HK}$):

$$\sum_{i \in \mathbf{I}_{j'k'j''k''}^{LK}} Y_{ijk}^{LK} \geq Y_{jkj'k'}^{T} + Y_{jkj''k''}^{B} - X_{jk} \quad (jkj'k') \in \mathbf{A}^T, (jkj''k'') \in \mathbf{A}^B \tag{8.31}$$

$$\sum_{i \in \mathbf{I}_{j'k'j''k''}^{HK}} Y_{ijk}^{HK} \geq Y_{jkj'k'}^{T} + Y_{jkj''k''}^{B} - X_{jk} \quad (jkj'k') \in \mathbf{A}^T, (jkj''k'') \in \mathbf{A}^B \tag{8.32}$$

In addition, the key component must have positive flow from the source:

$$Y_{ijk}^{LK} \leq Y_i \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \tag{8.33}$$

$$Y_{ijk}^{HK} \leq Y_i \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \tag{8.34}$$

Finally, the light key is the heaviest positive-flow component among the light key candidates and the heavy key is the lightest positive-flow component among the heavy key candidates:

$$Y_{ijk}^{LK} \leq 1 - Y_{i'} + (2X_{jk} - Y_{jkj'k'}^{T} - Y_{jkj''k''}^{B}) \qquad i \in \mathbf{I}_{j'k'j''k''}^{LK}, i' \in \mathbf{I}_{j'k'j''k''}^{LK}$$
$$(jkj'k') \in \mathbf{A}^T, (jkj''k'') \in \mathbf{A}^B, i' > i \tag{8.35}$$

$$Y_{ijk}^{HK} \le 1 - Y_{i'} + (2X_{jk} - Y_{jkj'k'}^{T} - Y_{jkj''k''}^{B}) \qquad i \in \mathbf{I}_{j'k'j''k''}^{LK}, i' \in \mathbf{I}_{j'k'j''k''}^{LK}$$

$$(jkj'k') \in \mathbf{A}^{T}, (jkj''k'') \in \mathbf{A}^{B}, i' < i \qquad (8.36)$$

Note that if the distillation node is not selected ($X_{jk} = 0$), then both Y_{ijk}^{LK} and Y_{ijk}^{HK} are equal to zero and Eqs. (8.31) – (8.36) are trivially satisfied.

8.2.5 Underwood Equations

We use the Underwood equations [63,171] to calculate the minimum vapor flowrate in the rectifying and stripping section ($V1_{jk}^{min}$ and $V2_{jk}^{min}$). The actual vapor flowrate can be estimated as 1.2 times the minimum vapor flowrate [173].

It is important to understand that the active roots in the Underwood equations depend on the key components, which are not necessarily consecutive in volatility. The key components of some distillation columns cannot be specified because the separation task carried out in a distillation column is unknown prior to optimization. There is an additional challenge to use the Underwood equations when some components have zero flow.

We use the concepts and methodologies from the previous chapter to develop a modified Underwood method, which is tailored to be used in the approach proposed in the present paper.

For each distillation node, a set of possible active roots are defined as $\mathbf{I}_{jk}^{R} = \{i \in \mathbf{I} | j \le i \le n - 1 - k + j, (j,k) \in \mathbf{C}\}$ so that the roots are connected to a subset of postulated components for that node. For example, for distillation node (1,1) in **Figure 8-1**, $\mathbf{I}_{11}^{R} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$ represents the three possibly active roots, regardless of the selection of separation tasks in the final solution. For node (3,3), $\mathbf{I}_{33}^{R} = \{\mathrm{C}\}$ represents the only active root for the separation of CD. The Underwood equations are written for all the distillation nodes, whether they are selected or not,

$$\sum_{i\in\mathbf{I}} \frac{\alpha_i F_{ijk}}{\alpha_i - \phi_{i'jk}} = 0 \qquad (j,k) \in \mathbf{C}, i' \in \mathbf{I}_{jk}^R, \tag{8.37}$$

$$\sum_{i\in\mathbf{I}} \frac{\alpha_i D_{ijk}}{\alpha_i - \phi_{i'jk}} = V1_{jk}^{min} \qquad (j,k) \in \mathbf{C}, i' \in \mathbf{I}_{jk}^R \tag{8.38}$$

$$\sum_{i\in\mathbf{I}} \frac{-\alpha_i B_{ijk}}{\alpha_i - \phi_{i'jk}} = V2_{jk}^{min} \qquad (j,k) \in \mathbf{C}, i' \in \mathbf{I}_{jk}^R \tag{8.39}$$

where α_i is the relative volatility of component i with respect to the heaviest (i.e. nth) component in $\mathbf{I}$ and $\phi_{i'jk}, i' \in \mathbf{I}_{jk}^R$ are the possibly active roots of the Underwood equations for distillation node (j,k).

The active roots are bounded by the relative volatilities of the key components,

$$\sum_i (\alpha_i + \epsilon) Y_{ijk}^{HK} \leq \phi_{i'jk} \leq \sum_i (\alpha_i - \epsilon) Y_{ijk}^{LK} \qquad (j,k) \in \mathbf{C}, i' \in \mathbf{I}_{jk}^R \tag{8.40}$$

where ϵ is a very small positive number representing the minimum absolute difference between the roots and relative volatilities to avoid numerical difficulties. According to Eqs. (8.29) and (8.30), when $X_{jk} = 0$, both Y_{ijk}^{HK} and Y_{ijk}^{HK} are equal to zero, and thus Eq. (8.40) will enforce $\phi_{i'jk} = 0, \forall i' \in \mathbf{I}_{jk}^R$.

To determine the values of $\phi_{i'jk}$, additional binary variables are introduced. First, binary variable $Z_{ijk} = 1$ if and only if i is a distributed component, which is more volatile than the heavy key but less volatile than the light key. For example, for distillation node (1,1) in **Figure 8-1**, if $Y_{A11}^{LK} = Y_{D11}^{HK} = 1$, then $Z_{B11} = Z_{C11} = 1$ (B and C are the distributed components). The Z_{ijk} binary should satisfy:

$$Z_{ijk} = \sum_{i''\leq i-1} Y_{i''jk}^{LK} - \sum_{i'\leq i} Y_{i'jk}^{HK} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \tag{8.41}$$

Binary variable, X_{ijk}^{R}, is introduced to determine if ϕ_{ijk} and $\phi_{i+1,jk}$ are distinct roots. If $X_{ijk}^{R} = 1$, then $\phi_{ijk} \neq \phi_{i+1,jk}$, implying that they are two distinct roots. X_{ijk}^{R} can be calculated based on Z_{ijk} and binary Y_i,

$$X_{ijk}^{R} = Z_{i+1,jk} Y_{i+1} \qquad (j,k) \in \mathbf{C}, i \in \mathbf{I}_{jk}^{RD} \tag{8.42}$$

where $\mathbf{I}_{jk}^{RD} = \{i \in \mathbf{I} | j \leq i \leq n-2-k+j, (j,k) \in \mathbf{C}\}$.

Nonnegative variable Δ_{ijk} represents the difference between ϕ_{ijk} and $\phi_{i+1,jk}$,

$$\phi_{ijk} - \phi_{i+1,jk} = \Delta_{ijk} \qquad (j,k) \in \mathbf{C}, i \in \mathbf{I}_{jk}^{RD} \tag{8.43}$$

which is (de)activated by the corresponding X_{ijk}^{R} as follows:

$$\epsilon' X_{ijk}^{R} \leq \Delta_{ijk} \leq (\alpha_{jk}^{max} - \alpha_{jk}^{min}) X_{ijk}^{R} \qquad (j,k) \in \mathbf{C}, i \in \mathbf{I}_{jk}^{RD} \tag{8.44}$$

where ϵ' is a positive number representing the minimum absolute difference between two distinct roots, and α_{jk}^{max} and α_{jk}^{min} are the relative volatility of the most and least volatile component in $\mathbf{I}_{jk}^{C}$, both of which can be obtained prior to optimization.

