Mathematical Modeling and Optimization of Heat Exchanger Design in Process Flowsheets

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree of

DOCTOR OF PHILOSOPHY

in

CHEMICAL ENGINEERING

by

SAIF R. KAZI

B.TECH., CHEMICAL ENGINEERING, INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

CARNEGIE MELLON UNIVERSITY

Pittsburgh, Pennsylvania

August, 2021

In memory of my loving aunt
and my cousin brother

Dedicated to my parents, my sister,
and my grandma

Acknowledgments

When I was applying for graduate school, one of my friends asked me whether I was sure about PhD and that 5 years can be a really long time. As I finish writing this thesis, I cannot disagree more with my friend's comment. Apart from the last year's pandemic, I have enjoyed every moment of my PhD journey and life at CMU. For this, I am sincerely grateful to a lot of admirable and thoughtful people.

I have been very fortunate to have Prof. Lorenz (Larry) Biegler as my PhD advisor. He is one of the most knowledgeable person I know and has supported me and guided me through the ups and downs of my PhD. I have learned a lot from him including how to use chopsticks and talk to people in German. *Du bist die beste berater und lehrer. Ich werde dir immer danke für meine Doctorikter.* I would also like to thank the committee members Prof. Ignacio Grossmann, Prof. Nikolaos Sahinidis and Prof. Franziska Weber for reading the thesis and giving constructive comments about it. I have also enjoyed attending your courses and learned important aspects which were used in the thesis. Special thanks to Prof. Michael Short for being such a nice guy to work with and collaborating on multiple projects. Our discussions on heat exchanger network and premier league soccer were the best moments of our Zoom calls. I would like to thank Dr. Youngdae Kim and Dr. Rui Huang for guiding and helping me during the internships at Argonne National Lab and United Technologies respectively. Both of you trusted me and taught me about research and its impact in different applications. I am also grateful to Dr. Clas Jacobson and Dr. Eugene Cliff from UTC who helped me with the heat exchanger modeling inside chiller refrigeration cycle. Thanks to Dr. Kai Liu and Prof. Amir Patel who introduced me to their research topics and included me to contribute to their study. Especially Kai, with whom I had countless debates and discussions over research and general knowledge. I am also thankful to my funding sources HighEFF in Norway with Prof. Johannes Jäschke and Prof. Truls Gundersen for their unconditional support throughout my PhD.

I believe what makes PSE@CMU so special is the amazing PhD students and post-docs who are part of it. I have had the good fortune to meet and talk to some of these brilliant PhD students over the lunch and coffee discussions. Special mention to Dr. David Thierry who has not only helped me with learning Pyomo and IPOPT but has also inspired me to make cheesecakes and pumpkin pie. David has been like a *big brother* for me and the other group members and has immensely helped us with our research. Dr. Joyce Yu and I had

some wonderful discussions over St. Louis, control, self-driving cars and food culture for which I am very thankful to her. I am also glad to have my other group members Yunhan, Robby, Can, Nori (Dr. Yoshio), Vibhav, Kwan-Han and Tom with whom I spend some nice moments. I am also grateful to my group alumni Dr. Wei Wan, Dr. John Eason, Dr. Devin Griffith and Dr. Yajun Wang for their insightful guidance about PhD and opportunities. Another special mention is for Dr. Anirudh Subramanyam (soon to be a Prof.) whose "*fundae*" about PhD, research, academia and life have helped me enormously. Thanks for introducing me to one of the best places in Pittsburgh (*Le Mardi Gras*). I am thankful to Dr. Chris Hanselman, Dr. Akang Wang, Dr. Carlos Nohra, Eyan and Aliakbar for their company and interesting discussions on non-academic matters. I would like to thank my PhD batch-mates Nick, Bhavya, Natalie (and Connor), Brad, Olga, Michael, Scott, Spiro, Dr. Yijia, Dr. Andrew, Dr. Kevin and Dr. Bowen for their friendships and company during the department happy hour. Special thanks to Dr. Can Li and Dr. David Bernal (both soon to be Profs.) who have been fantastic friends over the past 5 years and have taught me so much about their research and other things. I am glad that I got to meet some wonderful people Dr. Christina Schenk, Dr. Cornelius Masuku, Flemming Holtorf and Dr. Mandar Thombre as they visited CMU and our group. I was lucky to have many Indian friends - Devdeep, Pratik, Ayush, Ankit, Kunal, Parul and Dr. Yash Puranik who made sure I didn't miss home much. I am grateful to Chaitanya who has been an amazing friend and housemate for the entire PhD. Thanks to Neal, Manasvita and Deepak Dilipkumar for their banter and friendly chats. Dr. Vineet Jeco has been a close friend and supporter since my undergraduate and I sincerely appreciate and am grateful for his friendship. I am also thankful to my childhood friend Shubham Anand for his constant support and words of confidence. I would also like to thank him for his help with some of the MHEX figures.

None of this would have been possible without my family's support. I lost two of my strongest supporters my loving aunt and her son in tragic accidents during the initial phase of my PhD and I miss them a lot. Still the rest of my family has always supported me during this adventurous journey of my life. My sister Neha and my grandma have always believed in me and my abilities. My parents (*Baba* and *Maa*) have been an immense source of strength and support throughout my life. I know that I am not always a good son but I am deeply grateful for all that you have done for me and this thesis is dedicated to you. Finally, I would like to end with a quote "*Nothing lasts forever, for all good things it's true*"

Saif R. Kazi
Pittsburgh, PA
June 2021

Abstract

In the last two decades, optimization methods and algorithms have improved immensely in both criterion: number of problems solved and their solution times. Unfortunately, this has not translated in the more accurate modeling of chemical engineering process models. Current flowsheet models use simple mass and energy balance to represent process equipments and do not incorporate effect of its design in the model. In this thesis, the heat exchanger design is studied from the viewpoint of embedding it in the flowsheet models. A novel design model based on differential heat equation and the geometric structure of the exchanger is proposed. The model is capable of capturing the effects of phase change and variable thermophysical properties on the exchanger design. This is demonstrated by solving multiple examples of shell and tube heat exchangers and flooded phase change heat exchangers.

In the subsequent chapter, the exchanger design model is integrated into the heat exchanger network synthesis model using two separate methods: two-step hybrid strategy and trust region filter approach. Examples from previous studies are used to show the significance effect of including detailed exchanger design in the synthesis of heat exchanger networks. Afterwards, the design model for multi-stream heat exchanger using a discretized system of equations and complementarity constraints is developed. The design equations relating the heat exchanger area to the design variables and degrees of freedom are derived. This design model is solved simultaneously with a natural gas liquefaction model using a step-by-step initialization procedure. The optimization results for both the exchanger design and flowsheet variables are presented along with plots of temperature variation and liquid fraction for each stream inside the exchanger.

Contents

Acknowledgments **i**

Abstract **iii**

List of Tables

List of Figures

1 Introduction **1**
1.1 Motivation 1
1.2 Research Statement and Thesis Outline 3

2 Nonlinear Optimization **5**
2.1 Introduction 5
2.2 Optimality Conditions 6
2.2.1 First Order KKT Conditions 7
2.2.2 Constraint Qualifications (CQs) 7
2.2.3 Second Order Conditions 8
2.3 NLP algorithms 9
2.3.1 Interior point method 9
2.3.2 Reduced Gradient method 12
2.4 NLP Parameter Sensitivity 14
2.5 Dynamic Optimization 15
2.5.1 Discretization 16
2.5.2 Optimality Conditions 18
2.5.3 Solution Strategy 19
2.6 Equilibrium and Complementarity Constraints 21
2.6.1 MPEC and MPCC 21
2.6.2 Properties and Formulations 22
2.6.3 Hybrid Dynamic Systems 28
2.7 Summary 31

3 Heat Exchanger Design **32**
3.1 Introduction . . . 32
3.2 LMTD based methods . . . 34
3.2.1 Kern method . . . 35
3.2.2 Bell-Delaware method . . . 35
3.3 DAE method . . . 36
3.3.1 PDE to ODE . . . 37
3.3.2 Discretization . . . 38
3.3.3 Algebraic Equations . . . 40
3.3.4 Multiple Shell . . . 40
3.3.5 Error Analysis . . . 41
3.4 Solution Strategy . . . 45
3.4.1 Structural Design . . . 45
3.4.2 Area Design . . . 46
3.5 Phase Change . . . 48
3.5.1 Flooded Evaporator . . . 49
3.5.2 Flooded Condenser . . . 52
3.6 Examples . . . 53
3.6.1 Single shell Exchanger . . . 54
3.6.2 Multiple shell Exchanger . . . 63
3.6.3 Phase Change Exchanger . . . 66
3.7 Conclusions . . . 69

4 Heat Exchanger Network Synthesis **71**
4.1 Introduction . . . 71
4.1.1 HENS with Detailed Exchanger Design . . . 73
4.2 HENS model . . . 75
4.2.1 SWS MINLP model . . . 75
4.2.2 Network NLP model . . . 78
4.3 Two-step Hybrid Strategy . . . 79
4.3.1 Exchanger Design model . . . 79
4.3.2 Correction Factors . . . 79
4.4 Trust Region Based Strategy . . . 81
4.4.1 Trust Region Filter . . . 83
4.4.2 Trust Region Radius Update . . . 83
4.4.3 Integer Cut . . . 84

4.5 Case Study . . . 86
4.5.1 Example-1 . . . 86
4.5.2 Example-2 . . . 91
4.5.3 Example-3 . . . 95
4.6 Conclusions . . . 100

5 Multi-Stream Heat Exchanger Design **101**
5.1 Introduction . . . 101
5.2 Multi-Stream Heat Exchanger DAE model . . . 103
5.2.1 Heat Equation . . . 105
5.2.2 Thermodynamics . . . 108
5.2.3 Phase Change . . . 109
5.2.4 Design Equation . . . 110
5.3 Natural Gas Liquefaction Flowsheet . . . 112
5.3.1 Mass & Enthalpy Balance . . . 114
5.3.2 Process Constraints . . . 116
5.3.3 Objective Function . . . 116
5.4 Initialization and Numerical Results . . . 117
5.5 Conclusions . . . 122

6 Conclusions **123**
6.1 Summary and Contributions . . . 123
6.2 Recommendations for Future Work . . . 126

Bibliography **129**

Appendix **139**
.1 Heat Exchanger Design Equations . . . 139
.2 HENS Models . . . 142
.2.1 SWS MINLP model . . . 143
.2.2 Network NLP model . . . 144
.3 Derivation of Correction Factors . . . 147
.4 Trust Region Constraint and Parameters . . . 150
.5 Detailed Exchanger Design Results for Example 3 . . . 151
.6 Cubic Equation of State for SWHX model . . . 157

List of Tables

3.1 The main assumptions in LMTD based methods 36
3.2 Comparison between numerical and analytical solutions 43
3.3 Results of DAE model with different discretizations 44
3.4 Advantages and assumptions associated with the proposed strategy 47
3.5 TEMA Standard Values 54
3.6 Stream Flowrate Data 55
3.7 Exchanger Design Data 55
3.8 Comparison of LMTD and DAE-based Model Results 56
3.9 Comparison of DAE-based model with fixed and variable L_T 58
3.10 Comparison of DAE solution for 1-1 design and multiple passes 59
3.11 Slope Values for the thermodynamic properties 61
3.12 Detailed exchanger designs for Example with varying thermodynamics . . 62
3.13 Multiple shell exchanger Example Data 63
3.14 Detailed exchanger designs for Multi-shell Exchanger Example 64
3.15 Operating Conditions for Refrigeration Cycle 67
3.16 Exchanger Characteristics 67

4.1 Stream data applied for all streams in the examples 86
4.2 Stream data for Example 1 87
4.3 Detailed heat exchanger results with Hybrid Strategy for Example 1 88
4.4 Summary of solutions obtained for Example 1 in comparison with other studies 89
4.5 Detailed heat exchanger results with TRF based Strategy for Example 1 . . . 90
4.6 Stream data for Example 2 91
4.7 Detailed heat exchanger results with Hybrid Strategy for Example 2 with bypass 93
4.8 Summary of solutions obtained for Example 2 in comparison with other studies 94
4.9 Detailed heat exchanger results with TRF based Strategy for Example 2 without bypass 95

4.10 Stream data for Example 3 96
4.11 Summary of solutions obtained for Example 3 in comparison with other studies 98

5.1 Pressure drop inside the MHEX 115
5.2 Initial values and bounds for decision variables 117
5.3 Natural Gas Liquefaction Optimization Results 119
5.4 Natural Gas Liquefaction Stream Table 120
5.5 MHEX Design Values 120

1 Results of E1 - E5 for Example 3 without bypasses using Hybrid Strategy . . 151
2 Results of E6 - E10 for Example 3 without bypasses using Hybrid Strategy . 152
3 Results of E11 for Example 3 without bypasses using Hybrid Strategy 153
4 Results of E1 - E5 for Example 3 without bypasses using TRF Strategy . . . 154
5 Results of E6 - E10 for Example 3 without bypasses using TRF Strategy . . . 155
6 Results of E11 - E13 for Example 3 without bypasses using TRF Strategy . . 156
7 Parameters for Cubic Equation of State-I 157
8 Parameters for Cubic Equation of State-II 158

List of Figures

2.1 Flash Tank 29

3.1 Discretization of the shell and tube heat exchanger 38
3.2 Discrete structure consisting of finite elements 39
3.3 Structures for single and multiple shell heat exchanger 40
3.4 Reformulated discrete structure 41
3.5 Temperature Profiles of Numerical and Analytical Solutions 43
3.6 Fine Discretization of 1-2 heat exchanger 44
3.7 Flooded Evaporator 49
3.8 Flooded Exchanger Discretization 50
3.9 Temperature plots inside the heat exchanger designs 60
3.10 Temperature plot inside the optimal heat exchanger design 65
3.11 Single-stage Refrigeration Cycle 66
3.12 Evaporator Results 68
3.13 Condenser Results 68

4.1 Stagewise Superstructure representation of HENS model 75
4.2 Performance of the LMTD Smooth approximation 76
4.3 Flowchart for two-step hybrid strategy 80
4.4 Trust Region Algorithm Flowsheet 84
4.5 The overall trust region filter based strategy for simultaneous HENS design 85
4.6 Network results with Hybrid Strategy for Example 1 87
4.7 Network results with TRF Strategy for Example 1 89
4.8 Network results with Hybrid Strategy for Example 2 92
4.9 The solutions of the network topology and the detailed network designs over the course of the algorithm. 92
4.10 Network results with TRF based Strategy for Example 2 94
4.11 Exchanger Network for Example 3 without bypasses using Hybrid Strategy 97
4.12 Exchanger Network for Example 3 without bypasses using TRF based Strategy 99

5.1 Inside view of spiral wound heat exchanger 104

5.2 Cross-section view of SWHX 104
5.3 Discretization of SWHX into discrete finite elements 105
5.4 Discrete Element inside SWHX 106
5.5 Natural Gas Liquefaction Flowsheet 113
5.6 Reduced Natural Gas Liquefaction Flowsheet 113
5.7 Temperature and Liquid Mole Fraction inside Warm Bundle 121
5.8 Temperature and Liquid Mole Fraction inside Cold Bundle 121

1 Demonstration of evolution of correction factors over the course of the algorithm for Example 1 149

Chapter 1

Introduction

1.1 Motivation

Computational methods have vastly improved the energy efficiency and safety of chemical processes in industry. Advances in modeling tools and optimization methods have allowed for higher process intensification and complex design. Nonlinear programming (NLP) and Mixed integer nonlinear programming (MINLP) have been vastly used in modeling chemical engineering processes [1]. Many optimization algorithms and techniques have been developed to solve these nonlinear process models [2].

Although there has been a lot of success in solving these MINLP and NLP models, the level of detail and modeling in these models has not increased much through the years. Specifically, heat exchanger models have not been developed or updated in these optimization models. Heat exchangers are an essential part of any chemical engineering process and their design has significant effect on the process performance. Still, heat exchangers are modeled using simple and less accurate equations in process models. These equations do not consider the internal structural design of the exchanger and the complex fluid flow inside the exchanger. Moreover, phenomena like phase change and temperature dependence of physical properties are completely ignored while solving process flowsheet models.

Heat exchanger designs are going to be be more important in future process flowsheets as energy demands increase and effects of climate change require more process intensification. Heat pumps in automobiles, HVAC systems in buildings and natural gas liquefaction units in refineries are some of the examples of energy systems where heat exchangers with

sophisticated and detailed design are required. Reports on future trends in energy consumption by U.S. Energy Information Agency (EIA) and Department of Energy (DOE) show an increase in both energy use and efficiency requirement in centralized HVACs and heat pumps. The Annual Energy Outlook (2021) by EIA [3] also predict the demand and consumption of natural gas for industrial use and exports to increase till 2050.

Moreover, the rise in renewable energy production and use of zero carbon sources of energy like solar, wind, hydro and nuclear requires different ways of energy transfer at different temperatures. Renewable energy has been primarily used as source of electricity generation for residential and commercial purposes. For their application to extend as heating or cooling utility, exchangers with flexible design and operability over long range of temperature and pressure are needed.

There is a need to reexamine the modeling of heat exchanger design and its methods with a perspective on incorporating the effects of exchanger design in optimization of energy systems. New design models for heat exchangers are required which are more accurate than the current models and can account for phase change and physical property variations inside the exchanger. These models should be based on first principles physics (conservation laws) and incorporate the effects of the fluid flow pattern inside the exchanger. The design models should also be compatible and solvable with the optimization models of the advanced energy systems.

The second requirement is the development of optimization methods such that these models are solved simultaneously inside the process flowsheet models. The optimization algorithm should be robust and able to capture the detailed heat exchanger design including the geometric and performance design variables. The algorithm should also take advantage of modern and advanced solvers for NLP and MINLP to efficiently solve the simultaneous flowsheet models with detailed heat exchanger design.

In the next section, the main aim and the research statement of the thesis will be defined with a detailed outline including an overview of each chapter.

1.2 Research Statement and Thesis Outline

This thesis aims to address the gap between process modeling and equipment design by focusing on building new design models for exchanger which are not only more accurate but also suitable for derivative based optimization solvers. The thesis also focuses on optimization methods like two-step hybrid strategy and trust region algorithm in embedding the exchanger design models inside process flowsheet models.

The rest of the thesis and its chapters are organized as follows:

Chapter 2 presents necessary introductory details on nonlinear and dynamic optimization. Optimality and necessary (and sufficient) conditions are defined and discussed in detail. NLP solution algorithms based on KKT conditions are discussed and their respective solver codes used in the thesis are mentioned. Parametric sensitivity is defined and derived using implicit function theorem and linearized KKT matrix. Basic aspects of dynamic optimization are discussed with optimality conditions and discretization methods. Mathematical programming with equilibrium constraints (MPECs) and complementarity constraints (MPCCs) are introduced with its necessary and sufficient optimality conditions and NLP based methods to solve them. Finally, a hybrid dynamic example of an engineering system is presented to better understand the above topics.

Chapter 3 describes the details of heat exchanger design problem by using shell and tube heat exchanger (STHE) as the default exchanger type. Current methods based on log mean temperature difference (LMTD) are discussed along with the assumptions used in its derivation. A first principles physics based design model is derived from the PDE heat equations and the geometric design of STHE. A complementarity based formulation is developed to model the phase change phenomenon inside the exchanger. Multiple examples with changing thermophysical properties, multiple shell and phase change are solved to demonstrate the advantages of the newly developed model.

In Chapter 4, the classical heat exchanger network synthesis (HENS) problem is introduced with different types of models for various objective. The simultaneous HENS

MINLP model with detailed exchanger design is developed with a combination of network and exchanger design models. The simultaneous HENS model is solved with a two-step hybrid strategy and a newly developed trust region filter based strategy with DAE models is used for detailed exchanger design. Literature based examples are solved using both approaches and their results are compared to other studies showcasing the importance of including exchanger design in HENS models.

Chapter 5 introduces multi-stream heat exchanger (MHEX) and its most common type spiral wound heat exchanger (SWHX). Applications of MHEX in air separation units and natural gas liquefaction process is discussed. A DAE model based on heat equations and flow dynamics inside SWHX is developed. The phase change equations for multi-component mixture are derived by non-smooth flash equations. The DAE model is embedded into a mixed refrigerant (MR) based natural gas liquefaction flowsheet model and then solved as a large scale NLP using a step-wise initialization strategy.

Chapter 6 provides the summary and conclusions of the thesis chapters along with multiple possible directions for future work.

Chapter 2

Nonlinear Optimization

This chapter provides a brief overview of nonlinear programming (NLP) and dynamic optimization problems. Necessary and sufficient optimality conditions for NLP are presented along with discussion on constraint qualifications and multipliers. Two NLP solution methods are described along with their implementations in popular software codes. A brief introduction to dynamic optimization with regards to problem types, optimality conditions and solution methods is provided. Different approaches to solve dynamic optimization problem using NLP solvers are discussed.

Finally, the concept of equilibrium and complementarity constraints is introduced with regards to mathematical programming. Some theoretical properties and popular NLP reformulations are discussed along with applications in process engineering.

2.1 Introduction

Nonlinear optimization is a class of continuous optimization problems where the objective is a nonlinear function. Nonlinear optimization with linear or nonlinear constraint functions is known as nonlinear programming (NLP). A general NLP problem can be written as:

$$\min_{x} \quad f(x) \tag{2.1a}$$

$$s.t. \quad g(x) \leq 0, \tag{2.1b}$$

$$h(x) = 0 \tag{2.1c}$$

where $x \in \mathbb{R}^n$ represents the decision variables, $f(x) : \mathbb{R}^n \to \mathbb{R}$ is the objective function,

$g(x) : \mathbb{R}^n \to \mathbb{R}^p$ and $h(x) : \mathbb{R}^n \to \mathbb{R}^m$ are the inequalities and equality constraints respectively. The feasible region is defined by the set $\mathcal{F} = \{x \in \mathbb{R}^n : h(x) = 0, g(x) \leq 0\}$.

Some important aspects of nonlinear programming are defined below:

Definition 2.1a A point x^* is called *local minimum* of (2.1) if $\exists\ \epsilon > 0$ s.t.

$$f(x) \leq f(x^*), \quad \forall x \in \mathcal{F} \cap ||x - x^*|| \leq \epsilon \tag{2.2a}$$

Definition 2.1b A point x^* is called *global minimum* of (2.1) if

$$f(x) \leq f(x^*), \quad \forall x \in \mathcal{F} \tag{2.2b}$$

Definition 2.2a A set D is a convex set if and only if

$$\alpha x_1 + (1 - \alpha)x_2 \in D, \ \ \forall x_1, x_2 \in D, \forall \alpha \in (0, 1) \tag{2.2c}$$

Definition 2.2b A function $f(x)$ is convex on some convex set D if and only if

$$\alpha f(x_1) + (1 - \alpha)f(x_2) \geq f(\alpha x_1 + (1 - \alpha)x_2), \ \ \forall x_1, x_2 \in D, \forall \alpha \in (0, 1) \tag{2.2d}$$

Theorem 2.1 *If in the NLP given by (2.1), $g(x)$ is convex and $h(x)$ is linear, then the feasible region set $\mathcal{F}$ is convex. Moreover, if $f(x)$ is convex in $\mathcal{F}$ then every local minimum of (2.1) is also a global minimum of (2.1)*

The proof is provided in the book by Biegler [4]. The theorem actually shows the importance of having a convex objective function and inequalities with linear equations. Unfortunately, most engineering models have nonlinear equations and nonconvex objective function making it difficult to obtain the global minimum. Nevertheless, finding a local minimum for NLPs with large number of variables and constraints is good enough for many practical applications.

2.2 Optimality Conditions

Similar to unconstrained optimization, local minima for general NLPs can be obtained by solving a set of nonlinear equations also known as optimality conditions. These conditions are only applicable if the functions f, g and h have continuous first and second derivatives.

2.2.1 First Order KKT Conditions

First order optimality conditions for NLPs also known as Karush-Kuhn-Tucker (KKT) conditions use first derivatives and auxiliary variables called multipliers to calculate stationary points. The KKT conditions are necessary conditions under some assumptions for local minimum x^* of NLP (2.1) and are defined using a Lagrange function:

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \lambda^T h(x) + \mu^T g(x) \tag{2.3}$$

The KKT conditions for the NLP (2.1) at point x^* are:

$$\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \nabla f(x^*) + \nabla h(x^*)\lambda^* + \nabla g(x^*)\mu^* = 0 \tag{2.4a}$$

$$g(x^*) \leq 0 \tag{2.4b}$$

$$h(x^*) = 0 \tag{2.4c}$$

$$\mu^* \geq 0 \tag{2.4d}$$

$$g(x^*)^T \mu^* = 0 \tag{2.4e}$$

(λ^*, μ^*) are called Lagrange multipliers and the last KKT condition is also called a complementarity condition.