The following constraint ensures that no root is equal to the relative volatilities:

$$\left(\phi_{ijk} - \alpha_{i'}\right)^2 \geq \epsilon^2 \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C}, i' \in \mathbf{I}_{jk}^{RD}, \tag{8.45}$$

To illustrate, consider distillation node (1,1) in **Figure 8-1** ($\mathbf{I}_{11}^{R} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$). If the optimization determines that the separation task is ABC/CD, then $Y_{B11}^{LK} = Y_{D11}^{HK} = 1$ and $\phi_{i'jk}$ is bounded by α_B and α_D according to Eq. (8.40). Meanwhile, Eq. (8.41) enforces that $Z_{C11} = 1$. Assume $Y_i = 1$ for all $i \in \mathbf{I}$, then, according to Eq. (8.42), $X_{A11}^{R} = 0$ and $X_{B11}^{R} = 1$. Therefore, only Δ_{B11} is strictly positive based on Eq. (8.44). In the final solution, $\phi_{A11} = \phi_{B11} > \phi_{C11}$ is enforced and the values of ϕ are determined by the Underwood equations (Eq. (8.37) – (8.39)).

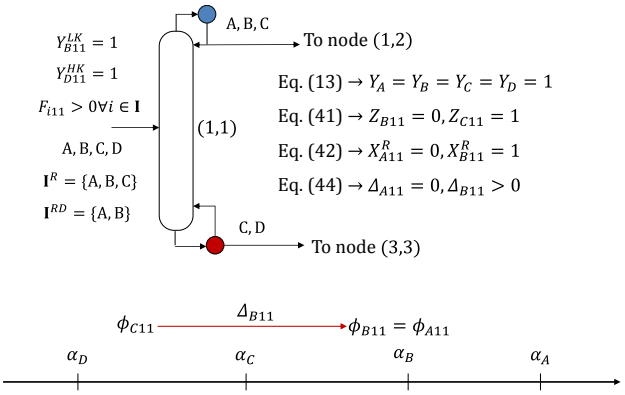


Figure 8-7. An example of how to determine the set of active roots and root loci.

8.2.6 Purity and Recovery

In general, the purity and recovery of the final products are specified. In the sinks, purity of the corresponding product must be at least ρ_i:

$$F_{ijk} \geq \rho_i \sum_{i'} F_{i'jk} \qquad (j,k) \in \mathbf{M}, i = j \qquad (8.46)$$

The minimum recovery of final product i (γ_i) is enforced as follows:

$$F_{ijk} \geq \gamma_i F_i^0 \qquad (j,k) \in \mathbf{M}, i = j \qquad (8.47)$$

Although not necessary, the proposed approach also allows us to enforce minimum recovery of key components (γ_{jk}^{LK} and γ_{jk}^{HK} for light and heavy keys) for each distillation node:

$$D_{ijk} \geq \gamma_{jk}^{LK} F_{ijk} - \delta_{ijk}^{UP}(1 - Y_{ijk}^{LK}) \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \qquad (8.48)$$

$$B_{ijk} \geq \gamma_{jk}^{HK} F_{ijk} - \beta_{ijk}^{UP}(1 - Y_{ijk}^{HK}) \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \qquad (8.49)$$

where the constraints on D_{ijk}/B_{ijk} are relaxed when i is not the light/heavy key components.

In addition, components lighter than the light key should not be present at the bottom and components heavier than the heavy key should not present at the top:

$$B_{ijk} \leq \beta_{ijk}^{UP} \sum_{i' \leq i} Y_{i'jk}^{LK} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \qquad (8.50)$$

$$D_{ijk} \leq \delta_{ijk}^{UP} \sum_{i' \geq i} Y_{i'jk}^{HK} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{C} \qquad (8.51)$$

Eqs. (8.48) – (8.51) are formulated so that they are valid regardless of the selection of nodes. When the nodes are not selected, the corresponding flowrates are deactivated by material balances and these inequalities are trivially satisfied.

8.2.7 Objective Functions

The objective is to minimize the sum of vapor flowrates of all the distillation columns ($\sum_{(j,k)\in\mathbf{C}} V1_{jk}$), which is an excellent surrogate of the total distillation cost. The optimization model (M1) can then be expressed as follows,

$$\min \sum_{(j,k)\in\mathbf{C}} V1_{jk} \qquad (M1)$$

$$s.t. Eqs.(8.1) - (8.51)$$

However, it is possible to incorporate the actual capital and operating cost of the distillation columns in the objective function. If that is the case, model M1 will include additional constraints such as the Fenske equation [62] for calculating the number of trays and the fair correlation [174] for calculating the column area. When the proposed model is used as a submodule in process synthesis, the objective becomes minimizing the total annualized cost of the overall process, which includes costs from the distillation network and other subsystems. Model M1 should include constraints for all the subsystems of interest.

8.3 Remarks and Extensions

8.3.1 Positive Feed Flowrate

The MINLP model presented in Section 2 can handle variable, including zero, flowrate of the postulated components from the source. If we know a priori that thee flowrates for all the postulated components are strictly positive, the model can be simplified substantially.

First, the source is connected only to node (1,1). As a result, Eqs. (8.8) – (8.16) can be removed. Alternatively, we only need to set $X_{11} = 1$ and $F_i^0 = F_{i11}$ to link the source to distillation node (1,1) and set $X_{jk} = 1$ for all $k = n$ to activate all the sinks.

Second, the key components can be determined in a more straightforward way. For every feasible connection between upstream node (j, k) and downstream nodes (j', k') and (j'', k''), the light and heavy keys can be uniquely defined:

$$\mathbf{I}_{j''}^{LK} = \{i \in \mathbf{I} | i = j'' - 1\} \tag{8.52}$$

$$\mathbf{I}_{j'k'}^{HK} = \{i \in \mathbf{I} | i = j' - k' + n + 1\} \tag{8.53}$$

which allows us to remove Eqs. (8.33) – (8.36).

Finally, a redundant constraint is introduced,

$$Y_{jkj'k'}^{T} + Y_{jkj''k''}^{B} \leq 1 \, (jkj'k') \in \mathbf{A}^T, (jkj''k'') \in \mathbf{A}^B, j'' > n - k' + j' + 1 \tag{8.54}$$

to deactivate separations such as ABC into A and C. Note that this separation is already infeasible because material balances and the purity constraint (Eq. (8.46)) forbid B from entering the sinks corresponding to A and C. Thus, Eq. (8.54) is redundant. Nevertheless, adding it can tighten the model.

8.3.2 Integration with Reactor Network Synthesis

So far, we have assumed that only one stream connects the reactor and distillation superstructures, which leads to one source. The source flowrates (F_i^0) are the linking variables that couple the two superstructures. However, it is not uncommon that the reactor superstructure produces more than one effluent stream each with different components. There are two approaches to address this with the assumption of one coupling stream (see **Figure 8-8**):

(1) Combine all the effluent streams into one stream, or

(2) Construct a distillation superstructure for each effluent stream.

The main disadvantage of approach (1) is the unnecessary mixing introduced by combining streams with different components. Consider two effluent streams from the reactor superstructure. Stream I has three components (ABC) and stream II has two components (BC). If the direct sequence is selected, then combining ABC with BC and then separate back into A and BC is clearly suboptimal.

The main disadvantage of approach (2) is the lack of interactions among distillation superstructures. Consider the same example with two effluent streams. Since two distillation superstructures are introduced, there are at least three distillation columns in any feasible solution. However, in reality, the distillation column that separates BC in the two superstructures can be combined into one so that only two distillation columns are needed.