2.2.2 Constraint Qualifications (CQs)

The constraint qualifications are conditions needed to be satisfied by the NLP constraints for a local minimum x^* of NLP to be a solution of the KKT conditions. There are multiple types of CQs defined for NLPs, but for the purpose of this work only two of them are discussed here: Linear independence constraint qualification (LICQ) and Mangasarian Fromovitz constraint qualification (MFCQ). We first define:

Definition 2.3 The active-set $\mathcal{A}(x^*)$ at a point x^* is defined as the set of active inequalities

$$\mathcal{A}(x^*) = \{i \in \{1, 2, ..p\} | g_i(x^*) = 0\} \tag{2.5}$$

Definition 2.4 *Linear Independence Constraint Qualification (LICQ)*: A point x^* is said to satisfy LICQ if the gradients of equalities and active inequalities are linearly independent.

$$\nabla h(x^*) \ \textit{and} \ \nabla g_i(x^*), i \in \mathcal{A}(x^*) \ \textit{are linearly independent} \tag{2.6}$$

Definition 2.5 *Mangasarian Fromovitz Constraint Qualification (MFCQ)*: A point x^* is said to satisfy MFCQ if the gradients of equalities are linearly independent and there exists a search direction d in the interior of the feasible region

$$\nabla h(x^*)^T d = 0 \ \textit{and} \ \nabla g_i(x^*)^T d < 0, \ \ i \in \mathcal{A}(x^*) \tag{2.7}$$

It can be easily shown that LICQ is a stronger condition than MFCQ. If a solution x^* of (2.1) satisfies MFCQ, then the corresponding Lagrange multipliers (λ^*, μ^*) are bounded. Moreover, if the solution x^* also satisfies LICQ then the multipliers are both bounded and unique.

2.2.3 Second Order Conditions

Second order optimality conditions depend on the curvature of the Lagrangian function and thus require second derivatives of the objective and constraint functions. First we define a conic set:

$$\mathcal{C}(x,\mu) = \{d \ | \ \nabla h(x)^T d = 0, \nabla g_i(x)^T d = 0 \ \ i \in \{i|\mu_i > 0\}, \nabla g_j(x)^T d \leq 0 \ \ j \in \{j|\mu_j = 0\}\} \tag{2.8}$$

Theorem 2.2a *(Second order necessary conditions) If x^* is a local minimum of (2.1), (x^*, λ^*, μ^*) satisfy KKT conditions and LICQ holds, then:*

$$d^T \nabla^2_{xx} \mathcal{L}(x^*, \lambda^*, \mu^*) d \geq 0 \quad \forall d \in \mathcal{C} \tag{2.9}$$

Theorem 2.2b *(Second order sufficient conditions) If (x^*, λ^*, μ^*) satisfy KKT conditions, LICQ and:*

$$d^T \nabla^2_{xx} \mathcal{L}(x^*, \lambda^*, \mu^*) d > 0 \quad \forall d \neq 0 \in \mathcal{C} \tag{2.10}$$

then x^* is a strict local solution of (2.1). Proof is provided in [4, p. 80-81]

2.3 NLP algorithms

Several gradient based algorithms have been developed to solve NLPs over the years. In this section, two popular approaches: 1) interior point method and 2) gradient projection method are presented and discussed.

2.3.1 Interior point method

Interior point (IP) methods are a class of NLP methods which handle the inequality constraints by adding them to the objective as a barrier function (φ). This is presented by reformulating the general NLP (2.1) into a bound constrained NLP using slack variables.

$$\min_{x \in \mathbb{R}^n} \quad f(x) \tag{2.11a}$$

$$s.t. \quad c(x) = 0, \tag{2.11b}$$

$$x \geq 0 \tag{2.11c}$$

A sequence of barrier problems are solved using a barrier parameter μ_l as follows:

$$\min_{x \in \mathbb{R}^n} \quad \varphi_{\mu_l}(x) = f(x) - \mu_l \sum_i log(x_i) \tag{2.12a}$$

$$s.t. \quad c(x) = 0 \tag{2.12b}$$

The solution to the barrier problem $x^*(\mu_l)$ remains strictly in interior of the feasible region and converges to the local minimum x^* of (2.11) as $\lim_{l \to \infty} \mu_l \to 0$ under sufficient optimality conditions listed in [4, p. 152].

The KKT conditions for (2.12) can be written as:

$$\nabla f(x) + \nabla c(x)\lambda - z \tag{2.13a}$$

$$c(x) = 0 \tag{2.13b}$$

$$Xz - \mu e = 0 \tag{2.13c}$$

z represents the multipliers for bound constraints. $X := diag(x)$ is a diagonal matrix with entries from x and $e := [1, 1, ...]^T$ is a vector with all entries equal to 1.

2.3.1.1 KKT Matrix

The KKT conditions (Eq.2.13) are a set of nonlinear equations which can be solved using Newton's method. The Jacobian matrix of the KKT system is also called the KKT matrix and the Newton step at iterate *k* is shown below:

$$\begin{bmatrix} W^k & \nabla c(x^k) & -I \\ \nabla c(x^k)^T & 0 & 0 \\ Z^k & 0 & X^k \end{bmatrix} \begin{bmatrix} d_x^k \\ d_\lambda^k \\ d_z^k \end{bmatrix} = - \begin{bmatrix} \nabla f(x^k) + \nabla c(x^k)\lambda^k - z^k \\ c(x^k) \\ X^k z^k - \mu e \end{bmatrix} \tag{2.14}$$

where $W^k := \nabla^2_{xx}\mathcal{L}(x^k, \lambda^k)$, $(d_x^k, d_\lambda^k, d_z^k)$ are search directions and $Z^k := diag(z^k), X^k := diag(x^k)$ are diagonal matrices. The KKT matrix can be further simplified by pivoting on X^k and eliminating the last row and column. This results in a smaller and symmetric KKT matrix.

$$\begin{bmatrix} W^k + \Sigma^k & \nabla c(x^k) \\ \nabla c(x^k)^T & 0 \end{bmatrix} \begin{bmatrix} d_x^k \\ d_\lambda^k \end{bmatrix} = - \begin{bmatrix} \nabla \varphi(x^k) + \nabla c(x^k)\lambda^k \\ c(x^k) \end{bmatrix} \tag{2.15}$$

where $\Sigma^k := (X^k)^{-1}Z^k$ and $\nabla\varphi(x^k) := \nabla f(x^k) - \mu(X^k)^{-1}e$. The search direction for bound multipliers d_z^k can be calculated as:

$$d_z^k = \mu(X^k)^{-1}e - z^k - \Sigma^k d_x^k \tag{2.16}$$

Symmetric linear solvers require the linear system to be non-singular and have correct inertia (number of positive,negative and zero eigenvalues) to solve them. To ensure this, the KKT matrix can be further modified using regularization terms.

$$\begin{bmatrix} W^k + \Sigma^k + \delta_w I & \nabla c(x^k) \\ \nabla c(x^k)^T & -\delta_c I \end{bmatrix} \begin{bmatrix} d_x^k \\ d_\lambda^k \end{bmatrix} = - \begin{bmatrix} \nabla \varphi(x^k) + \nabla c(x^k)\lambda^k \\ c(x^k) \end{bmatrix} \tag{2.17}$$

The parameters are updated using a suitable algorithm presented in [4, p. 98]

Newton's method is only locally convergent and thus it needs to be augmented with globalization strategies like line search method to converge globally from any initial point.

2.3.1.2 Line Search Filter

Line search methods solve the local convergence issue of Newton's method by scaling the Newton step by a step size parameter $\alpha_k \in (0, 1]$. This step size is either determined by using merit functions or by comparing the quality of the current iterate with a set of previous iterate solutions also known as filter set.

A filter set $\mathcal{F}$ is constructed using previous values of constraint violation $\theta(x) := ||c(x)||$ and barrier function φ.

$$\mathcal{F}_k \subseteq \{(\theta_k, \varphi_{\mu,k}) \in \mathbb{R}^2 | \theta^k \geq \theta^{min}\} \tag{2.18}$$

The value of step size α_k is calculated by sequentially checking the following conditions and changing the step size accordingly:

Sufficient decrease:

$$\theta(x^k + \alpha_k d_x^k) \leq (1 - \gamma_\theta)\theta_i \text{ or } \varphi_{\mu,k}(x^k + \alpha_k d_x^k) \leq \varphi_{\mu,i} - \gamma_\varphi \theta_i \quad \forall(\theta_i, \varphi_{\mu,i}) \in \mathcal{F}_k \tag{2.19}$$

Switching condition:

$$\nabla\varphi_{\mu,k}(x^k)^T d_x^k \leq 0 \quad \text{and} \quad \alpha_k[-\nabla\varphi_{\mu,k}(x^k)^T d_x^k]^{s_\varphi} \geq \delta[\theta(x^k)]^{s_\theta} \tag{2.20}$$

Armijo condition:

$$\varphi_\mu(x^k + \alpha_k d_x^k) \leq \varphi_\mu(x^k) + \eta_\varphi \alpha_k \nabla\varphi_\mu(x^k)^T d_x^k \tag{2.21}$$

After determining the best step size α^*, the variables and multipliers are updated as:

$$x^{k+1} = x^k + \alpha_x^k d_x^k, \quad \lambda^{k+1} = \lambda^k + \alpha_\lambda^k d_\lambda^k, \quad z^{k+1} = z^k + \alpha_z^k d_z^k \tag{2.22}$$

The filter is updated if the point satisfies the sufficient decrease condition but does not satisfy the other conditions:

$$\mathcal{F}_{k+1} = \mathcal{F}_k \cup \{\theta_k, \varphi_{\mu,k}\} \tag{2.23}$$

More details on the line search filter used in interior point methods can be found in [5]. There are other approaches like trust region strategy to globalize convergence of Newton's method which are discussed in later chapters of the thesis.

2.3.2 Reduced Gradient method

These methods are based on active set approach and directly handle the inequality constraints in the NLP. The method uses reduced gradient and reduced of constraint and objective function which are based on projection of the full gradient and full Hessian.

Consider the NLP with box constraints:

$$\min_{x \in \mathbb{R}^n} \quad f(x) \tag{2.24a}$$

$$s.t. \quad c(x) = 0, \tag{2.24b}$$

$$x_L \leq x \leq x_U \tag{2.24c}$$

Partition the variables into: basic, nonbasic and superbasic variables as $x = [x_B^T, x_N^T, x_S^T]^T$ based on their values. The nonbasic variables x_N are set at their bounds, whereas the basic variables x_B are calculated using the equality constraint $c(x) = 0$. Therefore only the superbasic variables x_S are the free variables or degrees of freedom in the optimization.

The KKT conditions for the NLP can be written as:

$$\nabla_{x_S} f(x) + \nabla_{x_S} c(x)\lambda = 0, \tag{2.25a}$$

$$\nabla_{x_N} f(x) + \nabla_{x_N} c(x)\lambda - z_L + z_U = 0, \tag{2.25b}$$

$$\nabla_{x_B} f(x) + \nabla_{x_B} c(x)\lambda = 0, \tag{2.25c}$$

$$c(x) = 0, \tag{2.25d}$$

$$x_{N,(i)} = x_{N,L,(i)} \text{ or } x_{N,(i)} = x_{N,U,(i)} \tag{2.25e}$$

The Eq.(2.25c) - (2.25d) are embedded inside Eq.(2.25a) - (2.25b) by solving for basic variables as an implicit function of superbasic variables. This can be done using the sensitivity information from the constraint function and substituting (2.25c) in (2.25a) as follows:

$$dc(x) = \nabla_{x_B} c(x)^T dx_B + \nabla_{x_S} c(x)^T dx_S = 0 \Rightarrow \left(\frac{dx_B}{dx_S}\right)^T = -(\nabla_{x_B} c(x))^{-1} \nabla_{x_S} c(x), \tag{2.26a}$$

$$\nabla_{x_S} f(x) - \nabla_{x_S} c(x)(\nabla_{x_B} c(x))^{-1} \nabla_{x_B} f(x) = 0 \Rightarrow \nabla_{x_S} f(x) + \left(\frac{dx_B}{dx_S}\right) \nabla_{x_B} f(x) = 0 \tag{2.26b}$$

Newton's method with line search is used to solve for x_S using Eq.(2.26b). To perform this effectively, reduced gradient and Hessian are defined using a matrix Z_P which lies in the null space of $\nabla c(x)^T$:

$$Z_P = \begin{bmatrix} -(\nabla_{x_B} c(x)^T)^{-1} \nabla_{x_S} c(x)^T \\ I \\ 0 \end{bmatrix} \tag{2.27}$$

The reduced gradient and Hessian are defined as $Z_P^T \nabla f$ and $Z_P^T \nabla^2_{xx} \mathcal{L}(x, \lambda, z) Z_P$ respectively. The size of reduced Hessian is mostly very small ($|x_S|$) as compared to full Hessian and can be approximated using a quasi-Newton method like L-BFGS. The classification of variables is updated after each iteration depending on the variable values.

Other approaches like sequential quadratic programming (SQP) methods have also been used to solve NLPs. These methods linearize the constraints and use a quadratic approximation of the Lagrange function at each iterate as shown below. More details on these type of methods can be found in [4].

$$\min_{d_x} \quad \nabla f(x^k)^T d_x + \frac{1}{2} d_x^T W^k d_x \tag{2.28a}$$

$$s.t. \quad c(x^k) + \nabla c(x^k)^T d_x = 0, \tag{2.28b}$$

$$x_L \leq x^k + d_x \leq x_U \tag{2.28c}$$

where $W^k = \nabla^2_{xx} \mathcal{L}(x, \lambda, z) = \nabla^2_{xx} f(x^k) + \sum_j \nabla^2_{xx} c_j(x^k) \lambda_j^k$

Following NLP software codes were used in this thesis:

- **IPOPT**: **I**nterior **P**oint **OPT**imizer is an open-source C/C++ code written for solving large scale NLPs. IPOPT uses a line search filter with interior point algorithm along with other heuristics mentioned in the paper by Wächter and Biegler [5].
- **CONOPT**: **CON**tinuous **OPT**imizer is a NLP solver which uses generalized reduced gradient method or SQP method along with sparse linear solvers for variable decomposition. It was first developed by Drud [6] and is robust for NLPs with highly nonlinear terms in objective function and constraints

2.4 NLP Parameter Sensitivity

Generally NLPs also consist of parameters which are fixed in the optimization problem:

$$\min_{x \in \mathbb{R}^n} \quad f(x,p) \tag{2.29a}$$

$$s.t. \quad c(x,p) = 0, \tag{2.29b}$$

$$x \geq 0 \tag{2.29c}$$

Consider the NLP solution $s^*(p) = [x^*(p), \lambda^*(p), z^*(p)]^T$ as a function of parameter p. If the NLP solution satisfies LICQ, SOSC and strict complementarity at some p_0, then the function $s^*(p)$ is continuously differentiable in some neighborhood $p \in N_p(p_0)$.

The value of $s^*(p)$ at any point $p \in N_p(p_0)$ can be approximated using the sensitivity matrix which is derived using *Implicit Function Theorem (IFT)* on the KKT conditions (2.13).

Assume the function $K(x, \lambda, z, p)$ represents the KKT conditions:

$$K(x^*, \lambda^*, z^*, p_0) = \begin{bmatrix} \nabla f(x^*, p_0) + \nabla c(x^*, p_0)\lambda^* - z^* \\ c(x^*, p_0) \\ X^* z^* - \mu e \end{bmatrix} = 0 \tag{2.30}$$

Applying IFT on function $K(x^*, \lambda^*, z^*, p_0)$ results in:

$$\nabla_s K(s^*, p_0)^T \left(\frac{ds^*}{dp}\right)^T + \nabla_p K(s^*, p_0)^T = 0 \tag{2.31}$$

The matrix $\nabla_s K(s^*, p_0)^T$ is nothing but the KKT matrix in Eq.(2.14) whereas $\nabla_p K(s^*, p_0)^T$ can be easily calculated as follows:

$$\nabla_s K(s^*, p_0)^T = \begin{bmatrix} W^* & \nabla c(x^*) & -I \\ \nabla c(x^*)^T & 0 & 0 \\ Z^* & 0 & X^* \end{bmatrix}, \nabla_p K(s^*, p_0)^T = \begin{bmatrix} \nabla^2_{xp} f(x^*)^T + \nabla^2_{xp} c(x^*)^T \lambda^* \\ \nabla_p c(x^*, p_0)^T \\ 0 \end{bmatrix} \tag{2.32}$$

The function $s^*(p)$ can thus be linearly approximated using Taylor series:

$$s^*(p) = s^*(p_0) + \left(\frac{ds^*}{dp}\right)^T (p - p_0) + O(||p - p_0||^2) \tag{2.33}$$

k_aug[7] is an IPOPT-based NLP sensitivity solver which calculates the sensitivity matrix

2.5 Dynamic Optimization

Dynamics optimization is a type of optimization problems where some equality constraints are differential in nature. A lot of engineering applications require modeling system dynamics with differential equations. Most design and control applications include modeling spatial or temporal dynamics using ordinary or partial derivatives of state variables. Thus, optimal design and optimal control problems are the most common dynamic optimization problems solved in engineering.

Differential equations can be classified into:

- Ordinary Differential Equation(ODE): The equation has simple derivatives with the differential variable and its derivatives depending on only one independent variable.

$$F(z(x), z'(x), z''(x), ...) = 0 \tag{2.34}$$

- Partial Differential Equation(PDE): The equation has partial derivatives with the differential variable and its derivatives depending on more than one independent variables.

$$F(z(x,t), z'_x(x,t), z'_t(x,t), ...) = 0 \tag{2.35}$$

Most engineering models also have algebraic equations along with differential equations. These type of models are called differential algebraic equations (DAE) and are common in optimal control and design problems.

A semi-explicit DAE is written as:

$$\frac{dz}{dx} = f(z(x), y(x), u(x), p), \quad z(0) = z_0 \tag{2.36a}$$

$$g(z(x), y(x), u(x), p) = 0 \tag{2.36b}$$

where $z(x)$ - differential variables, $y(x)$ - algebraic variables and $u(x)$ - decision variables

DAE index: The number of times the algebraic equations $g(.)$ have to be differentiated to convert the DAE system into a pure ODE system.

DAE index is a measure of complexity of the DAE system. Higher index systems (with DAE index > 2) are much more difficult to solve and require advanced numerical methods.

2.5.1 Discretization

Solving differential equations numerically require discretizing them into discrete algebraic equations. There exist multiple discretization methods for different types of differential equations. In this thesis, two most common types are discussed and used: 1) Finite Difference method and 2) Finite Element method.

2.5.1.1 Finite Difference method

Finite Difference method (FDM) discretizes a differential equation by approximating the derivatives using Taylor series on the differential variable (z). This is done by discretizing the domain of the scalar independent variable (x) as ($a = x_1 < x_2 < ... < x_N = b$)

$$\frac{dz}{dx} \approx D_f z(x_i) = \frac{z(x_{i+1}) - z(x_i)}{x_{i+1} - x_i} + O(\Delta x) \tag{2.37a}$$

The Eq.(2.37a) is called Forward Difference formula or Explicit Euler and if x_{i+1} is replace by x_{i-1}, then it is called the Backward Difference formula or Implicit Euler. Both these formula are only first order accurate since the approximation error is of the order Δx.

More accurate finite difference formula for first derivative can be derived as:

$$\frac{dz}{dx} \approx D_c z(x_i) = \frac{z(x_{i+1}) - z(x_{i-1})}{x_{i+1} - x_{i-1}} + O(\Delta x^2) \tag{2.37b}$$

Similarly second order derivative can be approximated using nodal values as follows:

$$\frac{d^2z}{dx^2} \approx D^2 z(x_i) = \frac{z(x_{i+1}) - 2z(x_i) + z(x_{i-1})}{(x_{i+1} - x_i)^2} + O(\Delta x^2) \tag{2.38}$$

FDMs are easily implementable and have nice convergence properties for continuously differentiable functions. The finite difference formulae also follow a binomial pattern and can be easily derived for a given derivative order and required order of accuracy.

$$\frac{d^n z}{dx^n} \approx D^n z(x_i) = \left[\sum_{j=0}^{n} (-1)^j \binom{n}{j} z(x_{i+(\frac{n}{2}-j)}) \right] / h^n \tag{2.39}$$

FDMs struggle with complex domain geometry of the independent variable and non differentiable solutions.

2.5.1.2 Finite Element method

Finite element method (FEM) is an advanced discretization technique which uses an integral form or the weak form to solve the differential equation. The method approximates the solution as a linear combination of (ϕ) basis functions (mostly polynomials).

$$z(x) \approx \sum_i z_i \phi_i(x) \tag{2.40}$$

The approximation solution is then substituted into the differential equation to form the residual function. The inner product of the residual and a weight function $w(x)$integrated over the $x-$domain is known as the weak form of the differential equation.

$$\int_a^b R(x)w(x)dx = \int_a^b \left(\sum_i z_i \phi_i'(x) - f(x) \right) w(x)dx = 0 \tag{2.41a}$$

$$\sum_i z_i \left(\int_a^b \phi_i'(x) w_j(x)dx \right) - \int_a^b f(x)w_j(x)dx = 0 \tag{2.41b}$$

In FEM, the weight function is chosen to be same as the basis function which transforms the weak form (Eq.2.41b) into a linear system of equations.

$$Az = b \tag{2.42}$$

where $A_{ij} = \int_a^b \phi_i'(x)\phi_j(x)dx$ and $b_j = \int_a^b f(x)\phi_j(x)dx$

FEM extends naturally for higher order derivatives with the integral form simplified after applying integration by parts.

$$\int_a^b \phi_i''(x)\phi_j(x)dx = \left[\phi_i'(x)\phi_j(x)\right]_a^b - \int_a^b \phi_i'(x)\phi_j'(x)dx \tag{2.43}$$

FEM can handle non-differentiable functions and complex geometrical domain because it uses the integral or weak form to solve the differential equation. It is also easier to increase the accuracy of the solution by choosing more accurate basis functions.

Other methods like Finite Volume method (FVM), Orthogonal Collocation method (OCM) and Discontinuous Galerkin method (DGM) are quite similar to FEM. More details on discretization methods and their properties are provided by Fletcher [8]. In this thesis, FDM and FEM are the two discretization methods used to discretize DAE systems.

2.5.2 Optimality Conditions

Similar to NLPs, solutions to dynamic optimization problems satisfy certain optimality conditions also known as Euler-Lagrange equations.

Consider the DAE optimization problem :

$$min \quad \phi(z(x_f)) \tag{2.44a}$$

$$s.t. \quad \frac{dz}{dx} = f(z(x), y(x), u(x), p), \quad z(0) = z_0, \tag{2.44b}$$

$$g_E(z(x), y(x), u(x), p) = 0, \tag{2.44c}$$

$$g_I(z(x), y(x), u(x), p) \leq 0, \tag{2.44d}$$

$$h_E(z(x_f)) = 0, \quad h_I(z(x_f)) \leq 0 \tag{2.44e}$$

The objective function and constraints Eq.(2.44e) depend only on the value of the differentiable variable at the boundary. Other forms of DAE optimization with integral in the objective can also be transformed into the above form. Assuming the functions and the solution are smooth and the DAE is index 1, the Hamiltonian function can be defined as:

$$H(x) := f(z, y, u, p)^T \lambda + g_E(z, y, u, p)^T \nu_E + g_I(z, y, u, p)^T \nu_I + h_E(z(x_f))^T \eta_E + h_I(z(x_f))^T \eta_I \tag{2.45}$$

Here $\lambda, \nu_E, \nu_I, \eta_E$ and η_I are adjoint variables which are also function of x. The Euler-Lagrange conditions for (2.44) are:

$$\frac{dz}{dx} = \frac{\partial H}{\partial \lambda} = f(z(x), y(x), u(x), p), \quad z(0) = z_0, \tag{2.46a}$$

$$\frac{d\lambda}{dx} = -\left[\frac{\partial f}{\partial z}\lambda + \frac{\partial g_E}{\partial z}\nu_E + \frac{\partial g_I}{\partial z}\nu_I\right] = -\frac{\partial H}{\partial z}, \tag{2.46b}$$

$$\lambda_f = \frac{\partial \phi(z_f)}{\partial z} + \frac{\partial h_E(z_f)}{\partial z}\eta_E + \frac{\partial h_I(z_f)}{\partial z}\eta_I, \tag{2.46c}$$

$$g_E(z(x), y(x), u(x), p) = \frac{\partial H}{\partial \nu_E} = 0, \tag{2.46d}$$

$$0 \leq \nu_I \perp g_I(z(x), y(x), u(x), p) \leq 0, \tag{2.46e}$$

$$h_E(z(x_f)) = \frac{\partial H}{\partial \eta_E} = 0, \tag{2.46f}$$

$$0 \leq \eta_I \perp h_I(z(x_f)) \leq 0, \tag{2.46g}$$

$$\frac{\partial f}{\partial y}\lambda + \frac{\partial g_E}{\partial y}\nu_E + \frac{\partial g_I}{\partial y}\nu_I = \frac{\partial H}{\partial y} = 0, \tag{2.46h}$$

$$\frac{\partial f}{\partial u}\lambda + \frac{\partial g_E}{\partial u}\nu_E + \frac{\partial g_I}{\partial u}\nu_I = \frac{\partial H}{\partial u} = 0, \tag{2.46i}$$

$$\frac{\partial z_0}{\partial p}\lambda(0) + \int_0^{x_f} \frac{\partial H}{\partial p} dx = 0 \tag{2.46j}$$

The optimality conditions (2.46) is a *Boundary-Value Problem* (BVP) with initial conditions for differential state variable z and boundary conditions for differential adjoint variable λ. Solving the BVP can sometimes be very challenging and there are multiple ways to handle the DAE optimization problem.

2.5.3 Solution Strategy

DAE optimization problems can be visualized as a mix of DAE problem inside an optimization problem or vice-versa. Depending on which is handled first, a solution strategy can be classified into two categories: 1) *Optimize-Discretize* and 2) *Discretize-Optimize*.

2.5.3.1 Optimize-Discretize (OD)

In this approach, the optimality conditions are applied first and the resulting Euler-Lagrange conditions are discretized later. As shown in the previous section, the Euler-Lagrange system is a BVP with variational inequalities which makes it very difficult to solve using simple numerical methods. BVPs also need to satisfy additional properties like *Dichotomy* which is required for the stability of the solutions under perturbation in initial or boundary conditions.

Still, discretization methods like FDM or FEM can be applied to Euler Lagrange conditions with the resultant nonlinear algebraic equations solved using Newton-type methods.

Other approaches like shooting methods or Runge-Kutta methods can also be applied to solve the differential system using sensitivity information to satisfy the boundary conditions.

Handling the inequalities and complementarity conditions is not straightforward and requires methods like interior-point, projection and penalty based methods. Anther disadvantage of the OD approach is that it does not enforce a descent direction criterion and might converge to non optimal stationary points.

2.5.3.2 Discretize-Optimize (DO)

An alternate to OD approach, the DO approach handles the differential equations first and then solves an optimization problem with algebraic constraints. There are two ways to apply DO method: Sequential approach and Simultaneous approach.

Sequential approach fixes the decision variables $u(x)$ and solves the DAE system first, then it calculates the sensitivity of the solution (state variables z, y) with respect to the input u. Using this sensitivity information, the gradient of the objective is calculated and used by the NLP solver to update the value of decision variables in an outer loop. This approach requires solving the DAE system multiple times which can significantly slow convergence.

Simultaneous approach discretizes the whole DAE system into a NLP with a large system of discretized equations and inequalities. The large-scale NLP can be solved using any of the NLP solvers by exploiting the sparsity pattern of the KKT matrix of the discretized system. The approach does not require any sensitivity information or multiple solution of the DAE systems with fixed input, making it more competitive than the sequential approach. Under special conditions, the discretized Euler-Lagrange conditions and the KKT conditions of the discretized large-scale NLP are analogous.

In this thesis, the *simultaneous approach* is used to solve a DAE optimization problem by discretizing the DAE using either FDM or FEM. The resultant NLP is then solved using IPOPT or CONOPT as the NLP solver with k_aug used to calculate NLP sensitivity.

2.6 Equilibrium and Complementarity Constraints

Many engineering systems require modeling switching or logic based decisions and constraints. These type of systems can be modeled using *mixed integer programming* (MIP) or they can be modeled using equilibrium and complementarity constraints. Nonlinear MIPs or MINLPs with large number of integer variables are notoriously difficult to solve and can take a lot of computational time to converge. Modeling the system with equilibrium or complementarity constraints is a better alternative and these type of optimization problems are known as *Mathematical Programming with Equilibrium Constraints* (MPEC) or *Mathematical Programming with Complementarity Constraints* (MPCC).

2.6.1 MPEC and MPCC

General MPECs can be represented as:

$$\min_{x,y,z} \quad f(x,y,z), \tag{2.47a}$$

$$s.t. \quad g(x,y,z) \leq 0, \tag{2.47b}$$

$$h(x,y,z) = 0, \tag{2.47c}$$

$$y = arg\{\min_{\hat{y}} x^T \hat{y}, \hat{y} \geq 0\}, \;\; x \geq 0 \tag{2.47d}$$

The last equation Eq.2.47d is called the equilibrium constraint because the variables y and x are constrained by an equilibrium optimization problem. Another way to represent the above problem is using complementarity constraints as shown here:

$$\min_{x,y,z} \quad f(x,y,z), \tag{2.48a}$$

$$s.t. \quad g(x,y,z) \leq 0, \tag{2.48b}$$

$$h(x,y,z) = 0, \tag{2.48c}$$

$$0 \leq y \perp x \geq 0 \tag{2.48d}$$

The last equation Eq.(2.48d) is called the complementarity constraint where $\perp$ is the complementarity operator.

The complementarity constraint Eq.(2.48d) represents the following logic statement:

$$x_i = 0 \quad \text{OR} \quad y_i = 0, \quad x, y \geq 0, \quad i = 1, ...n_c \tag{2.49}$$

here the OR operator is inclusive as both x and y can be equal to zero simultaneously.

The following mathematical formulations are equivalent to the complementarity constraint:

$$x^T y = 0, x \geq 0, y \geq 0 \tag{2.50a}$$

$$x_i y_i = 0, x \geq 0, y \geq 0 \tag{2.50b}$$

The formulations are also equivalent with the equality sign replaced by $\leq$ inequality.

Note that a similar formulation is also used in KKT conditions Eq.(2.4) discussed earlier in the Chapter. The original KKT conditions for NLP problems contain complementarity condition relating the inequalities and its multipliers which are formulated as equations.

Since MPECs and MPCCs are equivalent and interchangeable, in the rest of the thesis the MPECs are presented in the form of Eq.(2.48).

2.6.2 Properties and Formulations

Unlike general NLPs, MPECs with formulations (2.50) do not satisfy LICQ and MFCQ at any feasible point and thus the necessary KKT conditions cannot be applied to solve MPECs. To satisfy optimality conditions for MPECs, the LICQ and MFCQ definitions are modified as such using a *relaxed nonlinear problem (RNLP)*.

2.6.2.1 MPEC-CQs, RNLP and Stationarity conditions

For a feasible point (x^*, y^*, z^*) of MPEC (2.48), the corresponding RNLP is written as:

$$\min_{x,y,z} \quad f(x, y, z), \tag{2.51a}$$

$$s.t. \quad g(x, y, z) \leq 0, \tag{2.51b}$$

$$h(x, y, z) = 0 \tag{2.51c}$$

$$x_i = 0, \quad i \in I_X \setminus I_Y, \tag{2.51d}$$

$$y_i = 0, \quad i \in I_Y \setminus I_X, \tag{2.51e}$$

$$x_i \geq 0, y_i \geq 0 \quad i \in I_X \cap I_Y \tag{2.51f}$$

where $I_X = \{i : x_i^* = 0\}$ and $I_Y = \{i : y_i^* = 0\}$ are active sets at the feasible point. $I_{XY} = I_X \cap I_Y$ is also called the bi-active set.

MPEC-LICQ or MPEC-MFCQ is defined by the LICQ or MFCQ of the RNLP at a feasible point (x^*, y^*, z^*). Further if (x^*, y^*, z^*) is a solution to the RNLP, then it is called a *strongly stationary point* of the MPEC.

Stationary Conditions

The Lagrange function of MPEC (2.48) is defined as:

$$\mathcal{L} = f(x, y, z) + \lambda^T g(x, y, z) + \alpha^T h(x, y, z) - \mu^T x - \nu^T y \tag{2.52}$$

Similar to KKT conditions, weak stationary conditions exist for MPEC which are satisfied by multiple types of stationarity points.

$$\nabla_x f + \lambda^* \nabla_x g + \alpha^* \nabla_x h - \mu^* = 0, \tag{2.53a}$$

$$\nabla_y f + \lambda^* \nabla_y g + \alpha^* \nabla_y h - \nu^* = 0, \tag{2.53b}$$

$$\nabla_z f + \lambda^* \nabla_z g + \alpha^* \nabla_z h = 0, \tag{2.53c}$$

$$g(x^*, y^*, z^*) \leq 0, \quad h(x^*, y^*, z^*) = 0, \tag{2.53d}$$

$$0 \leq \lambda^* \perp g^* \leq 0, \tag{2.53e}$$

$$0 \leq x^* \perp y^* \geq 0, \tag{2.53f}$$

$$x_i^* > 0 \Longrightarrow \mu_i^* = 0, \;\; y_i^* > 0 \Longrightarrow \nu_i^* = 0 \tag{2.53g}$$

Note that the complementarity bound multipliers μ_i^* and ν_i^* need not necessarily be non-negative when both variables are at their bounds $x_i^* = y_i^* = 0$ (or weakly active). Negative multipliers mean that there exists feasible descent direction and the point is not locally optimum. Depending on the multiplier values, the stationary points are classified as:

- **C-stationary point**: $\mu_i \nu_i \geq 0, \forall i \in I_{XY}$
- **A-stationary point**: $\mu_i \geq 0$ or $\nu_i \geq 0, \forall i \in I_{XY}$
- **M-stationary point**: ($\mu_i \geq 0$ and $\nu_i \geq 0$) or $\mu_i \nu_i \geq 0, \forall i \in I_{XY}$
- **Strongly stationary point**: $\mu_i \geq 0$ and $\nu_i \geq 0, \forall i \in I_{XY}$

Strongly stationary point is the strongest or sufficient condition of stationarity for MPECs but there exists a tightest or necessary condition for MPECs called *Boulingard or B-stationary point*. A point (x^*, y^*, z^*) is a B-stationary point iff $d = 0$ is the solution to the LPEC:

$$\min_{d} \quad \nabla f(x^*, y^*, z^*)^T d, \tag{2.54a}$$

$$s.t. \quad g(x^*, y^*, z^*) + \nabla g(x^*, y^*, z^*)^T d \leq 0, \tag{2.54b}$$

$$h(x^*, y^*, z^*) + \nabla h(x^*, y^*, z^*)^T d = 0, \tag{2.54c}$$

$$0 \leq y^* + d_y \perp x^* + d_x \geq 0 \tag{2.54d}$$

Note that every strongly stationary is also a B-stationary point but the other way is only true if MPEC-LICQ is satisfied at the point or $I_{XY} = \emptyset$.

Second Order Conditions Second order conditions for MPECs are more involved and require more definitions and details which can be found in Ralph and Wright [9]. Nonetheless, following relation holds between second order conditions of MPEC and RNLP.

RNLP-SSOSC $\implies$ RNLP-SOSC $\implies$ MPEC-SOSC and MPEC-SSOSC $\implies$ MPEC-SOSC

2.6.2.2 NLP based Reformulations

Since the MPECs do not satisfy general LICQ and MFCQ conditions, it is difficult to solve them using standard NLP solvers. For this reason, MPECs are reformulated into NLPs as shown here:

Reg(ϵ)

$$\min_{x,y,z} \quad f(x, y, z) \tag{2.55a}$$

$$s.t. \quad g(x, y, z) \leq 0, \tag{2.55b}$$

$$h(x, y, z) = 0, \tag{2.55c}$$

$$x_i y_i \leq \epsilon \quad i = 1....n_c, \tag{2.55d}$$

$$x \geq 0, y \geq 0 \tag{2.55e}$$

RegComp(ϵ)

$$\min_{x,y,z} \quad f(x, y, z) \tag{2.56a}$$

$$s.t. \quad g(x, y, z) \leq 0, \tag{2.56b}$$

$$h(x, y, z) = 0, \tag{2.56c}$$

$$x^T y \leq \epsilon, \tag{2.56d}$$

$$x \geq 0, y \geq 0 \tag{2.56e}$$

RegEq(ϵ)

$$\min_{x,y,z} \quad f(x, y, z) \tag{2.57a}$$

$$s.t. \quad g(x, y, z) \leq 0, \tag{2.57b}$$

$$h(x, y, z) = 0, \tag{2.57c}$$

$$x_i y_i = \epsilon \quad i = 1....n_c, \tag{2.57d}$$

$$x \geq 0, y \geq 0 \tag{2.57e}$$

PF(ρ)

$$\min_{x,y,z} \quad f(x, y, z) + \rho x^T y \tag{2.58a}$$

$$s.t. \quad g(x, y, z) \leq 0, \tag{2.58b}$$

$$h(x, y, z) = 0, \tag{2.58c}$$

$$x \geq 0, y \geq 0 \tag{2.58d}$$

NCP(ϵ) The RegEq(ϵ) can also be formulated using a NCP function by replacing the Eq.(2.57d)

$$x_i + y_i - \sqrt{x_i^2 + y_i^2 + \epsilon} = 0 \tag{2.59}$$

The NLP regularizations either relax or penalize the complementarity constraint using a positive parameter ϵ or ρ respectively. These regularizations satisfy both LICQ and MFCQ property and thus can be solved using any NLP solver. The relaxation based approaches Reg(ϵ), RegComp(ϵ) and RegEq(ϵ) (or NCP(ϵ)) solve a sequence of NLPs with decreasing values of ϵ. The solution from each solve $w(\epsilon) = [x^T(\epsilon), y^T(\epsilon), z^T(\epsilon)]^T$ is used as the initial point for the next solve with lower ϵ.

The rate of convergence for these relaxation based approaches can be defined as the norm distance between the solution of the relaxed NLP ($w(\epsilon)$) and the MPEC solution (w^*). The rate of convergence results are presented by Ralph and Wright [9] as:

1. *If w^* satisfies MPEC-LICQ (or MPEC-MFCQ) and RNLP-SOSC (or MPEC-SOSC) and $w(\epsilon)$ is the solution to Reg(ϵ)-(2.55) or RegComp(ϵ)-(2.56). Then for some constant values of $r, \bar{\epsilon}$ and M, such that for $\epsilon \in [0, \bar{\epsilon}]$ if the solution $w(\epsilon)$ satisfies the condition $||w(\epsilon) - w^*|| \leq r/2$(or r) then the distance between the solutions is bounded by $||w(\epsilon) - w^*|| \leq M\epsilon$ (or $||w(\epsilon) - w^*|| \leq M\epsilon^{1/2}$)*

2. *If w^* satisfies MPEC-LICQ and MPEC-SOSC and $w(\epsilon)$ is the solution to RegEq(ϵ)-(2.57) or RegEq(ϵ)-(2.59). Then for some constant values of $r, \bar{\epsilon}$ and M, such that for $\epsilon \in [0, \bar{\epsilon}]$ if the solution $w(\epsilon)$ satisfies the condition $||w(\epsilon) - w^*|| \leq r$ then the distance between the solutions is bounded by $||w(\epsilon) - w^*|| \leq M\epsilon^{1/4}$*

On the other hand the penalty approach PF(ρ) directly puts the complementarity constraint in the objective multiplied by a penalty parameter ρ. It can be shown that if $\rho \geq \rho_c$, then the solution of PF(ρ) converges to the MPEC solution. Unfortunately the critical value ρ_c is not known apriori and the penalty parameter ρ needs to be increased iteratively until the NLP PF(ρ) converges to a solution which satisfies the complementarity constraints.

Theorem: *If $w(\rho)$ is a stationary point of PF(ρ) and the complementarity condition is satisfied $x(\rho)^T y(\rho) = 0$, then the stationary point is equal to the strongly stationary point of MPEC (w(ρ) = w^*). Moreover, if $w(\rho)$ satisfies LICQ for PF(ρ), then it also satisfies MPEC-LICQ*

2.6.2.3 PC^1 functions

Piecewise continuously differentiable PC^1 functions are defined as continuous functions which are continuously differentiable in sub-intervals but have finite non-differentiable kinks or points [10]. Common examples include: $max(0,x), min(0,x), |x|$ and $tri(x)$. These functions are common in engineering models and can be reformulated using complementarity constraints as shown here.

Max function - max(0,x)

$$x = s_p - s_n, \tag{2.60a}$$

$$0 \leq s_p \perp s_n \geq 0, \tag{2.60b}$$

$$y = s_p \tag{2.60c}$$

Min function - min(0,x)

$$x = s_p - s_n, \tag{2.61a}$$

$$0 \leq s_p \perp s_n \geq 0, \tag{2.61b}$$

$$y = -s_n \tag{2.61c}$$

Absolute value function - $|x|$

$$x = s_p - s_n, \tag{2.62a}$$

$$0 \leq s_p \perp s_n \geq 0, \tag{2.62b}$$

$$y = s_p + s_n \tag{2.62c}$$

Triangular or hat function - tri(x) $:= max(0, 1 - |x|)$

$$x = s_p - s_n, \tag{2.63a}$$

$$0 \leq s_p \perp s_n \geq 0, \tag{2.63b}$$

$$1 - (s_p + s_n) = L_p - L_n, \tag{2.63c}$$

$$0 \leq L_p \perp L_n \geq 0, \tag{2.63d}$$

$$y = L_p \tag{2.63e}$$

2.6.3 Hybrid Dynamic Systems

Hybrid dynamic systems are very common in engineering applications and are characterized with piecewise formulations. Hybrid dynamic systems can be defined as systems where the different equation form varies with the value of the state variable.

For example:

$$\frac{dz}{dx} = \begin{cases} f_-(z,u,y) & \sigma(z,x) \leq 0 \\ \nu f_-(z,u,y) + (1-\nu) f_+(z,u,y) & \sigma(z,x) = 0 \\ f_+(z,u,y) & \sigma(z,x) \geq 0 \end{cases} \tag{2.64a}$$

$$h(z,u,y) = 0, \tag{2.64b}$$

$$g(z,u,y) \leq 0 \tag{2.64c}$$

where z, y - state variable, u - decision variable, $\nu \in [0,1]$ - switching variable
f_-, f_+ - functional forms, σ - switching function

These type of hybrid dynamic systems are known as *Filippov* system where the right hand side of the differential equation is non-smooth. Filippov systems can be reformulated using complementarity constraints as follows:

$$\frac{dz}{dx} = \nu f_-(z,u,y) + (1-\nu) f_+(z,u,y) \tag{2.65a}$$

$$\sigma = s_p - s_n \tag{2.65b}$$

$$0 \leq s_p \perp \nu \geq 0 \tag{2.65c}$$

$$0 \leq s_n \perp 1 - \nu \geq 0 \tag{2.65d}$$

$$h(z,u,y) = 0 \tag{2.65e}$$

$$g(z,u,y) \leq 0 \tag{2.65f}$$

The above DAE is index-1 unless the switching variable is zero ($\sigma = 0$), when it becomes an index-2 DAE. Therefore, a high index DAE needs to be reformulated such that its index is 1 before being discretized using FEM or FDM.

Another aspect of solving hybrid dynamic systems is the location of the kinks or non-smooth point in the solution. Since numerical discretization based integration methods assume some type of smoothness property of the solution, it is necessary to determine the non-smooth point(s) such that the assumptions are valid in every discretization interval.

Moving grid strategy is one way to determine the kink in the solution profile. In this method, the step size(s) are treated as free variables and are allowed to vary within some bounds ($\Delta x^L \leq \Delta x_i \leq \Delta x^U$). Added condition(s) are imposed which ensure that the switching function σ is equal to zero at least at one grid point. More details on this strategy are given in Baumrucker et al. [11].

2.6.3.1 Flash Problem

A common chemical engineering application - flash tank is used to demonstrate the topics discussed in this chapter. A flash tank is used to separate two-phase (*mixture of liquid and vapor*) stream into individual liquid and vapor streams. It is commonly used before compressors, pumps or equipments to remove unwanted liquid or vapor in the stream.

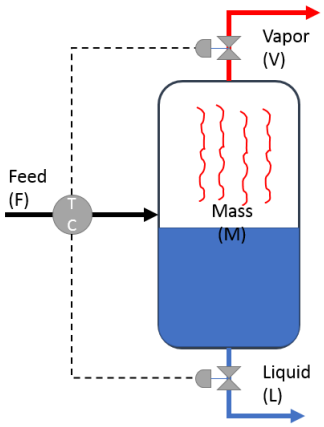


Figure 2.1: Flash Tank

Assume a single component stream is passing through the flash tank and the optimiza-

tion problem is to minimize the difference between the liquid outlet profile and the desired liquid output profile. The control variables are temperature and pressure inside the tank. The optimization problem can be written as:

$$\min_{T,P} \quad (L(t) - \bar{L}(t))^2 \tag{2.66a}$$

$$s.t. \quad \frac{dM}{dt} = \begin{cases} F - L & T - T_b(P) \leq 0 \\ F - L - V & T - T_b(P) = 0 \\ F - V & T - T_b(P) \geq 0 \end{cases} \qquad M(0) = M_{init} \tag{2.66b}$$

Eq.(2.66) can be reformulated into:

$$\min_{T,P} \quad (L(t) - \bar{L}(t))^2 \tag{2.67a}$$

$$s.t. \quad \frac{dM}{dt} = F - L - V, \quad M(0) = M_{init} \tag{2.67b}$$

$$T - T_b(P) = s_1 - s_2 \tag{2.67c}$$

$$0 \leq L \perp s_1 \geq 0 \tag{2.67d}$$

$$0 \leq V \perp s_2 \geq 0 \tag{2.67e}$$

The above DAE system can then be discretized using either FDM or FEM into:
For i = 1....N and $\Delta t = \frac{t_f}{N}$

$$\min_{T,P} \quad \sum_i (L_i - \bar{L}_i)^2 \tag{2.68a}$$

$$s.t. \quad \frac{M_{i+1} - M_{i-1}}{\Delta t} = F_i - L_i - V_i, \quad M_0 = M_{init} \tag{2.68b}$$

$$T_i - T_b(P_i) = s_{1i} - s_{2i} \tag{2.68c}$$

$$0 \leq L_i \perp s_{1i} \geq 0 \tag{2.68d}$$

$$0 \leq V_i \perp s_{2i} \geq 0 \tag{2.68e}$$

The complementarity constraints Eq.(2.68d-e) can be reformulated using any of the NLP reformulations (Reg, RegComp, RegEq or PF). The resultant NLP can then be solved using a KKT based solver like IPOPT or CONOPT.

2.7 Summary

In summary, the chapter presents a brief introduction to nonlinear optimization and nonlinear programming (NLP). Notions of local and global optimum are discussed along with definitions of convex function and convex set. Optimality conditions (first and second order) for NLPs are presented along with discussion on constraint qualifications (CQs).

NLP algorithms - Interior point method and Gradient projection method are presented along with details about KKT matrix, line search filter and code implementations. Sensitivity of NLP solution with respect to model parameter is discussed along with details on sensitivity matrix.

Dynamic optimization is introduced with characterization of types of differential equations - ODE, DAE and PDE. Discretization methods like finite difference (FDM) and finite element (FEM) are presented along with derivations and formulae. Optimality conditions for DAE optimization are presented and its solution strategies (optimize-discretize and discretize-optimize) are discussed with advantages and disadvantages for both approaches.

Equilibrium and complementarity constraints are introduced with definitions for MPEC and MPCC. Relaxed nonlinear problem (RNLP) and MPEC-CQs are defined along with first order stationary conditions and different types of stationary points. Different reformulations into NLPs are discussed along with results on convergence to the original solution are presented. Complementarity based formulations of commonly used PC^1 functions are given. Hybrid dynamic systems are introduced and formulation using complementarity constraints is derived.

Finally, a simple example of flash tank vessel is used to demonstrate the concepts of NLP, DAE and MPEC optimization simultaneously.

Chapter 3

Heat Exchanger Design

In this chapter, the heat exchanger design problem is introduced and discussed in terms of modeling and optimization. Current methods based on log mean temperature difference (LMTD) are discussed along with the assumptions required for its application. A new DAE based model without many assumptions is developed using first principles PDE conservation equations. Discretization based on exchanger geometry and numerical convergence of the method are discussed. An iterative solution strategy is proposed to calculate the overall design of a heat exchanger and is applied on few examples.

In the second part, a design model for phase change heat exchangers is developed using the DAE method and complementarity constraints. The model is used for flooded evaporator and condenser design with a varying local heat transfer coefficient.

3.1 Introduction

Heat exchangers are an important part of any process plant, as they are primarily used for heating and cooling any process stream. Most popular type of heat exchanger in process industries is the shell and tube heat exchanger (STHE) because of its compact design and versatile applicability. They have been widely studied and used in the chemical process industry for various applications like heat recovery, refrigeration etc. Although there exists literature on detailed design of STHE, the process optimization models rely on simplistic and inaccurate shortcut models. These models are primarily based on LMTD formula which has strong assumptions and do not consider the geometrical design of the exchanger. There also exists PDE based computational fluid dynamics (CFD) simulation

design methods [12, 13] but they are computationally very expensive and cannot be directly used for flowsheet optimization.

In engineering textbooks [14, 15], the heat exchanger design is performed by first assuming the value of some variable and then sequentially solving a system of nonlinear equations. These equations are based on LMTD formula, geometrical calculations and general heuristics. The design is then adjusted in a superficial way if the assumed value of the variable does not match its calculated value within some bounds. This method cannot guarantee convergence to an optimal or feasible solution and is prone to fail.

Simultaneously solving the system of nonlinear equations using a Newton based equation solver can overcome these difficulties but this requires the equations to be continuously differentiable. Unfortunately, the LMTD formula and some other geometrical equations are non-differentiable or non-smooth and are mostly approximated using a smooth algebraic equation such as Chen or Patterson approximation [16, 17].

Past work [18, 19, 20] on optimal design of STHE have used these system of nonlinear equations as constraints and used integer variables for other discrete design parameters such as number of tubes, baffles, tube diameter etc. to formulate a non-convex MINLP problem. Solving this MINLP problem is not straightforward and requires custom initialization techniques and tight variable bounds making it difficult to apply for general heat exchanger design problems. Bagajewicz and his collaborators [21, 22, 23, 24], developed a smart enumeration based approach which converts the nonlinearities using Glover inequalities and additional integer variables into a large MILP problem. Even though, they claim to guarantee global optimal design, the solution times are very high and the problem may become intractable. Other meta-heuristic methods like particle swarm [25], simulated annealing [26], genetic algorithms [27, 28] etc. have also been applied to solve the MINLP problem.