The "one source" assumption in the proposed model can be relaxed so that multiple sources can be considered, which is possible because of the way we model the distillation superstructure. In the modified approach, each source is assigned to one distillation network node in the same way

as in Section 2.2. Specifically, given a set of source nodes (**L**), each $l \in \mathbf{L}$ has its own Y^0_{jkl}, F^0_{il}, and $\acute{F}^0_{ijkl}$, which are the counterparts of Y^0_{jk}, F^0_i, and $\acute{F}^0_{ijk}$. Eqs. (8.8) – (8.12) are modified as follows:

$$\sum_{(j,k)\in\mathbf{N}} Y^0_{jkl} = 1 \qquad l \in \mathbf{L} \tag{8.55}$$

$$F^0_{il} = \sum_{(j,k)\in\mathbf{N}} \acute{F}^0_{ijkl} \qquad i \in \mathbf{I}, l \in \mathbf{L} \tag{8.56}$$

$$\acute{F}^0_{ijkl} \le f^{UP}_{il} Y^0_{jkl} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{N}, l \in \mathbf{L} \tag{8.57}$$

$$X_{jk} \ge Y^0_{jkl} \qquad (j,k) \in \mathbf{N}, l \in \mathbf{L} \tag{8.58}$$

$$X_{jk} \le \sum_{(j'k'jk)\in\mathbf{A}^T} Y^T_{j'k'jk} + \sum_{(j'k'jk)\in\mathbf{A}^B} Y^B_{j'k'jk} + \sum_l Y^0_{jkl} \qquad (j,k) \in \mathbf{N} \tag{8.59}$$

Eqs. (8.55) – (8.57) connect each source to one distillation network node; Eq. (8.68) ensures that the distillation network node is selected if it is connected to any sources; and Eq. (8.59) enforces that if a distillation network node is selected, then it should be connected to the source and/or upstream distillation nodes.

Binary variable Y_i previously representing whether a component has positive outgoing flow from the source is replaced by binary Y_{ijk}, which is equal to one if and only if $F_{ijk} \ge \delta$:

$$Y_{ijk}\delta \le F_{ijk} \le Y_{ijk} f^{UP}_{ijk} \qquad i \in \mathbf{I}, (j,k) \in \mathbf{N} \tag{8.60}$$

Binary Y_{ijk} replaces Y_i in all the relevant constraints in Section 2. For instance, the (de)activation of the sinks (Eq. (8.14)) is now determined by Y_{ijk} for $i = j$ and $k = n$.

To illustrate, consider the examples in **Figure 8-9** where two streams from the reactor superstructure are sent to the distillation superstructure. In **Figure 8-9**A, these two streams are both connected to node (1,1) so that node (1,1) contains all four postulated components. This configuration is admissible in approach (1). In **Figure 8-9**B and C, the source containing BD is

connected to node (2,2) and component B is absent in node (1,2) but not (2,2). The configuration in **Figure 8-9**B can be obtained from approach (2). However, the configuration in **Figure 8-9**C is feasible only when we allow the sources to be connected to different nodes in one superstructure.

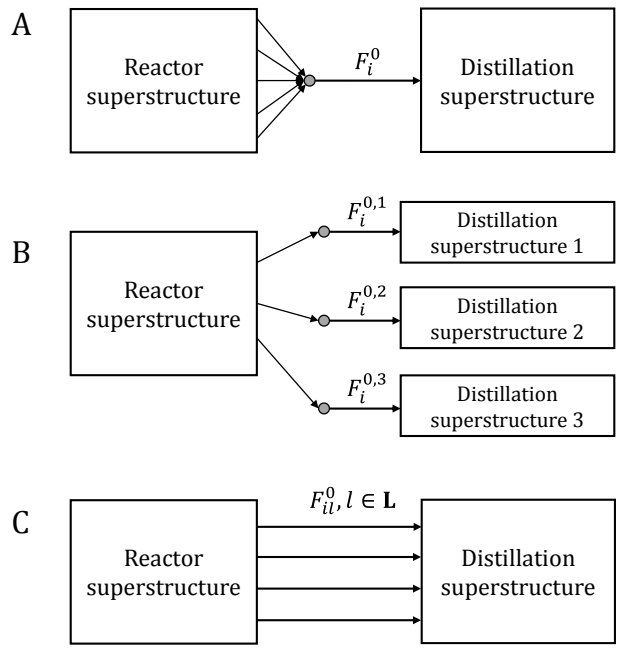


Figure 8-8. Three approaches for coupling the reactor and distillation superstructures.

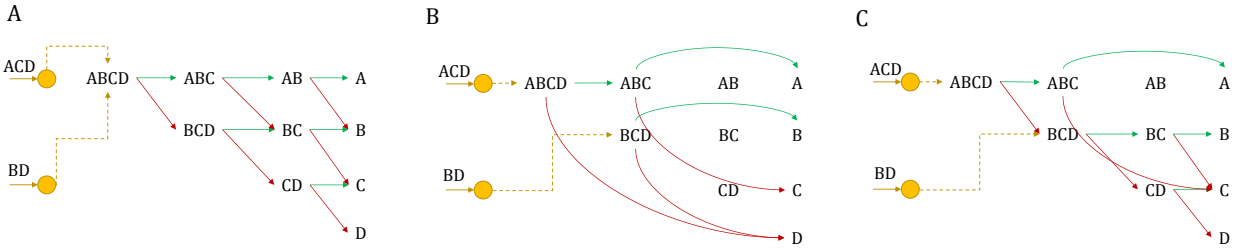


Figure 8-9. Three feasible configurations when the effluents from the reactor superstructure are sent to different nodes.

8.4 Applications

We study the simultaneous synthesis of reactor and distillation networks using the superstructure shown in **Figure 8-10** with products B and C, byproduct D, and reactant A. The reactor superstructure contains three reactors (R1, R2, and R3): R1 and R2 are two alternative reactors

(i.e. only one should be selected) producing C and D via two different reactions (RXN1 and RXN2). RXN1 is a conventional liquid-phase equilibrium reaction where A is converted to C and D, and thus the effluent of R1 contains components A, C, and D. RXN2 is a novel reaction that produces C and D from A in the presence of catalyst F. Compared to RXN1, RXN2 leads to a higher yield of product C and 100% conversion A, but is more expensive. Therefore, it is unknown which reactor would be preferred based on the economics. In Reactor R3, D and B are produced by reacting A and E. Since A is fully converted and E is in a different phase, only B and D appear in the effluent stream for distillation. Therefore, the reactor superstructure has two effluent streams that are sent to the distillation superstructure: one stream from reactor R1 or R2, and the other from reactor R3. Instead of mixing the two effluent streams or use two separate distillation superstructures, we construct only one distillation superstructure and let the optimization determine where to introduce the effluents (i.e. sources). In the distillation superstructure, nodes are introduced based on the four postulated components. The minimum recovery of the key components is 0.99 at each distillation node and the product purity must be at least 99%. The relative volatilities are the same as those in the previous example. The objective is to minimize the annualized cost of the overall process, including the cost of raw materials, reactor costs, distillation costs, minus the revenues of byproduct sales. The price of raw materials and byproducts (π_i), cost of reactors and distillation column (ω), and other specifications are shown in **Figure 8-10**, while modeling equations for the reactor superstructure can be found in the supplementary material.

The resulting MINLP model includes 1364 constraints and 621 variables (168 binaries), and it is solved to global optimality in 1057 seconds using BARON. In the optimal solution, R1 is selected and thus components A, C, and D appear in the reactor effluent, which is connected to node (1,1). The effluent of R3 containing components B and D is connected to node (2,2). In the distillation

superstructure (see **Figure 8-11**), the separation carried out in distillation node (1,1) is AC/D. The top product AC is further separated into A and C in node (1,2). The bottom product CD, however, is mixed with BD from the effluent of R2 in node (2,2) and thus the separation carried out in node (2,2) is B/CD. CD is further separated into C and D in node (3,3).

For comparison, we resolve the model by requiring R2 to be selected. The optimum objective function value is about 4% worse than that when R1 is selected. Instead of mixing the two reactor effluents, the optimization determines that the effluent of R2, which contains C and D, should be sent to node (3,3) and the effluent of R3 containing B and D should be sent to node (2,2).