This chapter describes the LMTD methods and discusses its associated assumptions, thus providing the motivation for a more robust applicable model. Then it presents a

new method inspired by Ravikumaur et al. [29, 30], Moorthy et al. [31] which replaces the LMTD equation by differential equations which model the heat transfer between the streams inside the exchanger locally and does not need the LMTD assumptions. A DAE based design model is thus developed and is applied on a few examples including phase change exchangers to show its advantages in comparison with the LMTD methods.

3.2 LMTD based methods

The heat exchanger design problem can be broadly defined as:
Given:

1. Hot and cold stream(s) with inlet-outlet temperatures and mass flow rates.
2. Stream Properties like specific heat (C_p), density (ρ), viscosity (μ), thermal conductivity (k). These can be constant values or functions of the state of the system.

Find:

1. Stream allocation of hot and cold streams into the shell-side or tube-side.
2. Number of shells, baffles, tube passes and tube size (length, diameter and thickness).
3. Shell size (shell diameter) and Number of tubes.
4. Tube packing layout (triangular, square), head types, baffle cut, etc.

The LMTD is a function of the inlet and outlet temperature calculated using the following equation:

$$LMTD = \frac{\Delta T_2 - \Delta T_1}{log\left(\frac{\Delta T_2}{\Delta T_1}\right)} \tag{3.1}$$

where ΔT_1 and ΔT_2 are the temperature differences between the streams at the inlet and outlet of the exchanger, given by

$$\text{Co-current case} \qquad \Delta T_1 = T_{h,in} - T_{c,in}, \quad \Delta T_2 = T_{h,out} - T_{c,out} \tag{3.2a}$$

$$\text{Counter-current case} \quad \Delta T_1 = T_{h,in} - T_{c,out}, \quad \Delta T_2 = T_{h,out} - T_{c,in} \tag{3.2b}$$

The derivation of the LMTD formula can be found in most heat transfer textbooks like Kern [32]. Note that the LMTD for the counter-current case is always higher than the co-current case, which is the reason that the former is the default choice in most cases. The above LMTD formula is only valid for double-pipe heat exchangers and one shell and one tube pass exchangers (1-1 exchangers), which are rarely used in industry. Therefore a correction factor (F_T) is used to calculate a modified LMTD for heat exchangers with multiple shells and multiple tube passes. These factors are derived using inlet-outlet temperatures as shown in the Appendix (Eq. 1).

The discrete structure of the exchanger, such as number of tube passes, number of baffles, tube size and fluid allocation, are determined by heuristics based on stream velocities and baffle spacing in order to minimize fouling and pressure drops.

The most standard LMTD-based methods in practice are the Kern method [32] and the Bell-Delaware [33] method.

3.2.1 Kern method

Kern's method uses geometry and basic trigonometric concepts to relate shell inlet area and tube dimensions. It calculates the Reynolds numbers on both tube and shell sides using stream velocities and equivalent diameter. The heat transfer coefficients are calculated using Nusselt number correlations, which vary depending on the fluid flow regime. The total pressure drops are calculated by multiplying the per tube pass or per baffle pressure drop with the number of tube passes, number of baffles and number of shells. The equations involved in Kern's method are presented in the Appendix (Eq. 2 and 3).

3.2.2 Bell-Delaware method

The Bell-Delaware method improves the accuracy of shell-side design of Kern's method by adding hydraulic equations that include the effects of baffle-cuts, baffle leakage etc. on the overall solution. These hydraulic equations are non-linear and non-smooth which

often makes them numerically ill-conditioned. The Bell-Delaware method also gives lower exchanger areas than Kern's method as reported in [23]. Due to the presence of highly ill-conditioned non-linear equations, the Bell-Delaware method requires more computational effort as compared to Kern's method.

Both methods are relatively successful in obtaining heat exchanger design for different types of shell and tube heat exchangers but are limited by assumptions listed in Table 3.1.

1.	Streams have constant physical properties and overall heat transfer coefficient is constant inside the heat exchanger,
2.	No phase changes occur inside the heat exchanger,
3.	Negligible heat losses and equal areas exist in each tube-pass
4.	The temperature of the shell fluid is assumed to be a constant average across a cross-section of the exchanger.

Table 3.1: The main assumptions in LMTD based methods

The last assumption is used to derive the F_T correction formulae, where the heat transfer equation inside a multiple tube-pass exchanger is transformed to the single pass exchanger heat equation form used in the LMTD derivation [32, p. 140]. This is a major limitation of the LMTD method, as it becomes inaccurate with multiple shells and multiple-pass STHE which are quite common in practice. The method can be made applicable for phase change exchangers as well, but requires some modifications (Xiao et al. [34]) and results in less accurate solutions.

3.3 DAE method

To overcome these challenges and drawbacks, a new method is presented which is able to design an exchanger without the assumptions in Table 3.1. This new method is based on formulating the heat exchanger design problem as a boundary-value DAE problem. The

DAE model contains a coupled differential equation system which is derived from the heat equation partial differential equation (PDE).

3.3.1 PDE to ODE

The steady state heat equation (Eq.3.3a) is a second order elliptic PDE. The left-hand side term represents convective heat transfer whereas the first term on the right-hand side represents conductive heat transfer. The second term represents external heat transfer which, in our case, is the heat exchange rate between fluids per unit volume:

$$\rho C_p u.\nabla T = \nabla.(k\nabla T) + q_v \tag{3.3a}$$

where ρ is density, C_p is specific heat capacity, u is velocity, k is thermal conductivity, T is temperature, and q_v is the volumetric heat.
The following assumptions are made:

- Zero or negligible heat transfer by conduction.
- Shell and tube side fluid flow in the vertical ($u_y \neq 0$) and horizontal($u_x \neq 0$) directions.

Using these assumptions, Eq.3.3a is simplified for shell and tube sides as:

$$\rho C_p^t u_x \frac{dT_t}{dx} = -q_v \tag{3.3b}$$

$$\rho C_p^s u_y \frac{dT_s}{dy} = q_v \tag{3.3c}$$

Substituting the mass flow rate in tube (shell) sides $m = \rho u_x dydz \;\; (m = \rho u_y dxdz)$ and volumetric heat flux $q_v = \frac{UdA(T_t - T_s)}{dxdydz}$ and simplifying results in

$$mC_p^t \frac{dT_t}{dA} = -U(T_t - T_s) \tag{3.3d}$$

$$mC_p^s \frac{dT_s}{dA} = U(T_t - T_s) \tag{3.3e}$$

Substituting $C_s = m_s C_p^s, \;\; C_t = m_t C_p^t$ results in the following coupled ODE system.

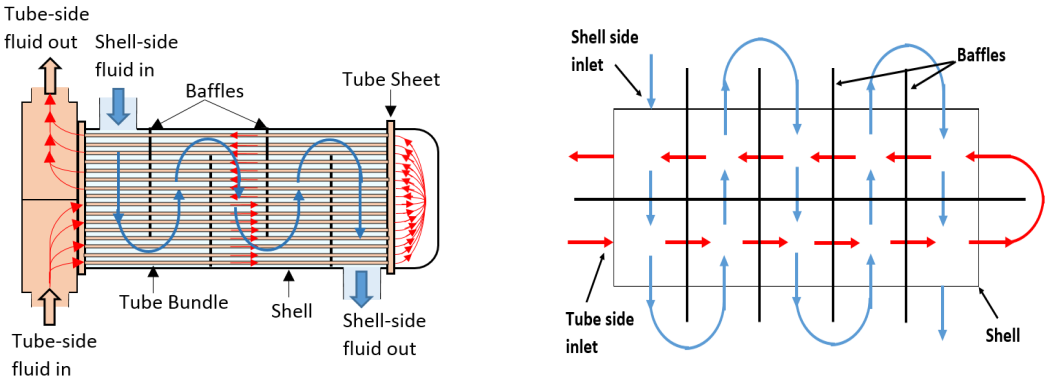
$$C_t \frac{dT_t}{dA} + U(T_t - T_s) = 0 \tag{3.3f}$$

$$C_s \frac{dT_s}{dA} - U(T_t - T_s) = 0 \tag{3.3g}$$

where dA represents the differential area for the heat exchanger and T_s and T_t are shell side and tube side temperatures, respectively. This simplification is based on the fact that even though the individual temperatures depend on x and y, they can be solved simultaneously using a lumped variable A, which is common for both the shell and tube side fluid.

3.3.2 Discretization

To solve the DAE system numerically, the coupled ODEs are discretized using a finite element method. The idea behind this method is to divide the heat exchanger into small elements and discretize the differential equations into a system of algebraic equations. This division of heat exchanger geometry is achieved by using baffles and exchanger shell heads between the passes to create a discrete element structure as shown in Figure 3.1.



(a) Two-pass shell and tube heat exchanger

(b) Heat Exchanger Discrete Structure

Figure 3.1: Discretization of the shell and tube heat exchanger

The minimum total number of elements depends on the number of baffles and number of tube passes which is given by Eq.3.4 where Nfe, N_b and N_{tp} are the number of finite elements, baffles and tube passes respectively.

$$Nfe = (N_b + 1)N_{tp} \tag{3.4}$$

Note that because of the overall mass balance, this discretization is energy conserving for detailed exchanger design, as is analyzed in later sections. The temperature of both

streams is assumed to be linear inside each element and is represented by linear combinations of the hat functions N_1 and N_2.

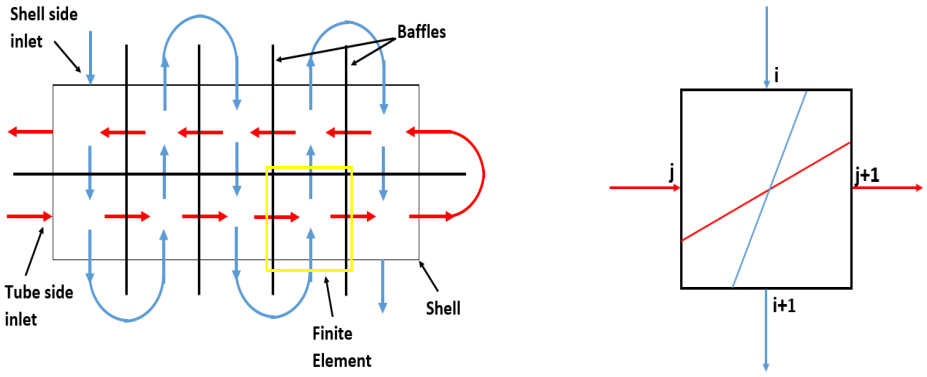
$$T_s(A) = N_1 T_s^i + N_2 T_s^{i+1} \tag{3.5a}$$

$$T_t(A) = N_1 T_t^j + N_2 T_t^{j+1} \tag{3.5b}$$

where $N_1 = 1 - \frac{A}{\Delta A}$ and $N_2 = \frac{A}{\Delta A}$

$\frac{dT_s(A)}{dA} = \frac{(T_s^{i+1} - T_s^i)}{\Delta A}$ and $\frac{dT_t(A)}{dA} = \frac{(T_t^{j+1} - T_t^j)}{\Delta A}$ i- vertical(shell), j-horizontal(tube)

ΔA is the area of each finite element and $A \in [0, \Delta A]$ is the independent variable.



(a) Discrete Structure (b) Finite Element

Figure 3.2: Discrete structure consisting of finite elements

Inserting the candidate solution into the ODE in Eq.3.3f and gives the following system of finite element equations.

$$C_t \frac{dT_t(A)}{dA} + UN_1(T_t^j - T_s^i) + UN_2(T_t^{j+1} - T_s^{i+1}) = 0 \tag{3.6a}$$

$$C_s \frac{dT_s(A)}{dA} - UN_1(T_t^j - T_s^i) - UN_2(T_t^{j+1} - T_s^{i+1}) = 0 \tag{3.6b}$$

Integrating Eq. 3.6 over A with N_1 as the weight function leads to:

$$\frac{C_t}{2}(T_t^{j+1} - T_t^j) + \frac{U\Delta A}{3}(T_t^j - T_s^i) + \frac{U\Delta A}{6}(T_t^{j+1} - T_s^{i+1}) = 0 \tag{3.7a}$$

$$\frac{C_s}{2}(T_s^{i+1} - T_s^i) - \frac{U\Delta A}{3}(T_t^j - T_s^i) - \frac{U\Delta A}{6}(T_t^{j+1} - T_s^{i+1}) = 0 \tag{3.7b}$$

Inlet temperatures ($T_s^1 = T_s^{in}$ and $T_t^1 = T_t^{in}$) and one outlet temperature ($T_s^{Nfe} = T_s^{out}$) are used as the boundary conditions for the discretized system. The overall heat transfer coefficient (U) and element area (ΔA) along with design variables are calculated using the algebraic equations.

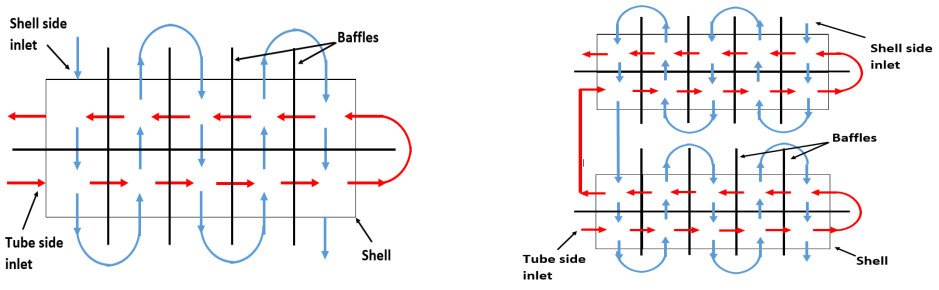
3.3.3 Algebraic Equations

The algebraic system consists of the same design equations used in LMTD-based method to calculate the design variables (Number of tubes, shell diameter) and heat transfer coefficients. The differences between the LMTD-based method and the DAE method described here are the replacement of the LMTD area equation and the equation for number of tubes (see Appendix Eq.2a and 2b) by the discretized equations (Eq.3.7) and an equation relating the variable ΔA to the number of tubes, respectively.

$$\Delta A = \frac{N_t \pi d_o L_t}{Nfe} \tag{3.8}$$

Note that the numerator is exactly equal to the heat exchange area of each shell and the number of finite elements is determined by Eq. 3.4. The number of tubes is treated as a continuous variable and can be related to the shell diameter using Eq. (2i) in the Appendix. The pressure drops in the discretized model are calculated using the same equations from Kern's method. The resultant heat exchanger design model is index-1 DAE system.

3.3.4 Multiple Shell



(a) Discrete Structure for 1-2 heat exchanger (b) Discrete Structure for 2-4 heat exchanger

Figure 3.3: Structures for single and multiple shell heat exchanger

The DAE model can be easily extended for multiple shell heat exchangers by connecting the individual discrete structures as shown in Figure 3.3. For the purpose of discretizing and indexing the nodes, the structure in Figure 3.3b is reformulated into Figure 3.4.

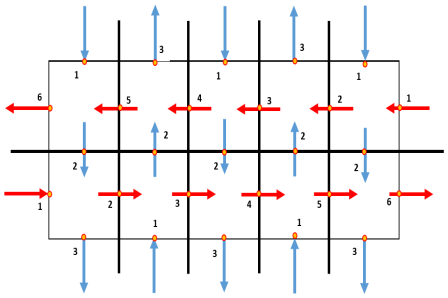


Figure 3.4: Reformulated discrete structure

The structures in both the figures represent the same heat exchanger but the latter structure is appropriate for indexing the nodes and discretizing the heat differential equation. The shell side flow is discretized into $N_b + 1$ individual streams flowing in the vertical direction separated by the baffles. The outlet node of each stream is connected to the inlet node of the next stream. Similarly the tube side flow is discretized horizontally into N_{tp} horizontal streams. The equations connecting the temperatures of shell side and tube side streams over the baffles and tube pass respectively are shown below:

$$T_s^{N_{tp}+1,k} - T_s^{1,k+1} = 0 \quad k = 1..N_b + 1 \tag{3.9a}$$

$$T_t^{N_b+2,m} - T_t^{1,m+1} = 0 \quad m = 1...N_{tp} - 1 \tag{3.9b}$$

3.3.5 Error Analysis

The accuracy of any numerical method is determined by comparing the numerical solution with the analytical solution. In general, the exact solution for a given DAE system is unknown and cannot be solved analytically as a function of design inputs. Here, first the algebraic design variables (ΔA and U) are fixed and then the analytical solution for a simpler problem are derived, the temperature profiles of the streams inside the exchanger.

$$C_t \frac{dT_t}{dA} + U(T_t - T_s) = 0 \tag{3.10a}$$

$$C_s \frac{dT_s}{dA} - U(T_t - T_s) = 0 \tag{3.10b}$$

Rearranging the equations

$$\frac{d(T_t - T_s)}{dA} + \left(\frac{U}{C_s} + \frac{U}{C_t}\right)(T_t - T_s) = 0 \tag{3.10c}$$

and substituting $\Delta T = T_t - T_s$ and $K = \left(\frac{U}{C_s} + \frac{U}{C_t}\right)$ leads to

$$\frac{d\Delta T}{dA} + K\Delta T = 0, \tag{3.10d}$$

Assuming constant properties and heat transfer coefficient, the ODE solution is given by

$$\Delta T(A) = \Delta T(0) exp(-KA) \tag{3.10e}$$

The temperature difference (ΔT) is evaluated at the nodes of discrete elements ($A = \Delta A$, see Figure 3.2b). As K is a given constant, the $exp(-K\Delta A)$ term is also a constant for fixed design. The individual stream temperatures inside the heat exchanger are determined by solving the linear system of equations.

$$T_t^{i+1} - T_s^{j+1} = (T_t^i - T_s^j)M \tag{3.11a}$$

$$C_t(T_t^{i+1} - T_t^i) + C_s(T_t^{i+1} - T_t^i) = 0 \qquad i, j = 1...Nfe \tag{3.11b}$$

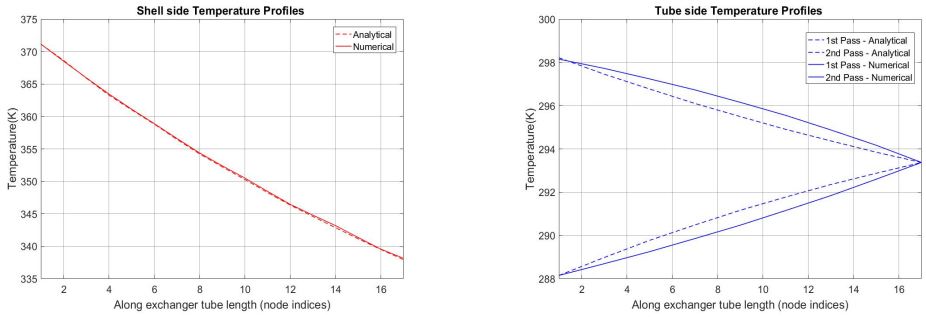
where $M = exp(-K\Delta A)$ and the second equation represents the energy balance inside each element. It can also be easily shown that Eq.3.10e is equivalent to the LMTD formula.

Given the inlet temperatures at the tube ($T_t^1 = T_t^{in}$) and shell ($T_s^1 = T_s^{in}$) side, the temperature profiles inside the exchanger for the tube and shell side fluid are compared with the numerical results obtained from the discretized DAE model with the lowest discretization level (For Example 1 $N_b = 7$ and $N_{tp} = 2$, $Nfe = (7 + 1) \times 2 = 16$).

Stream	Max. Error(K)	Max. Error(%)	Mean Error(K)	Mean Error(%)
Tube side	0.689	1.82	0.418	1.10
Shell side	0.320	3.20	0.145	1.45

Table 3.2: Comparison between numerical and analytical solutions

Example 1 (see data in Table 3.6) was used for this comparison and the results are presented in Table 3.2. The lower values of maximum and mean deviation between the temperatures calculated by the methods justifies the linear piecewise approximation as an accurate discretization method for the proposed DAE model.



(a) (b)

Figure 3.5: Temperature Profiles of Numerical and Analytical Solutions

Temperature profiles of the analytical and numerical solution are compared in Figure 3.5. Although the shell side temperature profile seems to have less error compared to the tube side, the relative error with respect to mass flow rates for both the sides is approximately equal. Further, the errors decrease with increasing discretization ($Nfe = (N_b + 1).N_{tp}.N_d$) and the results show that the numerical solution is a good approximation of the temperature variation inside the exchanger.

Next to show the method is consistent, as it converges asymptotically to the analytical solution as the discretization step size is decreased to zero. To check the consistency of the numerical method for the proposed DAE system, the problem is solved with increasing

levels of discretization or decreasing step sizes as shown in Figure 3.6. The shell side flow between two consecutive baffles is discretized further into individual streams with the N_d parameter depicting the number of discretizations.

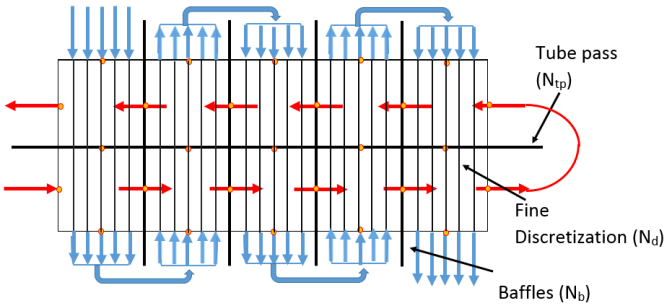


Figure 3.6: Fine Discretization of 1-2 heat exchanger

N_d	1	2	3	4	5	6
Area(m^2)	38.89	38.84	38.83	38.82	38.82	38.82
N_t	134	134	134	134	134	134
Nfe	16	32	48	64	80	96
No. of variables	70	110	150	190	230	270
No. of equations	70	110	150	190	230	270

N_d	1	2	3	4	5	6	...20
Area(m^2)	112.87	112.47	112.34	112.27	112.23	112.20	112.11
N_t	516	514	514	514	512	512	512
Nfe	40	80	120	160	200	240	400
No. of variables	147	238	329	420	511	602	1876
No. of equations	147	238	329	420	511	602	1876

Table 3.3: Results of DAE model with different discretizations

The heat exchanger area and number of tubes (N_t) required for Examples 1 and 3 with different levels of the discretization are presented in Table 3.3. The results show that the solution of the discretized DAE model asymptotically converges to a single design as the

discretization level is increased from $N_d = 1$ to $N_d = 6$, verifying the consistency of the discretization method. Moreover, the low level discretization ($N_d = 1$) seems to provide less than 0.2% and 0.6% error in Examples 1 and 3,respectively, of the final design with fewer variables and constraints, which makes it an ideal candidate for simultaneous design of heat exchanger network and exchanger design. Consistency of this stable numerical integration method implies that the method is convergent to the exact analytical solution.

3.4 Solution Strategy

A mixed strategy is developed to solve the overall heat exchanger design by using both the LMTD based method and the discretized DAE model.

3.4.1 Structural Design

The first step in the mixed strategy is to determine feasible topologies for the heat exchanger design. For this, use the LMTD-based model by solving the set of non-linear equations (Appendix Eq.2 and Eq.3) repeatedly using a Newton based solver with the set of values from Table 3.5. The number of shells is fixed and predetermined by ensuring the F_T correction factor to be greater than 0.75. The LMTD-based method is solved with different combinations of discrete values (for number of baffles, tube passes and fluid allocation) and tube dimensions (length and inlet/outlet diameter). To reduce the number of combinations to be solved, apply heuristics-based constraints such as $0.2D_s \leq L_b \leq D_s$, and velocity bounds to determine the range of values for the number of baffles. The number of exchanger combinations solved using LMTD-based method is $4 \times 3 \times 4 \times (UB - LB + 1) \times 2$(Fluid allocation) where UB and LB are the upper and lower range on the number of baffles, respectively. The solution of all the feasible designs are stored and the exchanger topologies are used in the next step to discretize the DAE model.

3.4.2 Area Design

The next step involves solving the DAE model for all the feasible solutions of the LMTD model by discretizing the DAE model using the feasible topologies obtained in the previous step. The discretized equations from the DAE model are solved simultaneously as a set of non-linear equations. This is executed for all exchanger combinations that have feasible LMTD solutions. The combination with the lowest objective cost is selected for the final optimization step, where the exchanger tube length is used as the degree of freedom variable with the variable bounds set as [0.5,2] times the nominal tube length found in the previous step. The fully discretized modeled is simultaneously optimized as an NLP using an interior point method solver IPOPT[5]. This final step is performed to determine the sensitivity of the exchanger design with respect to the tube length, as it is the easiest design specification which can be modified for design optimization purposes.
The optimization strategy with detailed steps is outlined below.

1. Given inlet and outlet temperatures, calculate number of shells using F_T correction.

2. Solve LMTD-based model (Appendix Eq.2 and 3) for every structural combination of heat exchanger geometry from Table 3.5 and obtain feasible solutions for exchanger structure.