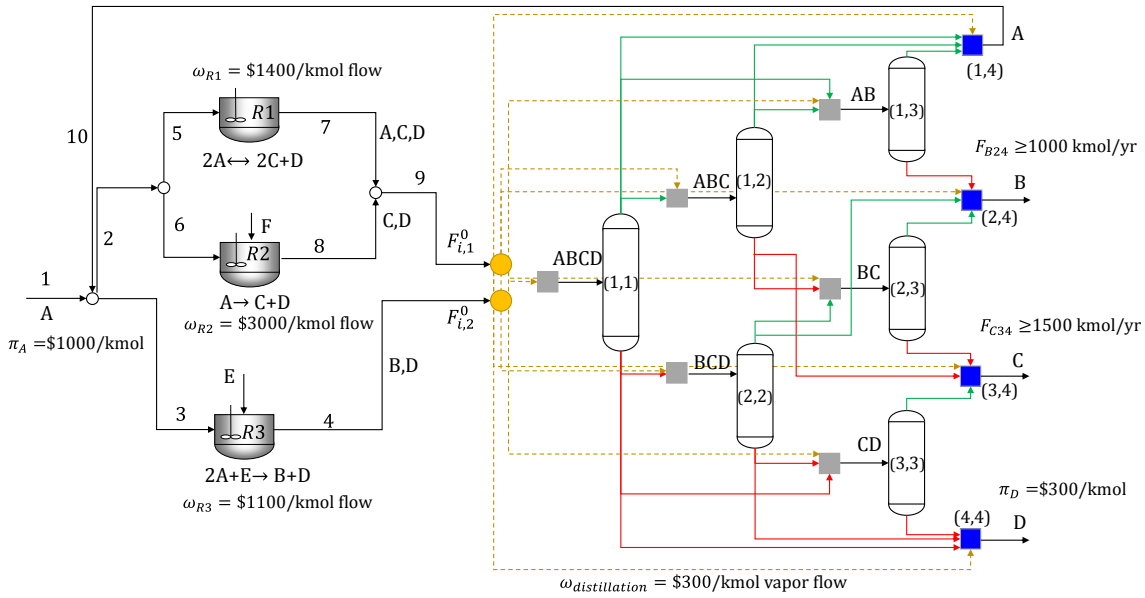


Figure 8-10. The reactor and distillation superstructure for the production of B and C.

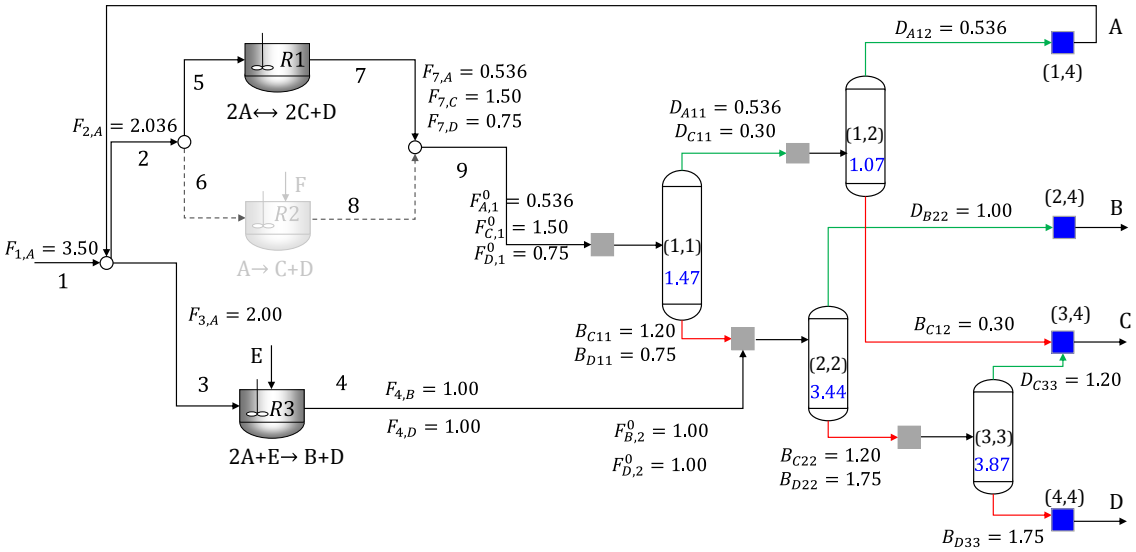


Figure 8-11. Optimal reactor and distillation configuration.

8.5 Conclusions

We proposed a novel approach for the synthesis of distillation sequences when one or more of the postulated components can have zero flow. Furthermore, the proposed approach considers multiple streams that couple the reactor and distillation superstructure, which is more favorable compared to the existing approaches where one coupling stream is assumed. The connections between the source(s) and distillation columns in the superstructure are automatically determined by the optimization, facilitating the simultaneous synthesis of reactor and distillation networks, although simplifications can be made when the source flowrates of all the postulated components are strictly positive. We generated the superstructure based upon the matrix representation and introduced binary variables to determine the selection of distillation columns and separation tasks (i.e. key components). To account for zero component flows, we introduced binary variables, which are essential for determining the key components and the active roots in the Underwood equations. Since sloppy splits are considered, a possible future direction is to extend the proposed approach to the synthesis of complex, thermally coupled distillation sequences.

Chapter 9
Conclusions

9.1 Contributions and Future Work

Heat integration: The main contribution of this thesis to the field is to introduce a set of optimization approaches for simultaneous process synthesis and heat integration. The concept of unclassified streams has never been studied before because all the previous approaches, including those for simultaneous process synthesis and heat integration, assume that stream classifications are always given. However, as mentioned in Chapter 2, it is not always the case, and in our humble opinion, any "simultaneous" approaches should be able to handle (1) variable flowrates, (2) variables inlet/outlet temperatures, and (3) unknown stream classifications. Therefore, to some extent, our approaches are the first to truly allow simultaneous process synthesis and heat integration. The simultaneous utility and area targeting approach in Chapter 3 is also an important contribution that allows us to go beyond previous approaches where only utility target is considered. The proposed approach offers a way to take capital investment into consideration without designing the heat exchanger network, which is a computationally expensive exercise. Finally, we extend the scope to simultaneous process and heat exchanger network synthesis. The heat exchanger network is designed and optimized along with the process, which may include classified and unclassified streams.

The way we deal with unclassified streams should be applicable to most classic heat integration approaches such as Duran and Grossmann [48] and Yee and Grossmann [34]. Therefore, a possible future direction is to extend some of these powerful classic approach to handle unclassified streams.

PWL fitting: We proposed a novel optimization approach for fitting discrete data points using piecewise linear approximation functions. Compared to existing MINLP approaches, ours is more efficient because it is MILP-based. One interesting extension is to use a piecewise linear function to bound the point-wise approximation error from a continuous univariate function. By slightly modifying the constraints, we can conveniently construct (convex/concave) over/under-estimator of a function with the fewest segments. This can potentially be useful for the community of global optimization.

While the scope of the work is currently limited to fitting univariate (i.e. one dimensional) data points, a future direction can be extending it to fitting multivariate data points. The main challenge is generalizing Theorem 1 to higher dimensions. In my opinion, it is possible to do so because the geometric insight used to derive Theorem 1 should have its analogy in higher dimensions.

Distillation column design: The graphical-inspired math program provides a new approach to design distillation columns using a simple yet accurate surrogate of thermodynamics. On one hand, the proposed method is more flexible than its graphical counterpart because it is completely equation oriented. On the other hand, it is simpler and more tractable than the rigorous approaches because the highly nonlinear thermodynamic model is replaced by "cheap" piecewise linear functions. It is interesting to point out that this is not the first graphical-inspired model we proposed. In fact, the heat integration model in Chapter 3 also has its graphical origin (i.e. the composite curves).

Going beyond distillation column design, we should continue exploring the opportunity to develop optimization counterparts of graphical methods in other areas. For example, graphical approaches similar to the McCabe and Thiele method have been successfully applied to designing multistage

extraction units, which means that developing a "graphical-inspired" optimization model for extractions should be straightforward.

Distillation network synthesis: The proposed approaches depart from previous methods by emphasizing how the distillation sequencing exercise is affected by considering simultaneous reactor and distillation network synthesis. Both the unit model (i.e. modified Underwood method) and the network model (i.e. matrix-based representation) are tailored to consider zero component flows at the reactor effluents. Similar to the SEN type of formulation, the proposed approach allows the selection of separation tasks in each unit, which is crucial because the tasks can vary depending on the component flows.

Among all the works presented in the thesis, this is probably the most extendable and "open" one. On one hand, since sloppy split is already accounted for, it is natural to consider the design of thermally coupled distillation sequences. On the other hand, the proposed approach assumes that the effluent streams should each matched to one distillation column inlet. In other words, each source should have one source arc active in the solution, which is enforced via binary variables. However, it is highly possible that the optimal strategy is to split the effluent streams and send them to different distillation columns in the network. It will be interesting to generalize our method to consider that. One thing to keep in mind is the tractability of the models since both extensions/generalizations will make the difficult problem more difficult.

The distillation network synthesis model can be combined with the proposed heat integration models, and when combined with a suitable reactor network synthesis model, eventually we can enable simultaneous reactor and distillation network synthesis and heat integration.

Reference

1. Kocis GR, Grossmann IE. A Modeling and Decomposition Strategy for the Minlp Optimization of Process Flowsheets. *Computers & Chemical Engineering.* 1989;13(7):797-819.
2. Achenie LKE, Biegler LT. A Superstructure Based Approach to Chemical Reactor Network Synthesis. *Computers & Chemical Engineering.* 1990;14(1):23-40.
3. Floudas CA, Ciric AR, Grossmann IE. Automatic Synthesis of Optimum Heat-Exchanger Network Configurations. *Aiche J.* 1986;32(2):276-290.
4. Floudas CA, Paules GE. A Mixed-Integer Nonlinear-Programming Formulation for the Synthesis of Heat-Integrated Distillation Sequences. *Computers & Chemical Engineering.* 1988;12(6):531-546.
5. Kokossis AC, Floudas CA. Optimization of Complex Reactor Networks .2. Nonisothermal Operation. *Chem Eng Sci.* 1994;49(7):1037-1051.
6. Lakshmanan A, Biegler LT. Synthesis of optimal chemical reactor networks. *Ind Eng Chem Res.* 1996;35(4):1344-1353.
7. Novak Z, Kravanja Z, Grossmann IE. Simultaneous synthesis of distillation sequences in overall process schemes using an improved MINLP approach. *Comput Chem Eng.* 1996;20(12):1425-1440.
8. Schweiger CA, Floudas CA. Optimization framework for the synthesis of chemical reactor networks. *Ind Eng Chem Res.* 1999;38(3):744-766.
9. Yee TF, Grossmann IE. Simultaneous-Optimization Models for Heat Integration .2. Heat-Exchanger Network Synthesis. *Comput Chem Eng.* 1990;14(10):1165-1184.
10. Ponce-Ortega JM, Jimenez-Gutierrez A, Grossmann IE. Optimal synthesis of heat exchanger networks involving isothermal process streams. *Comput Chem Eng.* 2008;32(8):1918-1942.
11. Hasan MMF, Jayaraman G, Karimi IA, Alfadala HE. Synthesis of Heat Exchanger Networks with Nonisothermal Phase Changes. *Aiche J.* 2010;56(4):930-945.
12. Wu W, Henao CA, Maravelias CT. A superstructure representation, generation, and modeling framework for chemical process synthesis. *AIChE Journal.* 2016;62(9):3199-3214.
13. Linnhoff B, Flower JR. Synthesis of heat exchanger networks: I. Systematic generation of energy optimal networks. *AIChE Journal.* 1978;24(4):633-642.
14. Furman KC, Sahinidis NV. A critical review and annotated bibliography for heat exchanger network synthesis in the 20th century. *Industrial & Engineering Chemistry Research.* 2002;41(10):2335-2370.

15. Morar M, Agachi PS. Review: Important contributions in development and improvement of the heat integration techniques. *Computers & Chemical Engineering.* 2010;34(8):1171-1179.

16. Klemeš JJ, Kravanja Z. Forty years of Heat Integration: Pinch Analysis (PA) and Mathematical Programming (MP). *Current Opinion in Chemical Engineering.* 2013;2(4):461-474.

17. Linnhoff B, Hindmarsh E. The pinch design method for heat exchanger networks. *Chem Eng Sci.* 1983;38(5):745-763.

18. Homannm EC. *Optimum Networks for Heat Exchange*. Los Angeles, CA, University of Southern California; 1971.

19. Cerda J, Westerberg AW, Mason D, Linnhoff B. Minimum utility usage in heat exchanger network synthesis A transportation problem. *Chem Eng Sci.* 1983;38(3):373-387.

20. Papoulias SA, Grossmann IE. A structural optimization approach in process synthesis—II. *Computers & Chemical Engineering.* 1983;7(6):707-721.

21. Ježowski J, Friedler F. A simple approach for maximum heat recovery calculations. *Chem Eng Sci.* 1992;47(6):1481-1494.

22. El-Halwagi MM, Manousiouthakis V. Synthesis of mass exchange networks. *AIChE Journal.* 1989;35(8):1233-1244.

23. Roxenby S, Manousiouthakis V. Non-Isothermal Separable Mass Exchange Networks: Minimum Utility Cost through the State Space Approach. *Report, LUTKDH=(TKKA-5004).* 1994:1-133.

24. Galli MaR, Cerdá J. Synthesis of structural-constrained heat exchanger networks—I. Series networks. *Computers & chemical engineering.* 1998;22(7):819-839.

25. Gundersen T, Duvold S, Hashemi-Ahmady A. An extended vertical MILP model for heat exchanger network synthesis. *Computers & chemical engineering.* 1996;20:S97-S102.

26. Gundersen T, Grossmann IE. Improved optimization strategies for automated heat exchanger network synthesis through physical insights. *Computers & chemical engineering.* 1990;14(9):925-944.

27. Gundersen T, Traedal P, Hashemi-Ahmady A. Improved sequential strategy for the synthesis of near-optimal heat exchanger networks. *Computers & chemical engineering.* 1997;21:S59-S64.

28. Floudas CA, Ciric AR, Grossmann IE. Automatic synthesis of optimum heat exchanger network configurations. *AIChE Journal.* 1986;32(2):276-290.

29. Colberg R, Morari M. Area and capital cost targets for heat exchanger network synthesis with constrained matches and unequal heat transfer coefficients. *Computers & chemical engineering.* 1990;14(1):1-22.

30. Ježowski JM, Shethna HK, Castillo FJ. Area target for heat exchanger networks using linear programming. *Ind Eng Chem Res.* 2003;42(8):1723-1730.

31. Biegler LT, Grossmann IE, Westerberg AW. Systematic methods for chemical process design. 1997.

32. Yuan X, Pibouleau L, Domenech S. Experiments in process synthesis via mixed-integer programming. *Chemical Engineering and Processing: Process Intensification.* 1989;25(2):99-116.

33. Floudas C, Ciric A. Strategies for overcoming uncertainties in heat exchanger network synthesis. *Computers & Chemical Engineering.* 1989;13(10):1133-1152.

34. Yee TF, Grossmann IE. Simultaneous optimization models for heat integration--II. Heat exchanger network synthesis. *Computers & Chemical Engineering.* 1990;14(10):1165-1184.

35. Mistry M, Misener R. Optimising heat exchanger network synthesis using convexity properties of the logarithmic mean temperature difference. *Computers & Chemical Engineering.* 2016;94:1-17.

36. Bagajewicz MJ, Pham R, Manousiouthakis V. On the state space approach to mass/heat exchanger network design. *Chem Eng Sci.* 1998;53(14):2595-2621.