3. Formulate and discretize the DAE model using the geometric structure of the feasible exchangers obtained in the previous step. Solve the discretized DAE system (Eq.3.7) along with the design equations (Eq.3.8, Appendix Eq.2b-j and 3) for all feasible solutions of the LMTD-based method.

4. The exchanger design with the lowest objective cost from the previous step is used to obtain the final design by simultaneously optimizing the discretized DAE model with the tube length as the degree of freedom.

The assumptions and advantages of using the proposed strategy are listed in Table 3.4.

	Advantages
1.	No integer variables are used, resulting in a robust overall strategy
2.	No custom initialization procedure or expensive pre-processing steps are required
3.	No LMTD and F_t calculations are required for solving the DAE model
4.	The DAE model can handle problems where stream properties vary significantly inside the exchanger
5.	The DAE model can be extended to phase changes and varying heat transfer coefficients through the use of complementarity constraints
6.	Temperature profiles inside the exchanger are determined simultaneously with the exchanger design. This aids in analyzing the engineering design and performance of the heat exchanger

Table 3.4: Advantages and assumptions associated with the proposed strategy

	Assumptions
1.	Shell and tube side fluid flows are unidirectional and turbulent
2.	Details such as baffle design, head type etc. are considered to have negligible effects on exchanger area and pressure drops
3.	Tube pitch is equal to 1.25 times d_{out}
4.	Number of tubes calculated is rounded to the nearest even number

The conventional LMTD-based method provides feasible topological solutions which are accurate enough for HENS problem and can be solved in a fraction of time as compared to MINLP models. Additionally, the shell-side hydraulic equations (e.g., from Bell-Delaware method) can also be implemented within the new framework by directly adding them to the DAE model as algebraic equations. In the next section, the DAE model is extended to design phase change heat exchangers like condensers and evaporators.

3.5 Phase Change

Phase Change is a physical phenomenon where a substance changes its physical state (solid, liquid and vapor) from one state to another. Phase change occurs when the temperature or pressure of the substance is either increased or decreased below a critical value. In process industry, streams are usually present in liquid or vapor state or as a mixture of liquid and vapor also known as two phase (2p) state. Flowsheets containing refrigeration cycle, cryogenic operations or heat distillation units have condensers and evaporators which are examples of phase change exchangers.

Modeling phase change is slightly more involved than without phase change heat transfer because the equations governing the phenomenon are non-smooth. For pure substances, phase change happens at a constant temperature and pressure known as saturation point. For liquid-vapor transition, this temperature is commonly known as boiling point. Although the temperature remains constant, the enthalpy of a pure substance varies as it undergoes phase change. Thus, the heat equation 3.3f can be transformed with enthalpy as the state dependent variable.

$$m_t \frac{dH_t}{dA} = -U(T_t - T_s) \tag{3.12a}$$

$$m_s \frac{dH_s}{dA} = U(T_t - T_s) \tag{3.12b}$$

The enthalpy of a stream undergoing phase change can be defined as a piecewise function of temperature, pressure and quality (liquid or vapor fraction).

$$H(T,P,x) = \begin{cases} H_{liq}(T,P) & T - T_b(P) \leq 0 \\ xH_{liq}(T,P) + (1-x)H_{vap}(T,P) & T - T_b(P) = 0 \\ H_{vap}(T,P) & T - T_b(P) \geq 0 \end{cases} \tag{3.13a}$$

here x represents the fraction of liquid in the two-phase mixture. T_b is the boiling point temperature of the stream. The piecewise formulation is similar to a Filippov dynamical system discussed in Section 2.6.3.

The enthalpy piecewise function can be similarly represented using complementarity constraints as follows.

$$H = xH_{liq} + (1-x)H_{vap} \tag{3.14a}$$

$$T - T_b = s_1 - s_2 \tag{3.14b}$$

$$0 \leq x \perp s_1 \geq 0 \tag{3.14c}$$

$$0 \leq 1 - x \perp s_2 \geq 0 \tag{3.14d}$$

The complementarity constraints can then be solved using the NLP reformulations discussed in the previous chapter. Another aspect of phase change heat transfer is the heat transfer coefficient which is illustrated in the next section.

3.5.1 Flooded Evaporator

Phase change exchangers can be of multiple types and one of the most common one is the flooded type exchanger. A flooded evaporator or condenser is an exchanger where part of the tubes are completely submerged inside the vaporizing or condensed liquid.

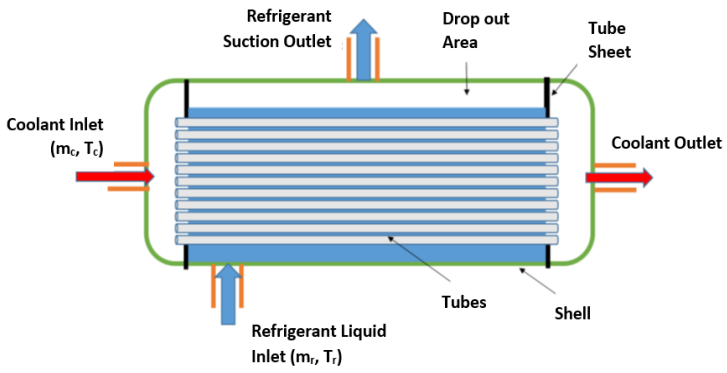


Figure 3.7: Flooded Evaporator

In the flooded evaporator shown in Figure 3.7, a two-phase stream enters the exchanger from the floor and vaporizes over the tubes by absorbing heat from the tubeside fluid. The vapors exit from the top of the exchanger as shown in the figure.

Note that a phase change exchanger has a simpler geometry without the baffles and tube-passes. Thus, the discretization of a flooded exchanger is slightly different and described as follows:

3.5.1.1 Tubeside

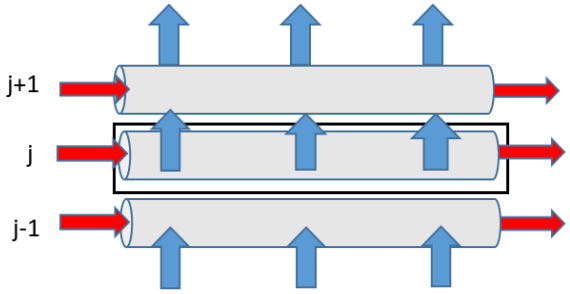


Figure 3.8: Flooded Exchanger Discretization

The tubeside fluid splits and flows inside the horizontal tubes exchanging heat with the shell fluid as shown in Figure 3.8. Since, the flow inside each tube is separated and independent, the heat equation for each tubeside stream can be represented as:

$$\dot{m}_t \frac{d(C_p T_t^j)}{dx} - k_t A^t \frac{d^2 T_t^j}{dx^2} + q^j(x) = 0 \tag{3.15a}$$

$$q^j(x) = PU(T_t^j - T_s^j) \tag{3.15b}$$

$$T_t^j(x = 0) = T_t^{in} \tag{3.15c}$$

$$\frac{dT_t^j}{dx}(x = L) = 0 \tag{3.15d}$$

$j = 1...N$ (N = no. of tubes), P-tube perimeter, $\dot{m}_t$- tubeside mass flow rate,
T_s - shellside fluid temperature and T_t - tubeside fluid temperature

The first and second equation are the second order heat equation and the heat exchange term for each tube respectively. The last two equations are initial and boundary conditions on the stream temperature. The differential equation is discretized using the FDM method discussed in previous chapter. The first order derivative term is discretized by centered

difference scheme and the second order term is discretized by the symmetric second order formula. The second order conduction term in Eq.3.15 is used to provide numerical stability and smoothness to the solution if the tube mass flow rate ($\dot{m}_t$) gets close to zero.

$$\dot{m}_t C_p \frac{(T_t^{i+1,j} - T_t^{i-1,j})}{2h_x} - k_t A^t \frac{(T_t^{i+1,j} - 2T_t^{i,j} + T_t^{i-1,j})}{(h_x)^2} + q^{i,j} = 0 \tag{3.16a}$$

$$q^{i,j} = PU^{i,j}(T_t^{i,j} - T_s^j) \tag{3.16b}$$

$$T_t^{1,j} = T_t^{in} \tag{3.16c}$$

$$\frac{3}{2}T_t^{M+1,j} - 2T_t^{M,j} + \frac{1}{2}T_t^{M-1,j} = 0 \tag{3.16d}$$

$i = 2....M$ (M-number of discretized tube elements)

3.5.1.2 Shellside

Since, the shellside fluid is undergoing phase change, the shellside heat equation is written with enthalpy as the state variable. It is assumed that the variation for shellside fluid in the horizontal direction is negligible and the enthalpy is only varying in vertical direction. With these assumptions, the heat equation in shellside can be represented as:

$$\dot{m}_s \frac{dH_s}{dy} - Q = 0 \tag{3.17a}$$

$$T_s(y = 0) = T_s^{in} \tag{3.17b}$$

$$x_s(y = 0) = x_s^{in} \tag{3.17c}$$

The heat equation is a first order ODE in this case with the initial conditions for both temperature and liquid fraction respectively. The shellside heat equation is discretized using FDM as follows:

$$\dot{m}_s \frac{(H_s^{j+1} - H_s^{j-1})}{2h_y} - Q^j = 0 \tag{3.18a}$$

$$T_s^0 = T_s^{in} \tag{3.18b}$$

$$x_s^0 = x_s^{in} \tag{3.18c}$$

The heat exchange terms Q^j is the cumulative heat absorbed by the shellside fluid from an individual tube and is calculated as:

$$Q^j = \sum_{i=1}^{N} q^{i,j} \tag{3.18d}$$

Additionally, the phase change equations Eq. 3.14 are added to relate the enthalpy to the temperature and liquid fraction. The h_x and h_y are the step sizes and are related to the tube dimensions (length L and diameter d_{out}) as:

$$h_x = \frac{L}{M} \quad \text{and} \quad h_y = \pi d_{out} \tag{3.19}$$

In this case, the overall heat transfer coefficient varies (U) inside the exchangers since the phase change heat transfer coefficient (h_{2p}) itself depends on the local heat flux. Phase changer heat transfer coefficient or latent heat transfer coefficient is much higher than the sensible heat transfer coefficient.

$$\frac{1}{U} = P\Big(\frac{1}{h_{s,2p}\pi d_{out}} + \frac{log(\frac{d_out}{d_{in}})}{2\pi k_m} + \frac{1}{h_t \pi d_{in}}\Big), \qquad h_{s,2p} = f(q) \tag{3.20}$$

3.5.2 Flooded Condenser

The flooded condenser model is very similar to the flooded evaporator model with instead of vaporization, a vapor stream enters the exchanger and condenses over the tubes. The vapors enter from the top of the condenser and releases heat to the tubeside fluid before condensing and leaving as saturated or subcooled liquid.

The same equations from the evaporator model can be used to model the heat exchange inside the condenser with some minor adjustments. The main difference between the models is the heat transfer coefficient calculation. Unlike in the evaporator model, the condenser has both latent and sensible heat transfer. The vapors initially cool down to boiling point temperature using sensible heat and then condense using latent heat before again sub-cooling down using sensible heat of the liquid. Since the two heat transfer coefficients

differ by magnitudes of scale, it is modeled using the rectangular function $\Pi(H)$:

$$h_{sens} = f_1(Re, Pr), \quad h_{cond} = f_2(q) \tag{3.21a}$$

$$\frac{1}{U_{sens}} = P\Big(\frac{1}{h_{sens}\pi d_o} + \frac{log(\frac{d_o}{d_{in}})}{2\pi k_m} + \frac{1}{h_{tube}\pi d_{in}}\Big) \tag{3.21b}$$

$$\frac{1}{U_{cond}} = P\Big(\frac{1}{h_{cond}\pi d_o} + \frac{log(\frac{d_o}{d_{in}})}{2\pi k_m} + \frac{1}{h_{tube}\pi d_{in}}\Big) \tag{3.21c}$$

$$U = U_{sens} + \Pi(H_r)(U_{cond} - U_{sens}) \tag{3.21d}$$

where

$$\Pi(H_s) = \begin{cases} 1 & H_{liq} \leq H_s \leq H_{vap} \\ 0 & elsewhere \end{cases} \tag{3.21e}$$

The last equation can be reformulated with a smooth approximation as:

$$\Pi(H_s) = \frac{1}{2}\Big(\frac{H_s - H_{liq}}{((H_s - H_{liq})^2 + \delta)^{\frac{1}{2}}} - \frac{H_s - H_{vap}}{((H_s - H_{vap})^2 + \delta)^{\frac{1}{2}}}\Big) \tag{3.21f}$$

where δ is a small relaxation parameter.

In the next section, both the DAE models (with and without phase change) are used to solve some example design problems. The detailed design results are then compared to the results from the LMTD method (only without phase change) and the temperature plots inside the exchanger are presented to show the variation of temperature and heat exchange inside the exchanger.

3.6 Examples

The heat exchanger design example problems are classified into three categories: a) single shell exchanger, b) multiple shell exchanger and c) phase change exchanger. The problems are formulated in the python-based modeling language Pyomo [35] and solved using Newton based NLP solver IPOPT 3.12 [5].

3.6.1 Single shell Exchanger

In these examples, based on the inlet-outlet temperature data provided in Table and F_T calculation a single shell exchanger is required for the required heat exchange. Steam velocities and pressure drop on both tube and shellside are bounded by design heuristic values given by the problem.

$$1 \leq v_t \leq 2.5, \qquad 0.3 \leq v_s \leq 1.25 \qquad v_s, v_t \ \ in \ \ m/s \tag{3.22a}$$

$$\Delta P \leq \Delta P_{design} \qquad \Delta P \ \ in \ \ kPa \tag{3.22b}$$

Number of shells, baffles, tube passes have integer values and their ranges are specified by TEMA standards. Tube dimensions(diameter, thickness and length) also have standardized values and, for the purpose of this study, the values given in Table 3.5 are used.

Tube length(ft)	8,12,16 and 20
Tube Outside Diameter(in)	5/8, 3/4 and 1
Number of Tube passes	2,4,6 and 8
Number of shells	≥ 1 and ≤ 6

Table 3.5: TEMA Standard Values

The objective function for typical heat exchanger design problems is either minimizing the total exchanger area or the total annual cost consisting of area cost and pumping cost as shown in Eq.3.23. The objective to minimize in each example is chosen to be consistent with other literature works

$$Pumping \ \ Cost = c_{cost}\left(\frac{\Delta P_t m_t}{\rho_t} + \frac{\Delta P_s m_s}{\rho_s}\right), \qquad \Delta P \ \ in \ \ kPa \tag{3.23a}$$

$$Area \ \ Cost = a_{cost}(Area)^{b_{cost}}, \qquad Area \ \ in \ \ m^2 \tag{3.23b}$$

$$Min \qquad Total \ \ Annual \ \ Cost = Area \ \ Cost + Pumping \ \ Cost \tag{3.23c}$$

The flow rates and inlet-outlet temperature data for these examples are reported in Table 3.6 along with constant values of thermophysical properties for both the streams. Table 3.7

reports additional design details like tube thermal conductivity, dirt resistance factors and allowable pressure drops for each example.

	Stream	$T_{in}(K)$	$T_{out}(K)$	$\dot{m}(kg/s)$	$\mu(Pa.s)$	$\rho(kg/m^3)$	$C_p(kJ/kg.K)$	$k(W/m.K)$
Ex-1	Hot	371.15	338.15	14.9	2.3e-04	777	2.684	0.11
	Cold	288.15	298.15	31.58	1.0e-03	998	4.180	0.60
Ex-2	Hot	368.15	313.75	27.78	3.4e-04	750	2.840	0.19
	Cold	298.15	313.15	68.88	8.0e-04	995	4.200	0.59
Ex-3	Hot	483.15	377.59	19.15	1.2e-04	789.72	2.428	0.106
	Cold	324.81	355.37	75.22	2.9e-04	820.12	2.135	0.123

Table 3.6: Stream Flowrate Data

	Tube thermal conductivity	Side	$r_d(m^2K/W)$	$\Delta P_{design}(kPa)$	$(a_{cost}, b_{cost}, c_{cost})$
Ex-1	50	Tube	0.00015	42.0	(123, 0.59, 1310)
		Shell	0.00015	7.0	
Ex-2	50	Tube	0.00017	68.95	(1, 0, 0)
		Shell	0.00017	68.95	
Ex-3	45	Tube	0.00035	78.805	(123, 0.59, 1310)
		Shell	0.00035	83.631	

Table 3.7: Exchanger Design Data

In Table 3.8, the LMTD and DAE model solutions with the minimum objective value are reported for each example. Note that the same structural design provides the minimum objective value for both the LMTD and DAE models. Also observe that exchanger designs obtained by the LMTD-based method and the DAE method are approximately the same for all three examples and within numerical error (rounding, discretization etc.) tolerances. The reason for this can be explained using the assumptions of LMTD-based methods stated earlier in the paper (Table 3.1).

	Example 1		Example 2		Example 3	
	LMTD Solution	DAE Solution	LMTD Solution	DAE Solution	LMTD Solution	DAE Solution
Area(m^2)	38.28	38.83	230.3	229.8	111.14	112.46
Duty(kW)	1320	1320	4339	4339	4908.8	4908.8
$LMTD(K)$	60.77	N/A	30.78	N/A	84.83	N/A
F_t	0.985	N/A	0.823	N/A	0.9166	N/A
N_{tp}	2	2	8	8	6	6
$D_s(m)$	0.47	0.46	1.15	1.15	0.70	0.71
N_t	126	134	790	790	508	514
N_b	7	7	4	4	6	6
$d_o(mm)$	25.40	25.40	25.40	25.40	19.05	19.05
$d_i(mm)$	21.18	21.18	21.18	21.18	16.19	16.19
$p_t(mm)$	31.75	31.75	31.75	31.75	23.8	23.8
$L_t(m)$	3.658	3.658	3.658	3.658	3.658	3.658
$v_t(m/s)$	1.32	1.35	1.03	1.07	1.39	1.43
$v_s(m/s)$	0.44	0.43	0.41	0.41	1.25	1.24
$h_t(W/m^2.K)$	4430.4	4532.7	1951.1	2022.9	2880.8	2953.0
$h_s(W/m^2.K)$	906.6	903.6	2728.4	2729.8	1798.6	1794.2
$U(W/m^2.K)$	575.95	572.33	724.2	725.2	568.0	564.5
$\Delta P_t(kPa)$	15.23	16.16	26.85	29.43	45.92	49.08
$\Delta P_s(kPa)$	5.25	5.22	8.92	8.93	77.30	76.96
Hot fluid allocation	Shell	Shell	Tube	Tube	Tube	Tube

Table 3.8: Comparison of LMTD and DAE-based Model Results

It can be easily verified that the examples satisfy all the assumptions except the assumption of average isothermal temperature of shell fluid at every cross section. Even with the high number of tube passes in Examples 2 and 3 (8 and 6 respectively), the difference between the two solutions is negligible. The reason for this behavior is explained in Kern [32] (p. 140) by the high turbulence on the shell side, meaning that the fluid is com-

pletely mixed, thus reducing the temperature gradient along the vertical direction. They also show that the F_T factor for a 1-8 exchanger is only 2 percent different from that of the 1-2 exchanger in extreme cases. The difference in tube side pressure drops is due more to the multiplicative effect of the number of tube passes and less because of the difference in tube side velocities. The total annual cost of the exchanger design for Example 2 is equal to \$5157.52/yr and \$5279.28/yr for LMTD and DAE model respectively, which is close to the result (\$5250/yr) reported in Mizutani et al. [18].

Results for other examples are also in the same order of magnitude compared to literature results although the models are significantly different. For Example 1, Ravagnani and Caballero [19] and Shenoy [36] reported lower exchanger areas (28.40m^2 and 28.31m^2 respectively) but incorrectly reported higher LMTD value (88.60K) which were calculated using Chen [16] approximation. Onishi et al. [20] used higher allowable pressure drops ($\Delta P_t \leq 45kPa$ and $\Delta P_s \leq 10kPa$) to obtain lower exchanger area (28.31m^2). Costa† and Bagajewicz [24] reported lower exchanger area (23.64m^2) but their model is based on different correlations.

Results for Example 3 are reported in Ravagnani and Caballero [19] and Onishi et al. [20] with the exchanger areas equal to 148.56m^2 and 139.12m^2 respectively using a MINLP model of Bell-Delaware method. Overall, observe that both the LMTD and DAE model obtain results in a more computationally efficient way. Note that for all these examples, the enumeration method used in this study obtains corresponding similar results to previous literature results in terms of both areas and topologies, without the need of solving computationally expensive MINLP models.

The optimization results for the DAE model with variable tube length are reported in Table 3.9 along with the DAE solution when the tube length is fixed. Note that both results are based on discretized models with equal discretization level with low approximation error and do not require F_T correction factors. Observe that although the exchanger areas for both solutions are very close, the design variables such as shell diameter, number of

tubes and design performance (pressure drops) are significantly different.

	Example 1		Example 2		Example 3	
	DAE Solution		DAE Solution		DAE Solution	
	(Fixed L_t)	(Variable L_t)	(Fixed L_t)	(Variable L_t)	(Fixed L_t)	(Variable L_t)
Area(m^2)	38.83	38.23	229.8	234.4	112.46	110.93
Duty(kW)	1320	1320	4339	4339	4908.8	4908.8
N_{tp}	2	2	8	8	6	6
$D_s(m)$	0.46	0.52	1.15	1.18	0.70	0.66
N_t	134	158	790	842	514	428
N_b	7	7	4	4	6	6
$d_o(mm)$	25.40	25.40	25.40	25.40	19.05	19.05
$d_i(mm)$	21.18	21.18	21.18	21.18	16.19	16.19
$p_t(mm)$	31.75	31.75	31.75	31.75	23.8	23.8
$L_t(m)$	3.658	3.044	3.658	3.494	3.658	4.34
$v_t(m/s)$	1.35	1.14	1.07	1.0	1.43	1.72
$v_s(m/s)$	0.43	0.49	0.41	0.42	1.24	1.12
$h_t(W/m^2.K)$	4532.7	3962.3	2022.9	1919.3	2953.0	3423.5
$h_s(W/m^2.K)$	903.6	960.1	2729.8	2761.3	1794.2	1697.7
$U(W/m^2.K)$	572.33	581.35	725.2	710.8	564.5	572.3
$\Delta P_t(kPa)$	16.16	10.60	29.43	25.26	49.08	78.805
$\Delta P_s(kPa)$	5.22	7.0	8.93	9.55	76.96	58.66
Hot fluid allocation	Shell	Shell	Tube	Tube	Tube	Tube

Table 3.9: Comparison of DAE-based model with fixed and variable L_T

Note that in the optimization step, all design variables except the structural design are solved simultaneously. The total annual cost in Example 2 for the optimized DAE model is equal to \$5169.21/yr as compared to \$5279.28/yr for the fixed tube length DAE model. The variable tube length does not reach its bound but other variables such as shell pressure drop (Example 1), tube velocity (Example 2) and tube pressure drop (Example 3) reach

their bounds. Overall, the optimization of the DAE model by varying tube length results in better design and objective values, showing the potential for this model to be used to obtain optimal exchanger designs based on continuous variable optimization.

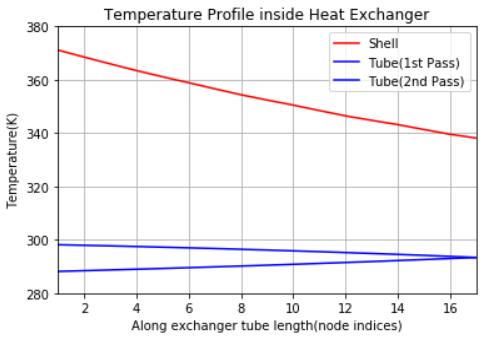
	Example 1		Example 2		Example 3	
	DAE Solution		DAE Solution		DAE Solution	
	1-1 exchanger	Multi-pass	1-1 exchanger	Multi-pass	1-1 exchanger	Multi-pass
Area(m^2)	38.73	38.83	227.65	229.8	110.9	112.46
Duty(kW)	1320	1320	4339	4339	4908.8	4908.8
N_{tp}	1	2	1	8	1	6
$D_s(m)$	0.48	0.46	0.71	1.15	0.51	0.71
N_t	266	134	936	790	456	514
N_b	5	7	7	4	17	6
$d_o(mm)$	19.05	25.40	15.875	25.40	15.875	19.05
$d_i(mm)$	16.19	21.18	13.24	21.18	13.24	16.19
$p_t(mm)$	23.8	31.75	19.84	31.75	19.84	23.8
$L_t(m)$	2.438	3.658	4.877	3.658	4.877	3.658
$v_t(m/s)$	0.60	1.35	0.54	1.07	1.46	1.43
$v_s(m/s)$	0.49	0.43	0.43	0.41	0.88	1.24
$h_t(W/m^2.K)$	2514.8	4532.7	2614.6	2022.9	2248	2953.0
$h_s(W/m^2.K)$	1102.1	903.6	1468.8	2729.8	1805.6	1794.2
$U(W/m^2.K)$	571.9	572.33	648.5	725.2	529.4	564.5
$\Delta P_t(kPa)$	1.69	16.16	2.5	29.43	12.91	49.08
$\Delta P_s(kPa)$	6.65	5.22	11.57	8.93	82.27	76.96
Hot fluid allocation	Shell	Shell	Shell	Tube	Shell	Tube

Table 3.10: Comparison of DAE solution for 1-1 design and multiple passes

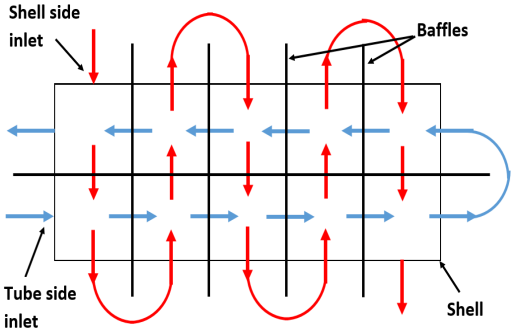
The DAE model can also be solved for 1-1 exchanger designs and the results are reported in Table 3.10 along with the DAE model results of multiple tube passes design. For 1-1 exchangers, Examples 1 and 2 can only be solved with the tube side velocity bounds

lowered to $0.5m/s$. Kern [32, p. 131] mentions that it is very difficult to obtain high velocity with 1-1 exchanger designs and these exchangers should be used only where large temperature crosses or high tube side flowrates. Smith [15] suggests using 1-1 exchanger design for smaller exchanger areas and high flow rates. The 1-1 exchanger design also results in lower tube side pressure drops and lower pumping costs as compared to other designs. Goncalves et al. [23] used a 1-1 exchanger design to solve Example 2 with a Bell-Delaware MILP model of total annual cost of \$3754/yr. In contrast, our DAE model for 1-1 exchanger design obtains a total annual cost of \$3812.24/yr for fixed tube length and decreases to \$3785.8/yr with variable tube length.