37. Huang KF, Al-mutairi EM, Karimi I. Heat exchanger network synthesis using a stagewise superstructure with non-isothermal mixing. *Chem Eng Sci.* 2012;73:30-43.

38. Isafiade A, Fraser D. Interval-based MINLP superstructure synthesis of heat exchange networks. *chemical engineering research and design.* 2008;86(3):245-257.

39. Ponce-Ortega JM, Jiménez-Gutiérrez A, Grossmann IE. Optimal synthesis of heat exchanger networks involving isothermal process streams. *Computers & Chemical Engineering.* 2008;32(8):1918-1942.

40. Ravagnani M, Caballero JA. Optimal heat exchanger network synthesis with the detailed heat transfer equipment design. *Computers & Chemical Engineering.* 2007;31(11):1432-1448.

41. Soršak A, Kravanja Z. Simultaneous MINLP synthesis of heat exchanger networks comprising different exchanger types. *Computers & Chemical Engineering.* 2002;26(4):599-615.

42. Hasan M, Jayaraman G, Karimi I, Alfadala H. Synthesis of heat exchanger networks with nonisothermal phase changes. *AIChE journal.* 2010;56(4):930-945.

43. Huang KF, Karimi IA. Simultaneous synthesis approaches for cost-effective heat exchanger networks. *Chem Eng Sci.* 2013;98:231-245.

44. Lei Y, Qi X, Zhang B, Chen Q, Hui CW. Simultaneous optimization of the complex fractionator and heat exchanger network considering the constraints of variable heat removals in delayed coking units. *Ind Eng Chem Res.* 2014;53(33):13073-13086.

45. Barbaro A, Bagajewicz MJ. New rigorous one-step MILP formulation for heat exchanger network synthesis. *Computers & chemical engineering.* 2005;29(9):1945-1976.

46. Nguyen DQ, Barbaro A, Vipanurat N, Bagajewicz MJ. All-at-once and step-wise detailed retrofit of heat exchanger networks using an MILP model. *Ind Eng Chem Res.* 2010;49(13):6080-6103.

47. Hong X, Liao Z, Jiang B, Wang J, Yang Y. New transshipment type MINLP model for heat exchanger network synthesis. *Chem Eng Sci.* 2017;173:537-559.

48. Duran MA, Grossmann IE. Simultaneous-Optimization and Heat Integration of Chemical Processes. *Aiche Journal.* 1986;32(1):123-138.

49. Baliban RC, Elia JA, Floudas CA. Simultaneous process synthesis, heat, power, and water integration of thermochemical hybrid biomass, coal, and natural gas facilities. *Computers & Chemical Engineering.* 2012;37:297-327.

50. Dowling AW, Biegler LT. A framework for efficient large scale equation-oriented flowsheet optimization. *Computers & Chemical Engineering.* 2015;72:3-20.

51. El‐Halwagi MM, Manousiouthakis V. Simultaneous synthesis of mass‐exchange and regeneration networks. *AIChE Journal.* 1990;36(8):1209-1219.

52. Kamath RS, Biegler LT, Grossmann IE. Modeling multistream heat exchangers with and without phase changes for simultaneous optimization and heat integration. *AIChE Journal.* 2012;58(1):190-204.

53. Pattison RC, Baldea M. Multistream heat exchangers: Equation‐oriented modeling and flowsheet optimization. *AIChE Journal.* 2015;61(6):1856-1866.

54. Srinivas B, El-Halwagi M. Synthesis of combined heat and reactive mass-exchange networks. *Chem Eng Sci.* 1994;49(13):2059-2074.

55. Grossmann IE, Yeomans H, Kravanja Z. A rigorous disjunctive optimization model for simultaneous flowsheet optimization and heat integration. *Computers & chemical engineering.* 1998;22:S157-S164.

56. Navarro-Amorós MA, Caballero JA, Ruiz-Femenia R, Grossmann IE. An alternative disjunctive optimization model for heat integration with variable temperatures. *Computers & Chemical Engineering.* 2013;56:12-26.

57. Zhang B, Luo X, Liu K, Chen Q, Li W. Simultaneous target of HEN and columns with variable feed temperatures for a toluene disproportionation plant. *Ind Eng Chem Res.* 2014;53(25):10429-10438.

58. Quirante N, Caballero JA, Grossmann IE. A novel disjunctive model for the simultaneous optimization and heat integration. *Computers & Chemical Engineering.* 2017;96:149-168.

59. Shah VH, Agrawal R. A matrix method for multicomponent distillation sequences. *AIChE journal.* 2010;56(7):1759-1775.

60. Caballero JA, Grossmann IE. Generalized disjunctive programming model for the optimal synthesis of thermally linked distillation columns. *Ind Eng Chem Res.* 2001;40(10):2260-2274.

61. Caballero JA, Grossmann IE. Design of distillation sequences: from conventional to fully thermally coupled distillation systems. *Computers & chemical engineering.* 2004;28(11):2307-2329.

62. Fenske M, Quiggle D, Tongberg C. Composition of straight-run Pennsylvania gasoline. *Industrial & Engineering Chemistry.* 1932;24(4):408-418.

63. Underwood A. Fractional distillation of multicomponent mixtures. *Industrial & Engineering Chemistry.* 1949;41(12):2844-2847.

64. Gilliland E. Multicomponent rectification estimation of the number of theoretical plates as a function of the reflux ratio. *Industrial & Engineering Chemistry.* 1940;32(9):1220-1223.

65. Smoker E. Analytical determination of plates in fractionating columns. *Trans Am Inst Chem Engrs.* 1938;34:165.

66. Nikolaides IP, Malone MF. Approximate design of multiple-feed/side-stream distillation systems. *Industrial & engineering chemistry research.* 1987;26(9):1839-1845.

67. Glinos KN, Malone MF. Design of sidestream distillation columns. *Industrial & Engineering Chemistry Process Design and Development.* 1985;24(3):822-828.

68. Sundaram S, Evans LB. Shortcut procedure for simulating batch distillation operations. *Industrial & engineering chemistry research.* 1993;32(3):511-518.

69. Maurer G, Prausnitz J. On the derivation and extension of the UNIQUAC equation. *Fluid Phase Equilibria.* 1978;2(2):91-99.

70. Fredenslund A, Jones RL, Prausnitz JM. Group ‐ contribution estimation of activity coefficients in nonideal liquid mixtures. *AIChE Journal.* 1975;21(6):1086-1099.

71. Viswanathan J, Grossmann IE. Optimal feed locations and number of trays for distillation columns with multiple feeds. *Ind Eng Chem Res.* 1993;32(11):2942-2949.

72. Rodebush W. A Simple Graphical Method of Calculating the Number of Plates Required for a Distilling Column. *Industrial & Engineering Chemistry.* 1922;14(11):1036-1037.

73. Ponchon M. Graphical study of distillation. *Tech Modern.* 1921;13:20.

74. Savarit R. Chimie industrie. *Special Number.* 1923;9:737-756.

75. McCabe WL, Thiele E. Graphical design of fractionating columns. *Industrial & Engineering Chemistry.* 1925;17(6):605-611.

76. Chambers JM. Extractive Distillation. Paper presented at: 3rd World Petroleum Congress1951.

77. Scheibel E. Principles of extractive distillation. *Chemical Engineering Progress.* 1948;44(12):927-931.

78. Dunn C, Millar R, Pierotti G, Shiras R, Souders M. Toluene Recovery by Extractive Distillation-Erection of a Commercial Plant on the Foundation of Laboratory Data. *Transactions of the American Institute of Chemical Engineers.* 1945;41(5):631-644.

79. Yaws CL, Li K-Y, Fang C. How to Find the Minimum Reflux for Binary Systems in Multiple-Feed Columns. *Chemical Engineer.* 1981;88(10):153-156.

80. Hengstebeck RJ. Distillation: principles and design procedures. 1961.

81. Papoulias SA, Grossmann IE. A structural optimization approach in process synthesis—I. *Computers & Chemical Engineering.* 1983;7(6):695-706.

82. Türkay M, Grossmann IE. Disjunctive programming techniques for the optimization of process systems with discontinuous investment costs-multiple size regions. *Industrial & engineering chemistry research.* 1996;35(8):2611-2623.

83. Lee S, Grossmann IE. Logic-based modeling and solution of nonlinear discrete/continuous optimization problems. *Annals of Operations Research.* 2005;139(1):267-288.

84. Kong L, Avadiappan V, Huang K, Maravelias CT. Simultaneous chemical process synthesis and heat integration with unclassified hot/cold process streams. *Computers & Chemical Engineering.* 2017;101:210-225.

85. Bausa J, Von Watzdorf R, Marquardt W. Shortcut methods for nonideal multicomponent distillation: 1. Simple columns. *Aiche J.* 1998;44(10):2181-2198.

86. Boukouvala F, Ierapetritou MG. Surrogate-based optimization of expensive flowsheet modeling for continuous pharmaceutical manufacturing. *Journal of Pharmaceutical Innovation.* 2013;8(2):131-145.

87. Kong L, Sen SM, Henao CA, Dumesic JA, Maravelias CT. A superstructure-based framework for simultaneous process synthesis, heat integration, and utility plant design. *Computers & Chemical Engineering.* 2016.

88. Gani R, Bek‐Pedersen E. Simple new algorithm for distillation column design. *AIChE Journal.* 2000;46(6):1271-1274.

89. Wu W, Henao CA, Maravelias CT. A Superstructure Representation, Generation and Modeling Framework for Chemical Process Synthesis. *AIChE Journal.* 2016.

90. Kong L, Sen SM, Henao CA, Dumesic JA, Maravelias CT. A superstructure-based framework for simultaneous process synthesis, heat integration, and utility plant design. *Computers & Chemical Engineering.* 2016;91:68-84.

91. Townsend DW, Linnhoff B. Surface area targets for heat exchanger networks. IChemE Annl Res. Mtg; 1984.

92. Linnhoff B, Ahmad S. Cost optimum heat exchanger networks—1. Minimum energy and capital using simple models for capital cost. *Computers & Chemical Engineering.* 1990;14(7):729-750.

93. Kemp IC. *Pinch analysis and process integration: a user guide on process integration for the efficient use of energy.* Butterworth-Heinemann; 2011.

94. Duran MA, Grossmann IE. Simultaneous optimization and heat integration of chemical processes. *Aiche J.* 1986;32(1):123-138.

95. Paterson WR. A replacement for the logarithmic mean. *Chem Eng Sci.* 1984;39(11):1635-1636.

96. Tawarmalani M, Sahinidis NV. A polyhedral branch-and-cut approach to global optimization. *Math Program.* 2005;103(2):225-249.

97. Kong L, Wu Y, Maravelias CT. Simultaneous Utility and Heat Exchanger Area Targeting for Integrated Process Synthesis and Heat Integration. *Industrial & Engineering Chemistry Research.* 2017;56(41):11847-11859.

98. Trespalacios F, Grossmann IE. Review of Mixed‐Integer Nonlinear and Generalized Disjunctive Programming Methods. *Chemie Ingenieur Technik.* 2014;86(7):991-1012.

99. Misener R, Floudas CA. ANTIGONE: algorithms for continuous/integer global optimization of nonlinear equations. *J Global Optim.* 2014;59(2-3):503-526.

100. Achterberg T. SCIP: solving constraint integer programs. *Math Prog Comp.* 2009;1(1):1-41.

101. Schilling G, Pantelides C. A simple continuous-time process scheduling formulation and a novel solution algorithm. *Computers & Chemical Engineering.* 1996;20:S1221-S1226.

102. Mockus L, Reklaitis G. Continuous time representation approach to batch and continuous process scheduling. 1. MINLP formulation. *Ind Eng Chem Res.* 1999;38(1):197-203.

103. Maravelias CT, Grossmann IE. New general continuous-time State− Task network formulation for short-term scheduling of multipurpose batch plants. *Ind Eng Chem Res.* 2003;42(13):3056-3074.

104. Velez S, Sundaramoorthy A, Maravelias CT. Valid inequalities based on demand propagation for chemical production scheduling MIP models. *Aiche Journal.* 2013;59(3):872-887.

105. Kong L, Wu Y, Maravelias CT. Simultaneous Utility and Heat Exchanger Area Targeting for Integrated Process Synthesis and Heat Integration. *Ind Eng Chem Res.* 2017.

106. de Farias Jr IR, Zhao M, Zhao H. A special ordered set approach for optimizing a discontinuous separable piecewise linear function. *Operations Research Letters.* 2008;36(2):234-238.

107. Croxton KL, Gendron B, Magnanti TL. A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science.* 2003;49(9):1268-1273.

108. Vielma JP, Nemhauser GL. Modeling disjunctive constraints with a logarithmic number of binary variables and constraints. *Math Program.* 2011;128(1-2):49-72.

109. Magnanti TL, Stratila D. Separable concave optimization approximately equals piecewise linear optimization. Paper presented at: International Conference on Integer Programming and Combinatorial Optimization2004.

110. Padberg M. Approximating separable nonlinear functions via mixed zero-one programs. *Operations Research Letters.* 2000;27(1):1-5.

111. Vielma JP, Ahmed S, Nemhauser G. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Operations research.* 2010;58(2):303-315.

112. Geißler B, Martin A, Morsi A, Schewe L. Using Piecewise Linear Functions for Solving MINLP s. *Mixed Integer Nonlinear Programming*: Springer; 2012:287-314.

113. Leyffer S, Sartenaer A, Wanufelle E. Branch-and-refine for mixed-integer nonconvex global optimization. *Preprint ANL/MCS-P1547-0908, Mathematics and Computer Science Division, Argonne National Laboratory.* 2008.

114. Abhishek K, Leyffer S, Linderoth J. FilMINT: An outer approximation-based solver for convex mixed-integer nonlinear programs. *INFORMS Journal on computing.* 2010;22(4):555-567.

115. Belotti P, Lee J, Liberti L, Margot F, Wächter A. Branching and bounds tighteningtechniques for non-convex MINLP. *Optimization Methods & Software.* 2009;24(4-5):597-634.

116. Bonami P, Biegler LT, Conn AR, et al. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization.* 2008;5(2):186-204.

117. Geißler B. *Towards Globally Optimal Solutions for MINLPs by Discretization Techniques with Applications in Gas Network Optimization: Ein Weg Zu Global Optimalen Lösungen Für MINLPs Durch Diskretisierungstechniken Mit Anwendungen Aus Der Gasnetzoptimierung.* Verlag Dr. Hut; 2011.

118. Bergamini ML, Grossmann I, Scenna N, Aguirre P. An improved piecewise outer-approximation algorithm for the global optimization of MINLP models involving concave and bilinear terms. *Computers & Chemical Engineering.* 2008;32(3):477-493.

119. Kallrath J. Combined strategic and operational planning–an MILP success story in chemical industry. *Or Spectrum.* 2002;24(3):315-341.

120. Frank S, Steponavice I, Rebennack S. Optimal power flow: a bibliographic survey I. *Energy Systems.* 2012;3(3):221-258.

121. Martin A, Möller M, Moritz S. Mixed integer models for the stationary case of gas network optimization. *Math Program.* 2006;105(2-3):563-582.

122. Bergamini ML, Aguirre P, Grossmann I. Logic-based outer approximation for globally optimal synthesis of process networks. *Computers & chemical engineering.* 2005;29(9):1914-1933.

123. Bertsimas D, Shioda R. Classification and regression via integer optimization. *Operations Research.* 2007;55(2):252-271.

124. Magnani A, Boyd SP. Convex piecewise-linear fitting. *Optimization and Engineering.* 2009;10(1):1-17.

125. Yang LJ, Liu SS, Tsoka S, Papageorgiou LG. Mathematical programming for piecewise linear regression analysis. *Expert Syst Appl.* 2016;44:156-167.