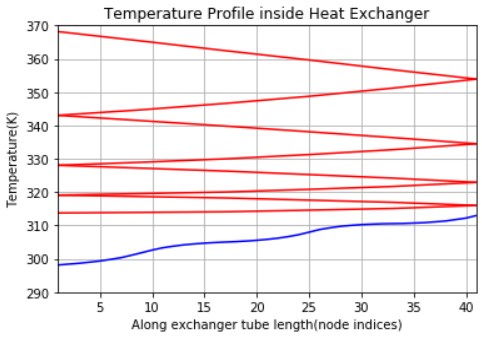
3.6.1.1 Temperature Plots



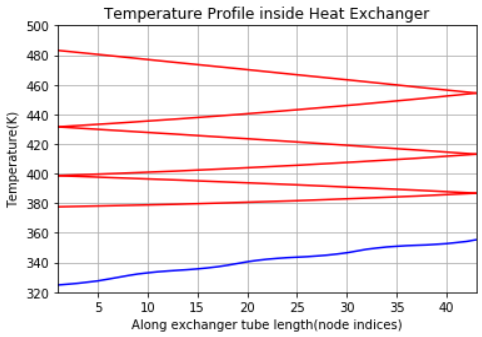
(a) Example 1 temperature profile



(b) Discrete Structure for 1-2 Exchanger



(c) Example 2 temperature profile



(d) Example 3 temperature profile

Figure 3.9: Temperature plots inside the heat exchanger designs

The temperature profiles for both the streams are plotted together for each of the examples in Figure 3.9. The red and blue lines represent the hot and cold stream temperatures respectively. Observe non-linear behavior of shell side temperature profile for example 2 and 3 with the former being more significant. This is due to the additional number of tube passes in example 2 than in example 3. Figures 3.9c and 3.9d show that the tube side temperature drop per pass decreases after each pass because, as the hot fluid inside the tubes is cooled the temperature difference between the hot and cold streams gets reduced, thus decreasing the heat exchanged inside the element.

3.6.1.2 Varying Thermo-physical Property

The next example consists of streams with temperature-dependent thermophysical properties. The purpose of this example is to show that the discrete method easily extends to this case, while LMTD method can not. The example data are created by modifying Example 2 with the thermodynamic properties (specific heat, density and thermal conductivity) declared as functions of stream temperature. For simplicity, linear polynomials are used to relate the physical properties to the stream temperatures .

$$C_p(T) = C_{p,ref} + A_{C_p}(T - T_{ref}) \tag{3.24a}$$

$$k(T) = k_{ref} + A_k(T - T_{ref}) \tag{3.24b}$$

$$\rho(T) = \frac{\rho_{ref}}{1 + A_\rho(T - T_{ref})} \tag{3.24c}$$

where F_{ref} ($C_{p,ref}, k_{ref}$ and ρ_{ref}) and T_{ref} are reference values for the thermodynamic properties and temperature respectively. In this example, T_{ref} is the average of the inlet and outlet temperatures and F_{ref} values are taken from Example 2 in Table 3.6. A_{C_p}, A_k and A_ρ values of the slope for hot and cold streams are tabulated below.

Stream	$A_{C_p}(kJ/kg.K^{-2})$	$A_\rho(K^{-1})$	$A_k(W/m.K^{-2})$
Hot	8.4e-03	1.6e-03	-2.4e-04
Cold	9.7e-03	8.4e-04	1.3e-03

Table 3.11: Slope Values for the thermodynamic properties

Unlike the LMTD-based methods, the DAE model does not need to assume constant values for thermodynamic properties. As a result, the heat transfer coefficients, which depend on stream thermodynamic properties, and velocities vary locally inside the heat exchanger in the new method. Due to this, the DAE method is able to determine the exchanger area more accurately as compared to the LMTD-based method which assumes constant properties inside the exchanger.

	LMTD Solution	DAE Model (Constant Properties)	DAE Model (Variable Properties)
Total Cost($/yr)	5157.25	5279.28	5873.75
Area Cost($/yr)	3045.50	3041.44	2597.75
Pumping Cost($/yr)	2111.75	2237.84	3276
Area(m^2)	230.3	229.8	175.9
Duty(kW)	4339	4339	4339
$LMTD(K)$	30.78	N/A	N/A
F_t	0.823	N/A	N/A
N_{tp}	8	8	8
$D_s(m)$	1.15	1.15	1.04
N_t	790	790	602
N_b	4	4	4
$d_o(mm)$	25.4	25.4	25.4
$d_i(mm)$	21.18	21.18	21.18
$p_t(mm)$	31.75	31.75	31.75
$L_t(m)$	3.658	3.658	3.658
$\Delta P_t(kPa)$	26.85	29.43	49.05
$\Delta P_s(kPa)$	8.92	8.93	9.88
Hot fluid allocation	Tube	Tube	Tube

Table 3.12: Detailed exchanger designs for Example with varying thermodynamics

Results from both approaches are compared in Table 3.12. The exchanger design obtained by both methods differs drastically, with the constant property models, obtaining

an exchanger area approximately 30 $\%$ more than the exchanger area obtained from the variable property DAE model. The number of tubes and pressure drops are also significantly different, with the variable property model design using fewer tubes and predicting higher pressure drops. The overall cost of the design using variable properties is approximately 11% higher than the cost for design with constant properties.

The DAE model with variable properties is simulated with the exchanger design in the second column (constant properties) of Table 3.12. Observe that the outlet hot and cold temperatures deviate from their values in Table 3.6. The outlet hot stream temperature reduces from $313.75K$ to $312.15K$, while the cold stream outlet temperature reduces from $313.15K$ to $312.45K$. This leads to a temperature cross, which shows that the design in the second column is infeasible. This example clearly shows that the LMTD method's assumption of constant values for physical properties (i.e. with F_{ref}, ρ_{ref} and T_{ref}) can lead to erroneous or infeasible solutions when stream properties are temperature dependent.

3.6.2 Multiple shell Exchanger

This example is extracted from Short et al. [37] of a multiple shell heat exchanger design. This example shows that the DAE method easily extends to multiple shells, while the LMTD method may obtain inaccurate design. The objective function here is to minimize the total annual cost comprising of area cost and pumping cost. The cost parameters, the temperature and flow rate data are provided in Table 3.13. The tube thermal conductivity and the dirt resistance factors are 50 Wm/K and $1.7 \times 10^{-4} m^2K/W$, respectively.

Stream	$T_{in}(K)$	$T_{out}(K)$	$\dot{m}(kg/s)$	$\mu(Pa.s)$	$\rho(kg/m^3)$	$C_p(kJ/kg.K)$	$k(W/m.K)$
Hot	533	418.9	28.5	2.4e-04	634	2.454	0.114
Cold	408.9	431.7	143	2.4e-04	634	2.454	0.114

a_{cost}=60, b_{cost}=0.6, c_{cost}=700

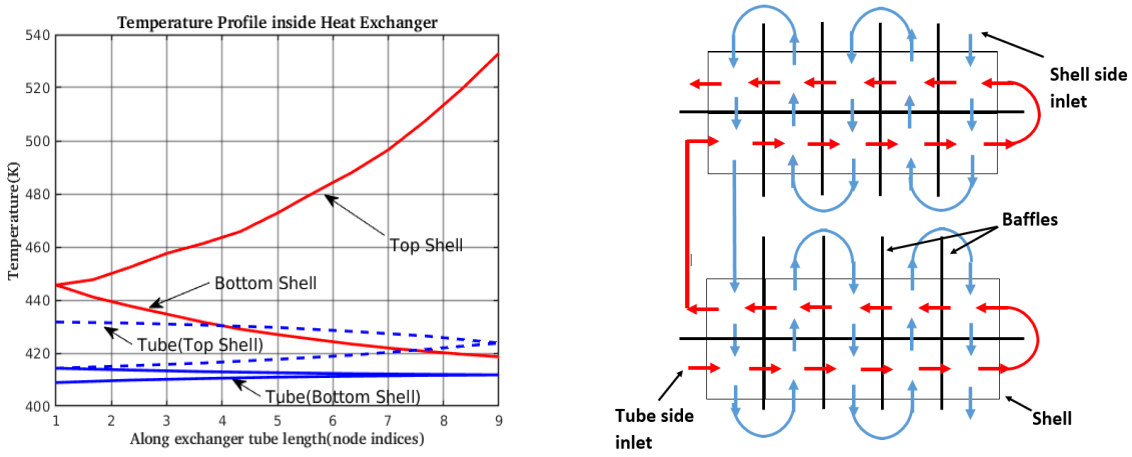
Table 3.13: Multiple shell exchanger Example Data

The F_T correction factor based on the inlet-outlet temperatures require at least two shells to be more than the critical value of 0.75. The detailed design from both the LMTD-based method and the DAE model are presented in Table 3.14.

	Short et al. (2016)	LMTD Solution	DAE Model (fixed L_t)	DAE Model (variable L_t)
Area(m^2)	426.5	521.6	569.5	581.4
Duty(kW)	7978	7980	7980	7980
$LMTD(K)$	39.44	39.44	N/A	N/A
F_t	0.92	0.92	N/A	N/A
N_s	2	2	2	2
N_{tp}	12	4	4	4
$D_s(m)$	0.94	1.07	1.17	1.25
N_t	548	1588	1952	2276
N_b	2	3	3	3
$d_o(mm)$	25.40	19.05	19.05	19.05
$d_i(mm)$	21.18	16.19	16.19	16.19
$p_t(mm)$	31.75	23.81	23.81	23.81
$L_t(m)$	4.88	2.438	2.438	2.135
$v_t(m/s)$	-	1.23	1.17	1
$v_s(m/s)$	-	0.33	0.32	0.34
$h_t(W/m^2.K)$	2324.3	1660.9	1602.2	1416.7
$h_s(W/m^2.K)$	1114.5	775.8	759.6	787.7
$U(W/m^2.K)$	550.3	421.7	409.3	400.9
$\Delta P_t(kPa)$	72.6	15.65	14.37	9.94
$\Delta P_s(kPa)$	12.1	7.53	7.24	8.84
Hot fluid allocation	Tube	Shell	Shell	Shell
Area Cost($/yr)	2995.85	3380.38	3563.17	3607.66
Pumping Cost($/yr)	4193.35	2707.87	2496.65	1847.56
Total Cost($/yr)	7189.20	6088.25	6059.82	5455.22

Table 3.14: Detailed exchanger designs for Multi-shell Exchanger Example

The new method proposes higher heat exchanger area with more tubes but it also provides lower costs and lower pressure drops on both tube and shell sides. This design allocates the hot fluid flowing in the shell side, as compared to the tube side in the initial literature study, and therefore provides lower total costs of $\$6088.25, \$6059.82, \$5455.22$ for the LMTD, DAE model (fixed L_t), and DAE model (variable L_t) respectively, as compared to $\$7189.20$ in Short et al. [37]. The optimal design also includes the tube side velocity reaching its lower bound at 1 m/s as pressure drop contributes significantly to the overall costs.



(a) (b)

Figure 3.10: Temperature plot inside the optimal heat exchanger design

The differences between the LMTD and DAE model solution are significant with the exchanger areas differing by more than 9%. The reason for this difference in results is an assumption used in the derivation of the F_T correction factor, which uses an average shell side fluid temperature along each cross-section to determine the formula. This assumption is based on high flow rate mixing the shell-side fluid between two baffles uniformly. While this assumption holds true for most cases with one shell exchangers, it is not true for multiple shell exchangers where the F_T correction factor is derived by combining the

F_T factors for individual shells. The assumption that the shell side temperature is an average isothermal temperature at any cross-section is certainly not true when the temperature difference of shell side fluid between two shells is large, as it is the case in this example and with many multi-shell arrangements. Kern [32] explains this in Page 140 and Page 176 of his seminal textbook, suggesting that "this may lead to an error of 10-15 per cent when large temperature differences exist between the average temperatures of the shell fluid in the two shell passes".

3.6.3 Phase Change Exchanger

A water-cooled chiller which is a single-stage refrigeration cycle is used to demonstrate the phase changer exchanger models as shown in Figure 3.11

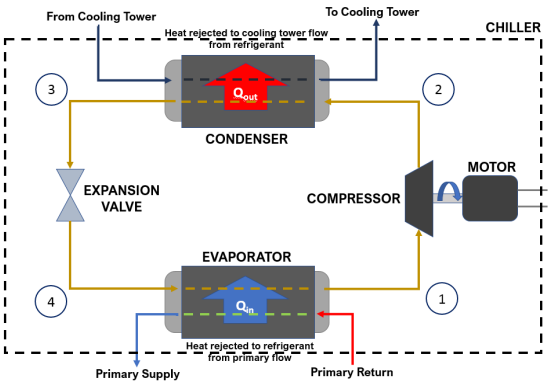


Figure 3.11: Single-stage Refrigeration Cycle

The flood condenser and flooded evaporator DAE models are used to design the phase change exchangers in the cycle. The refrigerant R-134a (1,1,1,2-Tetrafluoroethane) is used as the circulating fluid in the system which undergoes phase change in both the exchangers. The compressor and expansion valve are modeled using the following equations:

$$T_{out} = T_{in}\left(\frac{P_{out}}{P_{in}}\right)^{\frac{\gamma-1}{\gamma}}, \quad \gamma = 1.33 \tag{3.25a}$$

$$\Delta H_{valve} = 0 \tag{3.25b}$$

The parameters and operating conditions associated with the system are listed in Table 3.15. The heat transfer coefficient is calculated using a power function ($h_{ref} = 1479(q/P)^{0.3274}$). The complementarities in the DAE model are reformulated using Reg(ϵ) and the NLP is solved iteratively with decreasing value of ϵ until $\epsilon = 10^{-5}$ when the penalty term is added with $\rho = 10$ and solved simultaneously.

Parameter	Value
Refrigerant flow rate	1.8kg/s
Chilled water flow rate	23.75kg/s
Cooling water flow rate	44kg/s
Condenser Pressure	9bar
Evaporator Pressure	3.5bar

Parameter	Value
Condenser Sat. temp.	35°C
Evaporator Sat. temp.	5°C
Chilled water in temp.	12°C
Cooling water in temp.	27°C
Cooling water out temp.	32°C

Table 3.15: Operating Conditions for Refrigeration Cycle

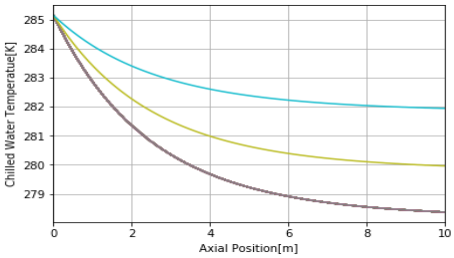
The design parameters used in the DAE models for flooded evaporator and condenser are listed below in Table 3.16. The tubeside heat equation in the DAE model is discretized with $M = 100$ finite elements.

Evaporator Parameter	Value
No. of Tubes(N)	50
Length of Tubes(L)	10m
Tube Diameter(d_{in})	0.0254m (1″)
Tube Thickness(t_d)	1mm
Tube Conductivity(k_m)	400 W/m K

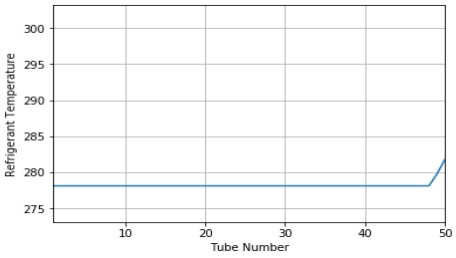
Condenser Parameter	Value
No. of Tubes(N)	40
Length of Tubes(L)	10m
Tube Diameter(d_{in})	0.0254m (1″)
Tube Thickness(t_d)	1mm
Tube Conductivity(k_m)	400 W/m K

Table 3.16: Exchanger Characteristics

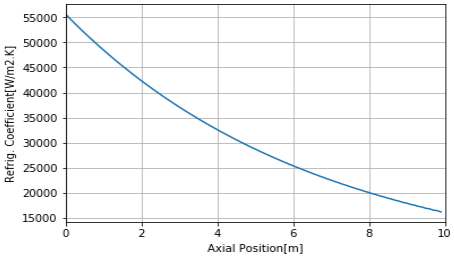
The example is modeled using Pyomo and solved as a simulation problem with a dummy objective function using IPOPT as the NLP solver. The solution of each NLP is used as the initial point for the subsequent NLP with lower epsilon value. This aids in stabilizing the convergence of the NLP to the solution of Reg(0).



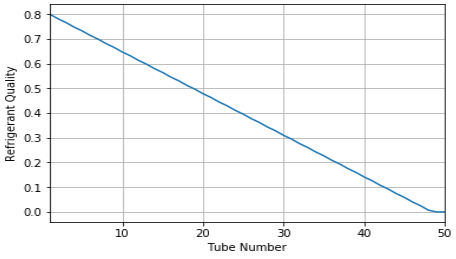
(a) Chilled Water Temperature(K)



(b) Refrigerant Temperature(K)

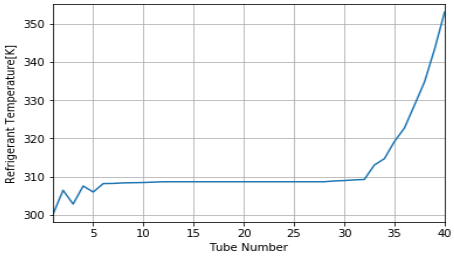


(c) Refrigerant H.T. Coefficient(W/m^2K)

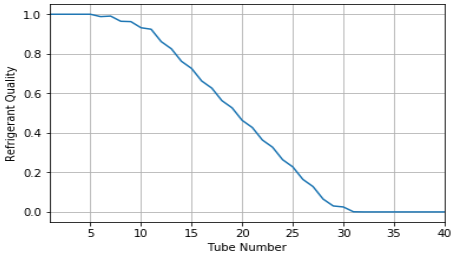


(d) Refrigerant Quality

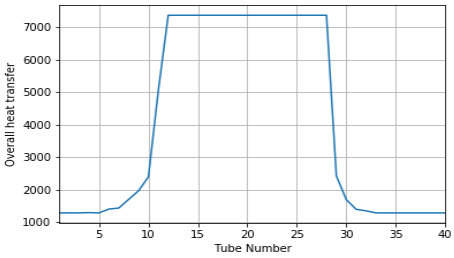
Figure 3.12: Evaporator Results



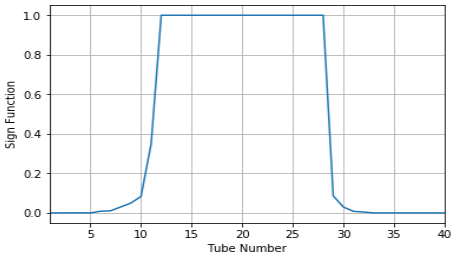
(a) Refrigerant Temperature(K)



(b) Refrigerant Quality



(c) Overall H.T. Coefficient(W/m^2K)



(d) Rectangular function

Figure 3.13: Condenser Results

The results from the simulation run are presented as plots of temperature, quality (liquid fraction) and heat transfer coefficient variation inside the condenser and evaporator in Figure 3.12 and 3.13.

For the given operating conditions, the refrigerant temperature and quality plots in Figure 3.12b-d show that it remains in two-phase region and gets completely vaporized on 48^{th} tube and exits with little superheat. Figure 3.12a shows the temperature profile of $N = 50$ tubes inside which chilled water is getting cooled down. The Figure 3.12c depicts the decreasing variation of refrigerant side heat transfer coefficient in the horizontal direction of the tubes.

The refrigerant temperature and quality profiles in Figure 3.13a-b shows that it enters the condenser as a superheated vapor and condenses between the 5^{th} and 32^{th} tubes. After that, it gets further cooled and leaves as a pure subcooled liquid. The overall heat transfer coefficient profile in Figure 3.13c clearly capture the variation and difference between the latent heat transfer and sensible heat transfer inside the condenser.

The performance of the refrigeration cycle is measured as the ratio of cooling duty and compressor power ($COP := \frac{Q_{cool}}{\dot{W}_{comp}}$). The COP for the given system is equal to $COP = 2.83$ with the cooling utility equal to $Q_{cool} = 283$kW which is approximately equal to 73 refrigeration tons. Although a single operating point is used here to test the phase changer exchanger models, the DAE models can be used to determine a family of heat exchanger designs for different operating conditions in the refrigeration cycle.

3.7 Conclusions

This chapter introduces the heat exchanger design problem from a mathematical programming and optimization perspective. The LMTD-based methods are described along with the assumptions used in their derivation. A new DAE exchanger model is introduced which is solved in tandem with the LMTD-based method using an iterative strategy. The conventional LMTD-based method provides feasible topological solutions (discrete deci-

sion variables) and can be solved in a fraction of time as compared to MINLP models. The DAE model is then used to accurately calculate the area of the exchanger, number of tubes and optimal tube length.

In first three examples, the assumptions made by the LMTD-method are satisfied, as a result the LMTD and DAE model results are same or equal to each other. But in the example with temperature dependent physical properties, the DAE model provides more accurate results. The example with multiple shell heat exchanger, the DAE and LMTD results differ more than 9% and is an improvement compared to the literature result. The DAE model is also extended to solve phase change heat exchangers which is not possible with standard LMTD methods. The results are consistent with experimental observations which shows that the model is accurate and consistent.

In conclusion, the DAE model is more accurate and generalized than the LMTD based models. The DAE model does not restrict the type of equations for heat transfer coefficients and different smooth correlations (eg. HTRI) can also be used with it. Also, the DAE model solves within few CPU seconds without any custom initialization unlike the MINLP models. This makes the DAE model ideal for applications like heat exchanger network synthesis (HENS) which requires solving design for exchangers with multiple shells in the optimal network. The DAE model is also applicable in work and heat exchanger network synthesis (WHENS) applications which consists of phase change exchangers and compressors.

Chapter 4

Heat Exchanger Network Synthesis

This chapter presents the classical heat exchanger network synthesis (HENS) problem with the extension of detailed exchanger design. Several different types of optimization models for HENS are presented and discussed in detail. Two strategies are introduced to incorporate the detailed exchanger DAE models (presented in Chapter 3) inside the HENS problem.

The two-step hybrid strategy using correction factors is described using the stagewise superstructure (SWS) MINLP model for HENS and a network level NLP model for stream splitting and bypass. Then, an alternate strategy based on trust region filter and integer cuts is introduced to solve the simultaneous HENS with detailed exchanger design. Both strategies are used to solve specific examples from literature and the results are compared with discussion on advantages and disadvantages of the two methods.

4.1 Introduction

Heat exchanger network synthesis (HENS) has been a classical problem in the process engineering community for more than 75 years (Broeck [38]) with more than 500 published articles and conference papers on the topic. The HENS problem is very relevant in process engineering as it solves the trade-off between operating costs (external utilities and pumping costs) and capital investment costs (number of shells and areas of heat exchangers). Many methods and algorithms have been developed to solve the network synthesis problem. However, few HENS studies have attempted to include detailed designs of the individual heat exchangers inside the network synthesis step.

The HENS problem was first formally defined by Masso and Rudd [39] and can generally be defined as follows

Given:

- a set of hot and cold streams with known inlet temperatures, target outlet temperatures and thermophysical properties
- capital cost parameters for exchangers
- cost of utilities (heating, cooling and pumping)

Obtain a heat exchanger network with the minimum total annualized cost (TAC) which is the sum of total annualized capital cost and the utility cost of the exchanger network. The problem is known to be NP-hard [40] and it gets difficult to solve large-scale HENS problems with limited computational resources.