126. Bazaraa MS, Sherali HD, Shetty CM. *Nonlinear programming: theory and algorithms.* John Wiley & Sons; 2013.

127. Kontogiorgis S. Practical piecewise-linear approximation for monotropic optimization. *INFORMS Journal on Computing.* 2000;12(4):324-340.

128. Meyer R. Two-segment separable programming. *Management Science.* 1979;25(4):385-395.

129. Lundell A. *Transformation techniques for signomial functions in global optimization.* Åbo Akademi University; 2009.

130. Boyd S, Vandenberghe L. *Convex optimization.* Cambridge university press; 2004.

131. Williams HP. *Model building in mathematical programming.* John Wiley & Sons; 2013.

132. Rosen JB, Pardalos PM. Global Minimization of Large-Scale Constrained Concave Quadratic Problems by Separable Programming. *Math Program.* 1986;34(2):163-174.

133. Strikholm B. *Determining the number of breaks in a piecewise linear regression model.* SSE/EFI Working Paper Series in Economics and Finance;2006.

134. Rebennack S, Kallrath J. Continuous Piecewise Linear Delta-Approximations for Univariate Functions: Computing Minimal Breakpoint Systems. *J Optimiz Theory App.* 2015;167(2):617-643.

135. Rebennack S, Kallrath J. Continuous Piecewise Linear Delta-Approximations for Bivariate and Multivariate Functions. *J Optimiz Theory App.* 2015;167(1):102-117.

136. Toriello A, Vielma JP. Fitting piecewise linear continuous functions. *European Journal of Operational Research.* 2012;219(1):86-95.

137. Risbeck MJ, Maravelias CT, Rawlings JB, Turney RD. A mixed-integer linear programming model for real-time cost optimization of building heating, ventilation, and air conditioning equipment. *Energ Buildings.* 2017;142:220-235.

138. Kong L, Maravelias CT. On the Derivation of Piecewise Linear Continuous Approximating Functions. Paper presented at: 2018 AIChE Annual Meeting2018; Pittsburgh, PA.

139. Vielma JP. Mixed integer linear programming formulation techniques. *Siam Review.* 2015;57(1):3-57.

140. Jeroslow RG, Lowe JK. Modelling with integer variables. *Mathematical Programming at Oberwolfach II*: Springer; 1984:167-184.

141. Reemtsen R, Rückmann J-J. *Semi-infinite programming.* Vol 25: Springer Science & Business Media; 1998.

142. López M, Still G. Semi-infinite programming. *European Journal of Operational Research.* 2007;180(2):491-518.

143. Aggarwal A, Floudas C. Synthesis of general distillation sequences—nonsharp separations. *Computers & Chemical Engineering.* 1990;14(6):631-653.

144. Andrecovich MJ, Westerberg A. An MILP formulation for heat‐integrated distillation sequence synthesis. *AIChE journal.* 1985;31(9):1461-1474.

145. Jain S, Smith R, Kim J-K. Synthesis of heat-integrated distillation sequence systems. *Journal of the Taiwan Institute of Chemical Engineers.* 2012;43(4):525-534.

146. Floudas C, Paules IV G. A mixed-integer nonlinear programming formulation for the synthesis of heat-integrated distillation sequences. *Computers & chemical engineering.* 1988;12(6):531-546.

147. Knight JR, Doherty MF. Optimal design and synthesis of homogeneous azeotropic distillation sequences. *Ind Eng Chem Res.* 1989;28(5):564-572.

148. Shenvi AA, Shah VH, Zeller JA, Agrawal R. A synthesis method for multicomponent distillation sequences with fewer columns. *AIChE Journal.* 2012;58(8):2479-2494.

149. Caballero JA, Grossmann IE. Synthesis of complex thermally coupled distillation systems including divided wall columns. *AIChE Journal.* 2013;59(4):1139-1159.

150. Caballero JA, Grossmann IE. Structural considerations and modeling in the synthesis of heat-integrated− thermally coupled distillation sequences. *Industrial & Engineering Chemistry Research.* 2006;45(25):8454-8474.

151. Drake JE, Manousiouthakis V. IDEAS approach to process network synthesis: minimum utility cost for complex distillation networks. *Chem Eng Sci.* 2002;57(15):3095-3106.

152. Bagajewicz MJ, Manousiouthakis V. Mass/heat ‐ exchange network representation of distillation networks. *AIChE Journal.* 1992;38(11):1769-1800.

153. Yeomans H, Grossmann IE. A systematic modeling framework of superstructure optimization in process synthesis. *Computers & Chemical Engineering.* 1999;23(6):709-731.

154. Papalexandri KP, Pistikopoulos EN. Generalized modular representation framework for process synthesis. *AIChE Journal.* 1996;42(4):1010-1032.

155. Doherty MF, Malone MF. *Conceptual design of distillation systems.* McGraw-Hill Science/Engineering/Math; 2001.

156. Sinnott RK, Towler G. *Chemical engineering design: SI Edition.* Elsevier; 2009.

157. Nishida N, Stephanopoulos G, Westerberg AW. A review of process synthesis. *AIChE Journal.* 1981;27(3):321-351.

158. Chen Q, Grossmann I. Recent developments and challenges in optimization-based process synthesis. *Annual review of chemical and biomolecular engineering.* 2017;8:249-283.

159. Koehler J, Poellmann P, Blass E. A review on minimum energy calculations for ideal and nonideal distillations. *Ind Eng Chem Res.* 1995;34(4):1003-1020.

160. Kong L, Maravelias CT. From Graphical to Optimization-Based Distillation Column Design: A McCabe-Thiele-Inspired Math Program. Paper presented at: 2018 AIChE Annual Meeting2018; Pittsburgh, PA.

161. Quirante N, Grossmann IE, Caballero JA. Disjunctive model for the simultaneous optimization and heat integration with unclassified streams and area estimation. *Computers & Chemical Engineering.* 2018;108:217-231.

162. Sorel E. La Rectification de L'alcool. Gauthiers - Villais et fils; 1893; Paris.

163. Lewis W. The efficiency and design of rectifying columns for binary mixtures. *Industrial & Engineering Chemistry.* 1922;14(6):492-496.

164. Margules M. Über die Zusammensetzung der gesättigten Dämpfe von Misschungen. *Sitzungsberichte der Kaiserliche Akadamie der Wissenschaften Wien MathematischNaturwissenschaftliche Klasse II.* 1895;104:1243-1278.

165. Green DW, Southard MZ. *Perry's chemical engineers' handbook.* McGraw Hill Professional; 2018.

166. Doherty M, Malone M. Conceptual design of distillation systems McGraw-Hill. *New York.* 2001.

167. Beebe Jr A, Coulter K, Lindsay R, Baker E. Equilibria in ethanol-water system at pressures less than atmospheric. *Industrial & Engineering Chemistry.* 1942;34(12):1501-1504.

168. Murphree E. Rectifying Column Calculations. *Industrial & Engineering Chemistry.* 1925;17(7):747-750.

169. PETERS J, WA. The efficiency and capacity of fractionating columns. *Industrial & Engineering Chemistry.* 1922;14(6):476-479.

170. Kong L, Maravelias CT. From Graphical to Optimization-based Distillation Column Design: a McCabe-Thiele-inspired Math Program – I. Simple Column. *AIChE Journal.* 2019.

171. Underwood A. Fractional distillation of multi-component mixtures. *Chem Eng Prog.* 1948;44:603-614.

172. Yeomans H, Grossmann IE. Optimal design of complex distillation columns using rigorous tray-by-tray disjunctive programming models. *Industrial & engineering chemistry research.* 2000;39(11):4326-4335.

173. Douglas JM. *Conceptual design of chemical processes.* Vol 1110: McGraw-Hill New York; 1988.

174. Stichlmair J, Fair JR. Distillation: principles and practices. 1998.