HENS solution methods can be characterized into two classes: Sequential methods and Simultaneous methods. In the sequential method, the HENS problem is decomposed into subproblems with different objectives or targets like minimum utility, minimum number of matches and minimum exchanger area, which are solved sequentially in the respective order. Then the detailed network designed is done such that the network design is as close to the original solution as possible. Heuristic method like pinch technology (PT) introduced by Linhoff and his collaborators [41, 42] has been vastly used to solve the individual problems using pinch analysis. They use temperature intervals and locate thermodynamic bottlenecks known as pinch points to minimize the utility cost of the network. Alternatively, mathematical programming models have also been used to solve the individual problems such as min. utility (LP), min. match (MILP, [43, 44]) and min. area [45]. These methods scale well for larger problems but they need not obtain the best exchanger network with minimum TAC as they don't incorporate the trade-off between capital and utility costs.

On the other hand, simultaneous method solves the HENS by directly considering the capital and utility costs in the objective as a single optimization problem. The most common representation of the simultaneous, mathematical programming network synthesis

problem is the stage-wise superstructure (SWS) formulation given by Yee and Grossmann [46], which is solved as a nonconvex MINLP. Since then, many variants of the SWS model have been developed to incorporate more characteristics of the HEN. Some papers [47, 48] have also proposed MINLP model(s) without stages and temperature intervals as an alternative simultaneous HENS model.

4.1.1 HENS with Detailed Exchanger Design

Furman and Sahinidis [49] provided a detailed review on over 450 research papers related to HENS and its models. In the review, only three references ([50], [51] and [52]) discussed the incorporation of detailed heat exchanger design inside the HENS model. Liporace et al. [50] showed the need to account for design details and performance indicators like heat transfer coefficient to avoid networks with infeasible exchanger units. Polley and Panjeh Shahi [52] developed correlations from [32] and Dittus-Boelter equations to relate pressure drops, heat exchanger areas, and heat transfer coefficients. Subsequently, Frausto Hernandez et al. [53] used these equations within the SWS model to show the effect of including exchanger design variables in the network synthesis. Sun et al. [54] used an iterative strategy which included the number of tubes and shells in the HENS model and subsequently uses LMTD method to perform detailed exchanger design and updates the heat transfer coefficients in the MINLP. However,the additional nonconvex constraints required in the MINLP make this method impractical, especially for large problems.

Mizutani et al. [18] and Ravagnani and Caballero [19] developed MINLP models for individual exchanger design and incorporated them inside the HENS model in a bilevel iterative strategy. Mizutani et al. [18] used Bell-Delaware method for heat transfer correlations and integer variables for tube dimensions and number of baffles etc. to formulate a non-convex MINLP model. Ravagnani and Caballero [19] modified the MINLP model with TEMA standards for number of tubes and additional pressure drop and fouling limit constraints. Mizutani et al. [55] used logic-based outer approximation method [56] to include the design details as corrections in the SWS model. Ravagnani and Caballero [57]

updates the individual stream heat transfer coefficients in the SWS model based on the solution of the detailed exchanger design. Although, they show in their solutions that the optimal network topology depends on the design of the individual exchangers in it, the lack of complete exchanger design details in the HENS model and the nonconvex nature of their design models made it difficult to converge or obtain global minimum. Xiao et al. [34] developed a similar method but included detailed exchanger design for utility exchangers with phase change using a LMTD method and solved the MINLP models using heuristic methods: simulated annealing and genetic algorithms.

The previous studies only used single shell exchangers, thus limiting the use of their algorithms on general cases with multiple shell exchangers. Short et al. [37] developed a two-step hybrid strategy using correction factors for number of shells, heat transfer coefficients and pressure drops in the SWS MINLP model. This method could include multi-shell exchangers and did not require solving nonconvex MINLP for exchanger design. They used a similar approach to solve the multi-period HENS problem [58] where the detailed exchanger had to be feasible in multiple operational periods. The major drawback of their strategy was the iterative solving of the exchanger design done manually after the HENS MINLP model was solved. This made the approach less practical and difficult to converge for larger networks with more number of hot and cold streams.

In this chapter, the two-step hybrid strategy is combined with DAE models for exchanger design developed in the previous chapter. A network design model with stream splitting and stream bypasses is presented which is used together with the SWS MINLP model to obtain optimal flows and temperature of the streams. In the second part of the chapter, a trust region based framework is developed to incorporate the detailed exchanger design DAE models using surrogate models and integer cuts in the network synthesis model. The trust region based strategy is able to provide a measure of optimality and feasibility to the solution of the overall HENS problem. Both approaches are used to solve example problems from the literature and their results are discussed.

4.2 HENS model

In this thesis, the optimal exchanger network is obtained by solving the SWS MINLP model with isothermal mixing. Then the integer solution is used to formulate a network level NLP with non-isothermal mixing, stream splitting and bypass constraints. The NLP solution provides the optimal stream flowrates and inlet-outlet temperatures across the exchangers in the network.

4.2.1 SWS MINLP model

The SWS MINLP model introduced by Yee and Grossmann [46] has $N + 2$ stages where $N = max(N_h, N_c)$, and N_h, N_c denote the number of hot and cold streams respectively. The first and last stages are used to represent feasible connections between process streams and utility streams (Figure 4.1). Each possible assignment between one hot and one cold stream (including utility streams) is represented by a binary variable ($z_{i,j,k}$). The formulation only considers single hot and cold utility, although it can be easily extended for multiple utilities.

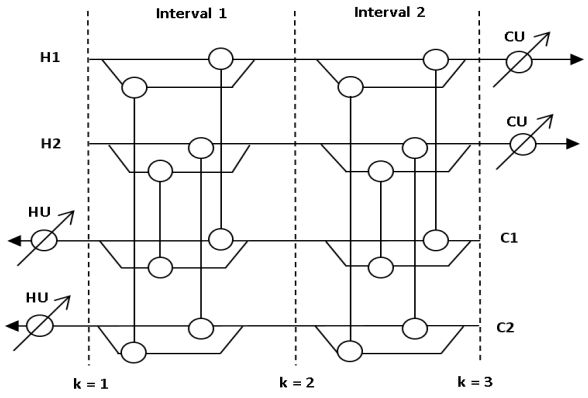


Figure 4.1: Stagewise Superstructure representation of HENS model

Energy balances for individual streams are written at the stage boundaries with the assumption of isothermal mixing of split streams. Heat exchanger areas are calculated using the nonlinear LMTD formula with constant values for overall heat transfer coefficient U.

Although the inverse of the LMTD is a convex function [59], it is non-differentiable ($\alpha = 1$) which makes it unsuitable for derivative-based KKT solvers.

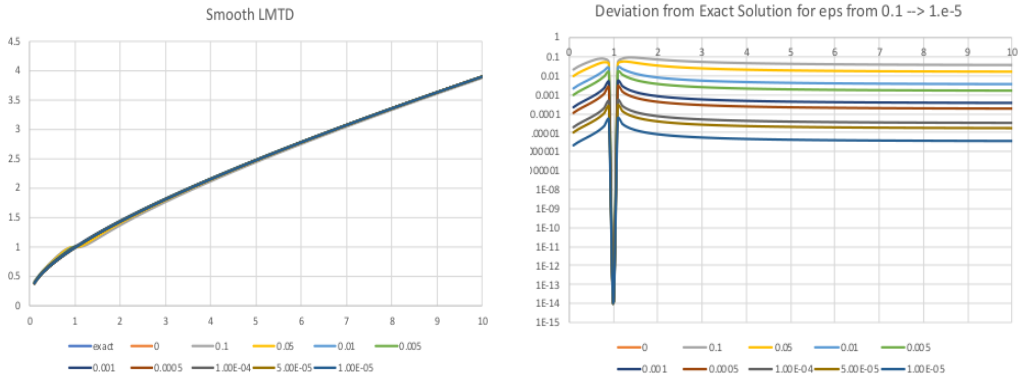
$$LMTD_{i,j,k} = (dt_{i,j,k+1} - dt_{i,j,k})/ln(dt_{i,j,k+1}/dt_{i,j,k}) = dt_{i,j,k}(\alpha - 1)/ln(\alpha) \tag{4.1}$$

where $LMTD_{i,j,k}$ is the LMTD for an exchanger between hot stream i and cold stream j in superstructure interval k, $dt_{i,j,k}$ and $dt_{i,j,k+1}$ are temperature differences between streams at either end of the exchangers (Appendix), and $(\alpha = dt_{i,j,k+1}/dt_{i,j,k})$. The LMTD equation (Eq.4.1) can be rewritten as $LMTD_{i,j,k} = dt_{i,j,k}|(\alpha - 1)|/|ln(\alpha)|$.

The absolute value function ($|x|$) is non-smooth but can be approximated as $|x| = \sqrt{x^2 + \epsilon}$, where ϵ is a small positive number. Thus the LMTD equation can be approximated as:

$$LMTD_{i,j,k} = dt_{i,j,k}((\alpha - 1)^2 + \epsilon)^{1/2}/(ln(\alpha)^2 + \epsilon)^{1/2} \tag{4.2}$$

Eq. 4.2 has bounded derivatives, is numerically stable and has improved accuracy compared to other approximations (Chen [16] and Patterson [17]).



(a) (b)

Figure 4.2: Performance of the LMTD Smooth approximation

Huang et al. [47] also used a similar approximation but couldn't converge it their MINLP model at the time. Since then, the improvements in global solver BARON (Ver. - 20.4.14) has made it possible to use it in SWS MINLP model. The heat balance, monotonicity and approach temperature constraints are detailed in the Appendix section (Eq.4-13).

4.2.1.1 Objective function

The objective function consists of minimizing total annualized cost (TAC) which is calculated by the following expression:

Total Annualized Cost (TAC) = Exchanger Area Cost + Pumping Cost + Utility Cost

where Exchanger Area Cost = Fixed Cost $\times$ Number of Shells + Variable Cost $\times$ Area^{AE}

Pumping Cost = Pumping per kPa cost $\times$ Pressure Drops(kPa)

Utility Cost = Steam Cost $\times$ Hot Utility + Cooling Water Cost $\times$ Cold Utility

The full expression for the objective is shown in Equation 4.3

$$\begin{aligned}\min \quad \Bigg[& \sum_{i\in H} CUC qc_i + \sum_{j\in C} HUC qh_j + CF\Big(\sum_{i\in H}\sum_{j\in C}\sum_{k\in K} z_{ij,k} NSP_{i,j,k} + \sum_{i\in H} zcu_i + \sum_{j\in C} zhu_j\Big) \\ & + \sum_{i\in H}\sum_{j\in C}\sum_{k\in K} NSP_{i,j,k} AC\Big(\frac{q_{i,j,k}}{NSP_{i,j,k}(U_{i,j,k})(LMTD_{i,j,k})}\Big)^{AE} + \sum_{i\in H} AC\Big(\frac{qc_i}{(U_i)(LMTD_i)}\Big)^{AE} + \\ & \sum_{j\in C} AC\Big(\frac{qh_j}{(U_j)(LMTD_j)}\Big)^{AE} + \sum_{i\in H}\sum_{j\in C}\sum_{k\in K} Pcost\left(\Delta P^c_{i,j,k} + \Delta P^h_{i,j,k}\right)\Bigg]\end{aligned} \tag{4.3}$$

where CUC and HUC are the costs of the cold and hot utilities respectively, $q_{i,j,k}$ is the energy transferred between hot process stream i and cold process stream j in interval k, qh_j and qc_i is the energy transferred from hot utility to cold stream j and from cold utility to hot stream i, respectively. $z_{i,j,k}$ is the binary variable representing a process stream match between hot process stream i and cold process stream j in interval k. CF is the fixed cost associated with an exchanger, AC is a variable cost factor based on the area, AE is the size exponent, and $Pcost$ is the cost associated with pumping. The number of shell passes, $NSP_{i,j,k}$ and the hot & cold pressure drops ($\Delta P^c_{i,j,k}$ and $\Delta P^h_{i,j,k}$) are parameters with fixed values.

Note that the exchanger areas are directly calculated in the objective using the LMTD approximation Eq.4.2. Since the model assumes isothermal mixing, the temperature at the end of each interval as shown in Figure 4.1 is equal for all streams, which means that the mixing and other constraints in the model are linear.

4.2.2 Network NLP model

The network-level NLP model is used to incorporate non-isothermal mixing with stream splitting and stream bypass. The model is used to calculate the optimal flowrates for streams/sub-streams and the inlet-outlet temperatures of each exchanger. The binary variables from the solution of the SWS MINLP model are used as fixed parameters to initialize exchangers between streams. The flow rates and the temperatures are variables with constraints for non-isothermal mixing and bypass.

- **Non-isothermal mixing constraints**(Appendix Eq.16 and 18): Non-isothermal mixing is modeled using bilinear constraints for streams, which are selected from the MINLP solution at each stage. Intermediate temperatures are defined along with split flowrates to model the heat balance for every selected stream. The split flowrates are bounded by the total flow rate of the stream and the intermediate temperatures are constrained to be monotonic.
- **Bypass constraints**(Appendix Eq.26): Stream bypassing reduces the individual pressure drops and increases area costs, and can reduce the overall TAC of the network through additional degrees of freedom. Instead of adding constraints, the stream bypass is modeled as a choice by removing a subset of constraints, which prohibit split streams to have zero flowrates whenever an exchanger exists for that stream.

Unlike in the MINLP, the pressure drops in the NLP model are variables and are calculated either with reduced model or correlations with correction factors. The objective function is the same as in the MINLP model. Since the NLP model consists of nonlinear constraints it is nonconvex and a global optimum is difficult to obtain. As it is shown in the next two sections, a global solution for the NLP model is not required and only a feasible solution is enough to guarantee convergence to the solution of the overall HENS design problem. The complete NLP model with variables and constraints is described in detail in the Appendix section (Eq.14-26).

4.3 Two-step Hybrid Strategy

The hybrid strategy is a way to incorporate details of the exchanger design in the synthesis of heat exchanger network. It does that by comparing the exchanger area, number of shells, pressure drops and heat transfer coefficients from the detailed exchanger design and the values calculated in the HENS MINLP and NLP models.

4.3.1 Exchanger Design model

The detailed design for individual exchangers in the network is determined by solving the DAE based exchanger model described in Chapter 3. The flowrates and the inlet-outlet temperatures across each exchanger calculated from the network NLP model are used as inputs to the exchanger design model.

The design strategy involves solving design equations using the LMTD equation and determining the optimal values for integer decision variables such as number of shells, tube-passes, baffles, tube dimension and stream allocation. The integer variables determine the geometrical structure of the exchanger which is then used to discretize the interior of the exchanger into small discrete elements. The discrete elements are then used to solve the first principles based DAE model consisting of heat equation and design equation for exchanger area and heat transfer coefficients, which are then used to determine the correction factors.

4.3.2 Correction Factors

The correction factors act as updates in the HENS model and adjust the value of design variables: number of shells, pressure drops and heat transfer coefficients. There are two sets of correction factors: the MINLP correction factor and the NLP correction factors. The factors are calculated by doing a ratio comparison between the values in the HENS model and the exchanger design model. The derivation of these corrections factors and their implementation in the HENS models is shown in the Appendix section .3.

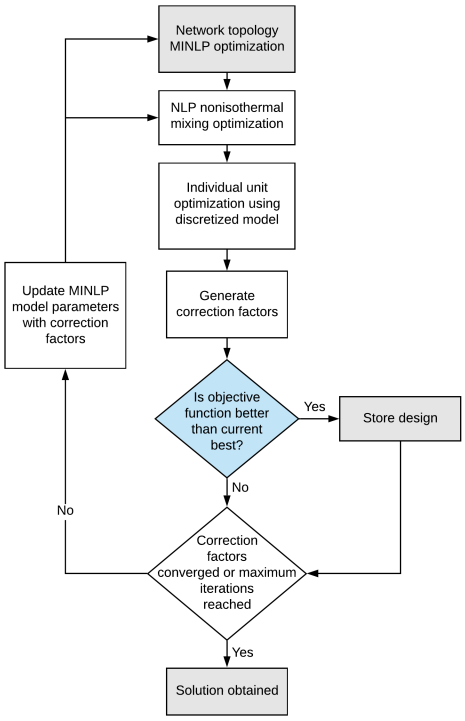


Figure 4.3: Flowchart for two-step hybrid strategy

The hybrid strategy is presented as a flowchart in Figure 4.3. A filter is also applied so that values for the correction factors do not change too dramatically between iterations. All correction factors are set to 1 at iteration 0 and a maximum change of 5 % is allowed between iterations, as was done in Short et al. [37].

Note that the method is sensitive to the initial value of the parameters and it is favorable to choose parameters as close to the correct value as possible, but also which underestimate the objective function. This ensures that no potential solutions are excluded. While the method cannot guarantee globally optimal solutions for the full problem, the coupling of the solutions of the detailed unit models with the MINLP and NLP subproblems does allow for the topology optimization (solved with a global solver) to be ”guided” by information provided from the detailed models through these correction factors.

4.4 Trust Region Based Strategy

Trust region methods are another way to solve the simultaneous optimization of HEN with detailed exchanger design. Trust Region methods are used to solve both nonlinear system of equations and optimization problems. The main principle of trust region methods is to approximate the objective or constraint functions by a surrogate function within a region of confidence known as the trust region. The trust region is expanded or contracted based on the accuracy of the approximation and the solution of the optimization problem.

In the simultaneous HENS design problem, the network NLP model is used as the optimization problem with the detailed exchanger design being the black-box constraints. The HENS NLP model can be represented as:

$$\min_{x} \quad f(x), \tag{4.4a}$$

$$s.t. \quad h(x) = 0, \tag{4.4b}$$

$$g(x) \leq 0, \tag{4.4c}$$

$$y = d(w) \tag{4.4d}$$

where $x^T = [z^T, y^T, w^T]$, w and y denote the input and outputs to the black-box function (Detailed Exchanger Design, $d(w)$) respectively.

The black box constraints are approximated using a reduced model which is updated after each iterative solution of the trust region subproblem ($TRSP_k$). These updates are derived by solving the DAE model for detailed exchanger design and calculating the sensitivity of the design w.r.t to the model inputs.

$$\min_{x} \quad f(x), \tag{4.5a}$$

$$s.t. \quad h(x) = 0, \tag{4.5b}$$

$$g(x) \leq 0, \tag{4.5c}$$

$$y = r_k(w), \tag{4.5d}$$

$$||x - x_k|| \leq \Delta_k \tag{4.5e}$$

$r_k(w)$ - reduced model, Δ_k - trust region radius and x_k - previous solution

A trust region constraint is added which restricts the NLP solution to be bounded within the confidence region or trust region of approximation. The size of this trust region or trust region radius (Δ_k) is determined using a filter which compares the NLP objective value (Eq.4.5) and the accuracy of the reduced model with the values from previous iterations.

- **Reduced Models:** The NLP suboptimization model incorporates detailed exchanger designs using RMs which are based on both shortcut/surrogate models and outputs from the DAE-based design model. The RMs used in this study are constructed based on the matching conditions in Alexandrov et al. [60] and Yoshio and Biegler [61] i.e. $r_k(x_k) = d(x_k)$ and $\nabla r_k(x_k) = \nabla d(x_k)$.

$$r_k(x) = s(x) + (d(x_k) - s(x_k)) + (\nabla d(x_k) - \nabla s(x_k))^T (x - x_k) \tag{4.6}$$

  where s represents the surrogate model. Note that in the absence of any surrogate model (where $s(x)$ is set to an arbitrary constant) the above equation reduces to the first-order Taylor series expansion of $d(x)$ at x_k.
- **Trust region constraints:** Trust region constraints are added to ensure the solution of the NLP x^* is close enough to x_k such that the approximation is a valid reduced model. The constraint is defined on the infinite-norm of the distance vector: $\|x - x_k\|_\infty \leq \Delta_k$ where Δ_k is the trust region radius.

Note that it is possible that the $TRSP_k$ becomes infeasible for the given value of Δ_k. This can be addressed by an additional 'consistency' step in the original TR method [62].

$$\min_{x,q} \quad f(x) + \beta q^T e, \tag{4.7a}$$

$$s.t. \quad h(x) = 0, \tag{4.7b}$$

$$g(x) \leq q, \;\; q \geq 0 \tag{4.7c}$$

$$y = r_k(w), \tag{4.7d}$$

$$||u - u_k|| \leq \Delta_k \tag{4.7e}$$

where $e = [1, 1, \ldots, 1]^T$. On the other hand, in this study the $TRSP_k$ is reformulated by adding a penalty variable to $g(x) \leq 0$, and adding an ℓ_1 penalty to the objective. Also by

partitioning $x = [u^T \ v^T]^T$ (u-degrees of freedom, v-derived variables) trust region constraint can be reformulated as described in Yoshio and Biegler [61]. Assuming that the equality constraints are solvable for any given u_k, this reformulated NLP is always feasible and has the same solution as in Eq.4.5 for sufficiently large values of the penalty parameter β.

4.4.1 Trust Region Filter

The trust region filter (TRF) strategy developed here is modified from [62] and [61]. The filter was originally developed by Fletcher et al. [63] for solving NLPs using SQP method and was modified later to solve NLPs with black-box constraints.

The TRF method compares the improvement in model feasibility and the objective function value to determine the value of trust region radius for the next iterate. Define: model infeasibility (θ) and the search step (Δx_k) as: $\theta(x) = ||r_k(w) - d(w)||$ and $\Delta x_k = x_k^* - x_k$, along with a filter set $\mathcal{F}_k = \{(\theta_j, f_j) : j < k, j \in \mathcal{Z} \subset \mathbb{N}\}$ which stores the pair of values (θ, f) for a subset of previous iterates. A solution pair (θ_k, f_k) of the $TRSP_k$ is added to the filter if it satisfies a certain level of decrease in either model infeasibility or objective function values, with $\gamma_\theta, \gamma_f \in (0, 1)$ as fixed parameters, i.e.,

$$\theta(x_k + \Delta x_k) \leq (1 - \gamma_\theta)\theta_j \ \text{ or } \ f(x_k + \Delta x_k) \leq f_j - \gamma_f \theta_j \qquad \forall (\theta_j, f_j) \in \mathcal{F}_k \tag{4.8}$$

To ensure that the objective value decreases for small θ, a switching condition is imposed to characterize each step $f(x_k) - f(x_k + \Delta x_k) \geq \kappa_\theta \theta(x_k)^{\gamma_s}$, where $\kappa_\theta \in (0, 1)$ and $\gamma_s \in (1/2, 1)$ are tuning parameters. If the step Δx_k satisfies this switching condition, it is termed an f-type step and not added to the filter; otherwise it is classified as a θ-type step.

4.4.2 Trust Region Radius Update

The trust region radius (Δ) is updated based on the type of step and the previous value of the radius. If the step is f-type then the trust region radius is simply expanded ($\Delta_{k+1} = \gamma_e \Delta_k$), where $\gamma_e > 1$ is the expansion parameter. For a θ-type step the trust region radius is

updated using a ratio test, based on the decrease of infeasibility, $\rho_k = 1 - \frac{\theta(x_k+\Delta x_k)}{\theta(x_k)}$, and the trust region radius is updated by comparing the ratio ρ_k using parameters $0 < \eta_1 \leq \eta_2 < 1$ and $0 < \gamma_c < 1 < \gamma_e$. Here $\Delta_{k+1} = \gamma_c \Delta_k$, if $\rho_k \leq \eta_1$ or $\Delta_{k+1} = \gamma_e \Delta_k$, if $\rho_k \geq \eta_2$. Otherwise, $\Delta_{k+1} = \Delta_k$. A sketch of the TRF algorithm is presented in Figure 4.4 below.

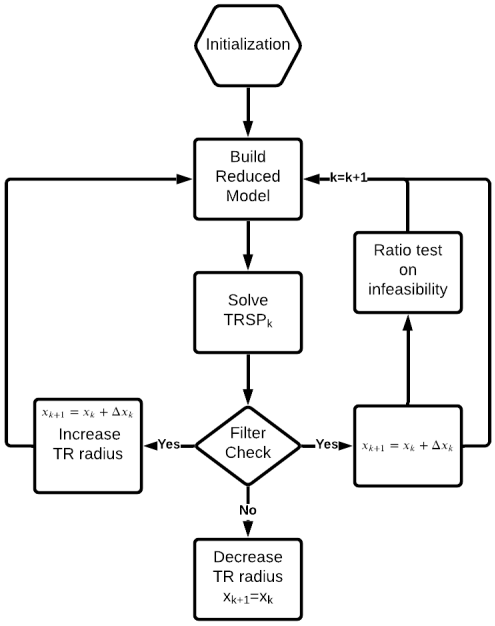


Figure 4.4: Trust Region Algorithm Flowsheet

4.4.3 Integer Cut

The HENS MINLP model and the NLP model with trust region strategy are incorporated using integer cuts. Although the HENS MINLP model assumes isothermal mixing, it has fewer variables and constraints than the NLP model. The MINLP design parameters (heat transfer coefficients, pressure drops and number of shells) are selected such that the MINLP solution is lower than the solution of the overall HENS problem. Since the NLP model incorporates the detailed design of the exchanger using the TRF method, the NLP solution is a feasible solution to the overall HENS problem.

Therefore, the MINLP solution and the NLP solution are used as lower and upper bounds of the overall HENS problem respectively. An integer cut constraint removing

the current integer solution is added to the MINLP model after each solve such that the lower bound is increased monotonically.

$$\sum_{i,j,k \in S_0} z_{i,j,k} + \sum_{i,j,k \in S_1} (1 - z_{i,j,k}) \geq 1 \tag{4.9}$$

where S_0 is the set containing unselected binary variables, and S_1 contains the set with binary variables selected. In addition, some equivalent topologies can also be removed by precomputing equivalent binary combinations, prior to generating the integer cut. This is done to reduce symmetry and save on computation.

The MINLP model is solved again with the integer cuts and if the MINLP objective value (new lower bound) is above the current upper bound, then the current upper bound solution is the optimal network solution of the overall HENS problem. If not, then the upper bound is either updated with the NLP subproblem solution or remains the same. If the MINLP model provides global solutions, then the lower bound will increase monotonically, but it is not necessarily true for the upper bound or the NLP solution. The complete trust region based strategy for simultaneous optimization of HEN with detailed design is shown in Figure 4.5

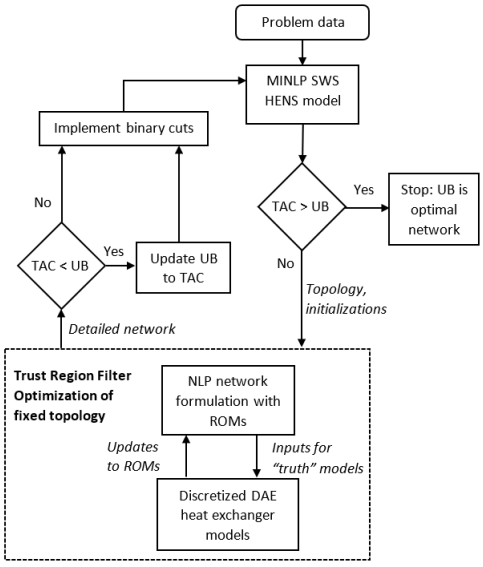


Figure 4.5: The overall trust region filter based strategy for simultaneous HENS design

4.5 Case Study

The hybrid strategy with correction factors and the TRF based strategy both are used to solve three examples which were previously solved in literature ([55, 57, 37, 34]). The stream physical properties are kept constant with values given in Table 4.1 below.

μ (kg/m s)	ρ (kg/m^3)	C_p (J/kg)	k (W/mK)	r_d (W/mK)
2.4E-4	634	2454	0.114	1.7E-4

Table 4.1: Stream data applied for all streams in the examples

Both frameworks were written inside a Python environment using the algebraic modelling language Pyomo [35]. The MINLP models are initially solved with the global solver BARON [64] and if it does not converge, then the optimality gaps are relaxed and solved with either BARON or DICOPT [65] to obtain a feasible integer solution. The NLP models are solved using IPOPT 3.12 with the derivative information from the detailed exchanger model used for the reduced model update. The detailed exchanger design model consists of the DAE model whose sensitivity is calculated using the NLP sensitivity code k_{aug} [7].

In all examples, the utility exchangers are not considered within the detailed designs, in order to compare with the results obtained by other authors who made the same assumption. Each case is solved with both reformulations, i.e. with and without bypass, and the results are listed in comparison with other studies. In the trust region strategy, single tube-pass exchanger model was used only for small (when MINLP area $< 25m^2$) exchangers because these units have high auxiliary costs due to very low tubeside flowrates, require excess maintenance, and high space requirements [15].

4.5.1 Example-1

Example 1 has two hot streams, two cold streams, and single hot and cold utilities, within a two-stage superstructure. Problem parameters are shown in Table 4.2, where NS refers to number of shells. This problem has many solutions that exist close together, as utility costs dominate and exchangers are small.

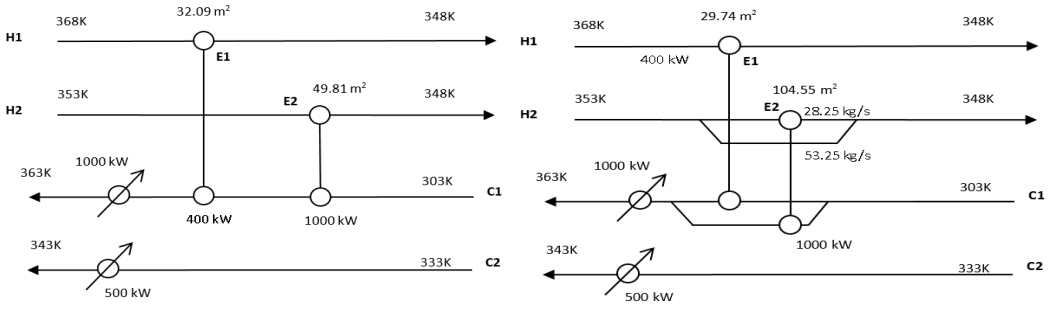
	m (kg/s)	T_{in} (K)	T_{out} (K)
H1	8.15	368	348
H2	81.5	353	348
C1	16.3	303	363
C2	20.4	333	343
HU		500	500
CU		300	320

where ΔT_{min} is 10 K, Area cost = $1000(NS) + 60(Area/NS)^{0.6}\$/year$. Pumping cost = $0.7(\Delta P_t m_t/\rho_t + \Delta P_s m_s/\rho_s)$, where ΔP = Pa, m = kg/s, and $\rho = kg/m^3$. CU cost = \$6/kW year and HU costs = \$60/kW year. Overall heat-transfer coefficients of utility streams = 444 W/m^2K.

Table 4.2: Stream data for Example 1

4.5.1.1 Hybrid Strategy

The hybrid strategy, without bypasses converges after 58 iterations (30 min CPU time) with the best-performing solution found in the 2nd and 14th iterations. With stream bypasses, the algorithm does not converge, oscillating between the same networks repeatedly. The best solution is found in iteration 17 (31 min CPU time) and is shown in Figure 4.6b, having a TAC of \$96,435.73/yr. This network performs better than the solution without bypass, as the high flowrate stream is split in order to lower pumping costs.



(a) Without bypass

(b) With bypass

Figure 4.6: Network results with Hybrid Strategy for Example 1

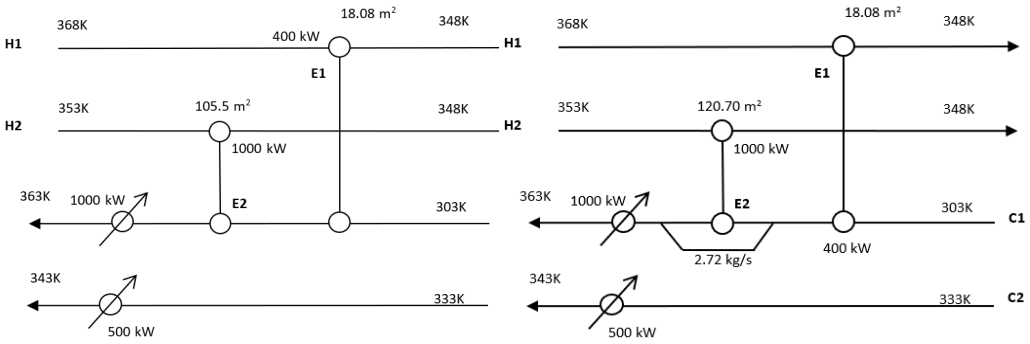
The detailed exchanger design for both the bypass and without bypass configuration are presented in Table 4.3. Note that the area for Exchanger E2 is higher in the network with bypass but the pressure drops on both tube and shell side are lower.

	Without Bypass		With Bypass	
	Exchanger 1	Exchanger 2	Exchanger 1	Exchanger 2
Area(m^2)	32.09	49.81	29.74	104.65
Duty(kW)	400	1000	400	1000
Tube-side flowrate (kg/s)	8.15	16.3	8.15	28.25
Shell-side flowrate (kg/s)	16.3	81.5	4.03	12.27
N_s	1	1	1	1
N_{tp}	4	2	2	2
$D_s(m)$	0.42	0.404	0.33	0.57
N_t	264	170	164	574
N_b	3	2	12	6
$d_o(mm)$	15.88	15.88	15.88	15.88
$d_i(mm)$	13.24	13.24	13.24	13.24
$p_t(mm)$	19.84	19.84	19.84	19.84
$L(m)$	2.438	4.877	3.66	3.66
$v_t(m/s)$	1.42	1.52	1.15	1.13
$v_s(m/s)$	0.500	0.978	0.34	0.32
$h_t(W/m^2.K)$	1940.1	1980.2	1638.0	1618.2
$h_s(W/m^2.K)$	1224.5	1631.6	993.4	964.4
$U(W/m^2.K)$	544.3	614.7	467.0	458.6
$\Delta P_t(kPa)$	23.90	18.33	10.18	9.88
$\Delta P_s(kPa)$	5.40	12.38	6.42	5.38
Hot fluid allocation	Tube	Shell	Tube	Tube
Tube Arrangement	Square	Square	Square	Square

Table 4.3: Detailed heat exchanger results with Hybrid Strategy for Example 1

4.5.1.2 TRF based Strategy

The results for Example 1 with the TRF based strategy is also similar to the solutions of hybrid strategy. The TRF based strategy on without bypass scenario converges after 7 MINLP iterations and 860 CPUs. Within each of these iterations, the TRF algorithm converges in 25 - 40 iterations. The optimal networks are shown in Figure 4.7 and the results are compared with the hybrid strategy and other papers in Table 4.4. When considering bypasses, the algorithm converges in 7 MINLP iterations and about 780 CPUs. This is due to the integer-cut algorithm in combination with the trust-region network NLP.



(a) Without bypass (b) With bypass

Figure 4.7: Network results with TRF Strategy for Example 1

	Without Bypass	With Bypass	Hybrid Strategy	Short et al.[52]	Mizutani et al.[49]	Ravagnani et al.[51]
Total Annual Cost ($/a)	96,714.74	96,858.87	97,360.94	97,159.3	95,852.0	96,137.71
Utility Costs ($/a)	90,000	90,000	90,000	90,036	90,000	90,000
Area Costs ($/a)	1,821.05	1,903.40	1,604.67	1,631.68	1,608.0	1,675.52
Pumping Costs ($/a)	893.68	955.45	1,756.26	1,491.63	244.0	462.19
Fixed Costs ($/a)	4,000	4,000	4,000	4,000	4,000	4,000
Number of matches	2	2	2	2	2	2
Number of exchangers	4	4	4	4	4	4

Table 4.4: Summary of solutions obtained for Example 1 in comparison with other studies

In Table 4.4, the "number of matches" refer to the number of heat exchange services between process streams, while "number of heat exchangers" refers to the total number of exchanger shells, which includes utility exchangers (each comprised of one shell). The exchanger design details are presented in Table 4.5.

	Without Bypass		With Bypass	
	Exchanger 1	Exchanger 2	Exchanger 1	Exchanger 2
Area(m^2)	18.07	105.54	18.07	120.70
Duty(kW)	400	1000	400	1000
Tube-side flowrate (kg/s)	16.3	81.5	16.3	81.5
Shell-side flowrate (kg/s)	8.15	16.3	8.15	13.6
N_s	1	1	1	1
N_{tp}	1	2	1	2
$D_s(m)$	0.35	0.89	0.35	0.95
N_t	204	542	204	620
N_b	4	5	4	10
$d_o(mm)$	15.88	25.4	15.88	25.4
$d_i(mm)$	13.24	21.2	13.24	21.2
$L(m)$	1.78	2.44	1.78	2.44
$v_t(m/s)$	0.92	1.34	0.92	1.18
$v_s(m/s)$	0.51	0.35	0.51	0.51
$h_t(W/m^2.K)$	1373.5	1695.2	1373.5	1522.6
$h_s(W/m^2.K)$	1073.4	820.0	1073.4	1002.4
$U(W/m^2.K)$	453.1	426.1	453.1	453.5
$\Delta P_t(kPa)$	2.17	7.99	2.17	6.13
$\Delta P_s(kPa)$	4.29	5.38	4.29	21.72
Hot fluid allocation	Shell	Tube	Shell	Tube
Arr	Square	Square	Square	Square

Table 4.5: Detailed heat exchanger results with TRF based Strategy for Example 1

Mizutani et al. [55] uses $F_t = 1$ with smaller areas and pressure drops, and the solution of Ravagnani and Caballero [57] contains a unit with 34 baffles for a 4.877 m tube length, thus violating a constraint on minimum baffle spacing. Also the other approaches rely on solving large, nonconvex MINLPs and require significant bounding and initialisation strategies, whereas the TRF based approach solves reliably and quickly.

4.5.2 Example-2

Example 2 consists of three hot streams, three cold streams and single hot cold utilities. Data for the problem are shown in Table 4.6 and the stream data is the same as that of the first example, found in Table 4.1. Note that in this example, the pumping costs are significantly higher than those in the other case studies.

	m (kg/s)	T_{in} (K)	T_{out} (K)
H1	16.3	423	333
H2	65.2	363	333
H3	32.6	454	433
C1	20.4	293	398
C2	24.4	293	373
C3	65.2	283	288
HU		700	700
CU		300	320

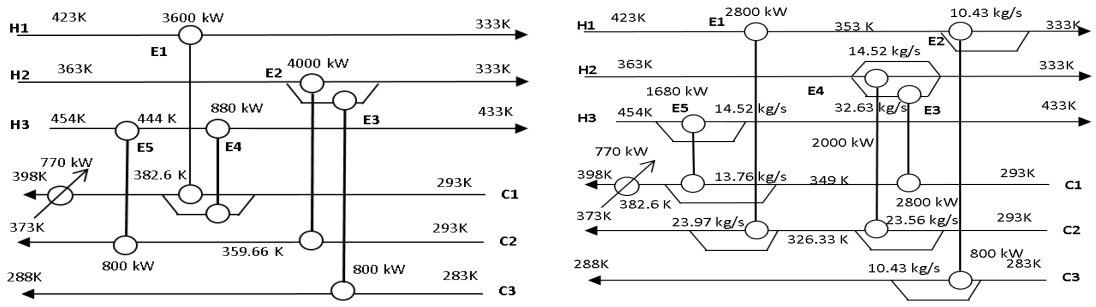
where ΔT_{min} is 10 K, Area cost = $1000(NS) + 60(Area/NS)^{0.6}\$/year$. Pumping cost = $1.3(\Delta P_t m_t/\rho_t + \Delta P_s m_s/\rho_s)$, where ΔP = Pa, m = kg/s, and $\rho = kg/m^3$. CU cost = \$6/kW year and HU costs = \$60/kW year. Overall heat-transfer coefficients of utility streams = 444 W/m^2K.

Table 4.6: Stream data for Example 2

4.5.2.1 Hybrid Strategy

The hybrid strategy obtains a TAC solution of \$76,686.29/yr in the 110th iteration of the algorithm for network without bypasses. The solution includes the extra fixed cost associated with multiple shells, as shown in the objective function. The solution contains two

matches that require multiple shell exchangers, and thus inflate the objective function and significantly increase pressure drops for those matches as shown in Figure 4.8a.



(a) Without bypass (b) With bypass

Figure 4.8: Network results with Hybrid Strategy for Example 2

For the network solution with stream bypass, the best performing network is obtained in iteration 33 of the algorithm. This can be seen in the evolution of the objective functions over multiple iterations, shown in Figure 4.9. Notice how the topology optimizations begin by underestimating the objective function and how the corrections factors drive the two solutions (of the detailed unit optimizations and network solutions) towards each other. The best solution found at iteration 33 is identified after 45 CPU minutes of computational time. The best network topology found is shown in Figure 4.8b.

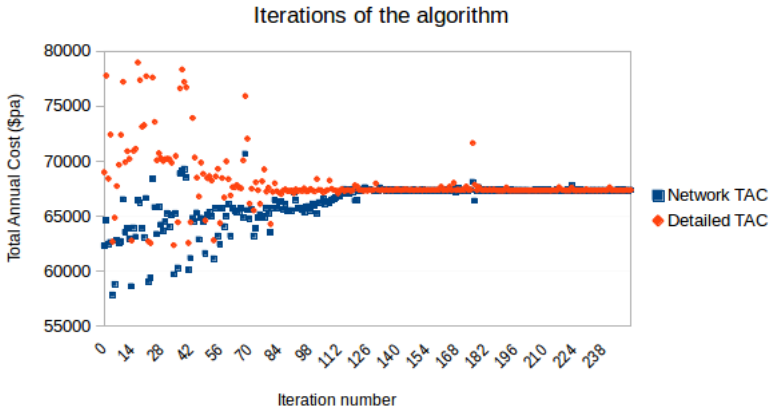


Figure 4.9: The solutions of the network topology and the detailed network designs over the course of the algorithm.

The exchanger design for Example 2 optimal network with bypass configuration is presented in Table 4.7.

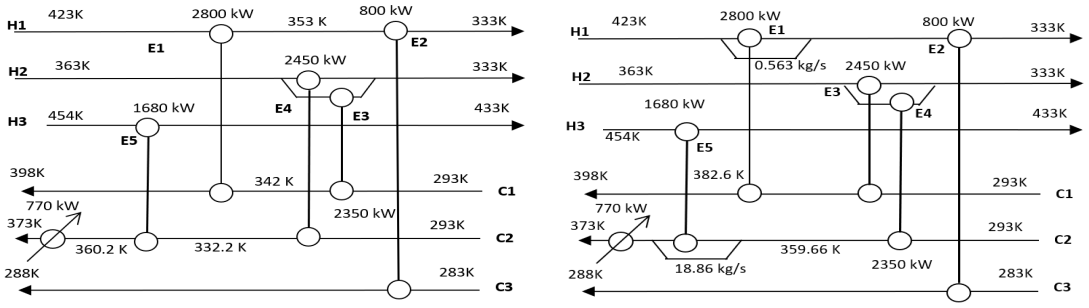
	Exchanger 1	Exchanger 2	Exchanger 3	Exchanger 4	Exchanger 5
Area(m^2)	173.7	49.77	305.89	130.45	71.74
Duty(kW)	2800	800	2800	2000	1680
N_{tp}	4	4	4	4	4
NS	2	1	2	1	1
$D_s(m)$	0.53	0.51	0.6	0.78	0.6
N_t	478	410	630	1074	590
N_b	6	5	7	3	5
$d_o(mm)$	15.88	15.88	15.88	15.88	15.88
$d_i(mm)$	13.24	13.24	13.24	13.24	13.24
$p_t(mm)$	19.84	19.84	19.84	19.84	19.84
$L(m)$	3.658	2.438	4.877	2.438	2.438
$v_s(m/s)$	0.466	0.392	0.443	0.402	0.449
$v_t(m/s)$	1.148	1.156	1.189	1.119	1.142
$h_s(W/m^2.K)$	1178.0	1071.3	1145.4	1085.8	1130.1
$h_t(W/m^2.K)$	1641.0	1649.9	1687.8	1607.7	1633.4
$U(W/m^2.K)$	504.5	484.7	503.5	483.2	494.7
$\Delta P_t(kPa)$	20.44	16.07	26.80	15.08	15.68
$\Delta P_s(kPa)$	20.55	6.05	23.92	6.48	8.63
Hot fluid allocation	Shell	Tube	Tube	Tube	Tube

Table 4.7: Detailed heat exchanger results with Hybrid Strategy for Example 2 with bypass

4.5.2.2 TRF based Strategy

Using the TRF based strategy, the solution for Example 2 without bypass is obtained within two MINLP iterations, with 32 trust region iterations for the network NLP solution. For the case with bypass, the algorithm terminates after 580 CPUs, finding the optimal solution in the first iteration, with the TRF algorithm terminating after 26 iterations. The optimal

networks are shown in Figure 4.10 and the results are compared with the hybrid strategy and other papers in Table 4.8.



(a) Without bypass

(b) With bypass

Figure 4.10: Network results with TRF based Strategy for Example 2

	Without Bypass	With Bypass	Hybrid Strategy	Mizutani et al.(2003b)	Ravagnani et al.(2007b)
Total Annual Cost ($/a)	74,217.37	73,845.71	76,686.29	190,532	74,165.48*
Utility Costs ($/a)	46,200	46,200	46,200	173,456	46,200
Area Costs ($/a)	8,135.85	8,621.58	8,410.50	3,388	13,887.57
Pumping Costs ($/a)	9,509.85	9,024.12	13,075.79	17,076	2,077.91
Fixed Costs ($/a)	10,000	10,000	9,000	9,000	11,000*
Number of matches	5	5	5	6	6
Number of exchangers	10	10	9	9	11*

Table 4.8: Summary of solutions obtained for Example 2 in comparison with other studies
* Authors did not add fixed cost for number of shells, and so this value is updated to more accurately depict results in this thesis

Similar to previous example, the network with stream bypass has a lower TAC than without bypass network solution. The solutions given by the hybrid strategy and the TRF based strategy are either better or competitive to the literature results. The exchanger design for the network without bypass using TRF strategy is presented in Table 4.9

	Exchanger 1	Exchanger 2	Exchanger 3	Exchanger 4	Exchanger 5
Area(m^2)	399.4	44.1	193.0	159.2	31.4
Duty(kW)	2800	800	2450	2350	1680
N_{tp}	4	1	2	6	2
NS	4	1	2	1	1
$D_s(m)$	0.70	0.84	0.61	0.84	0.53
N_t	822	488	442	1310	162
N_b	5	2	5	3	4
$d_o(mm)$	15.88	21.2	19.05	15.88	21.2
$d_i(mm)$	13.24	25.4	15.88	13.24	25.4
$p_t(mm)$	19.84	31.75	23.81	19.84	31.75
$L(m)$	2.438	1.13	3.658	2.438	2.438
$v_s(m/s)$	0.454	0.404	0.431	0.376	0.752
$v_t(m/s)$	1.137	0.598	1.201	1.676	1.81
$h_s(W/m^2.K)$	1161	765	1039	1046	1240
$h_t(W/m^2.K)$	1628	886	1641	2221	2149
$U(W/m^2.K)$	500	324.7	505.2	527	561
$\Delta P_t(kPa)$	62.2	0.622	19.2	49.9	14.4
$\Delta P_s(kPa)$	44.1	2.41	14.5	6.09	11.9
Hot fluid allocation	Shell	Shell	Shell	Shell	Shell

Table 4.9: Detailed heat exchanger results with TRF based Strategy for Example 2 without bypass

4.5.3 Example-3

This example has 7 hot process streams, 3 cold process streams, and single hot and cold utilities, with stream data presented in Table 4.10. This network contains significantly larger exchangers, where multiple shells are common in practice. In this example, the utility costs dominate the final TAC, and optimal usage of process heat is imperative. The solutions contain a higher number of matches than other results in the literature, meaning that more heat exchanger services between process streams are utilized than other studies.

	m (kg/s)	T_{in} (K)	T_{out} (K)
H1	134	413	313
H2	235	433	393
H3	12.1	483	318
H4	28.5	533	333
H5	102	553	483
H6	14.2	623	443
H7	38.9	653	433
C1	235	543	658
C2	143	403	543
C3	104	293	403
HU		700	700
CU		293	298

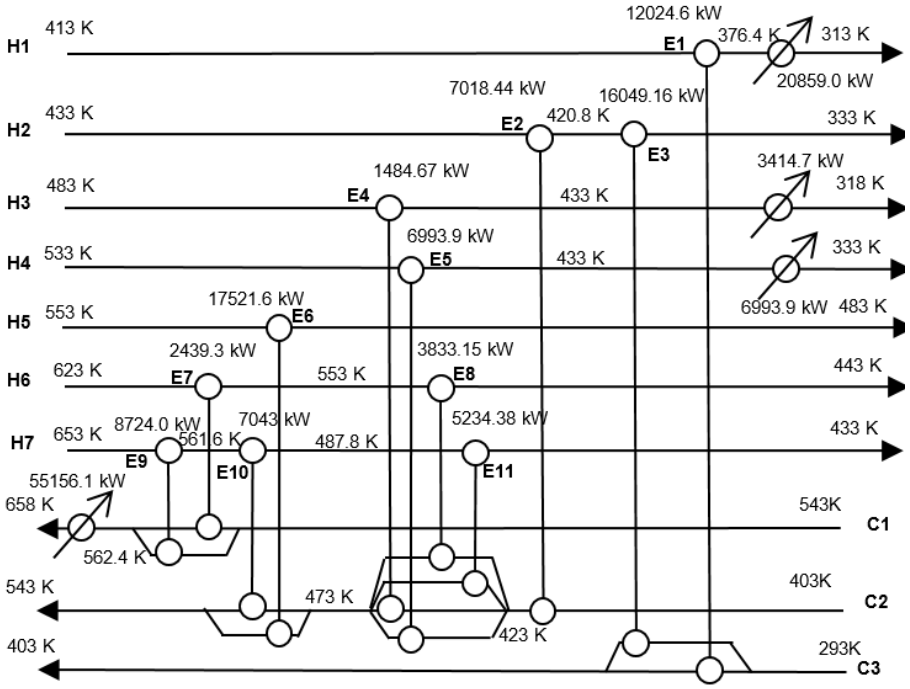
where ΔT_{min} is 10 K, Area cost = $1000(NS) + 60(Area/NS)^{0.6}\$/year$. Pumping cost = $0.7(\Delta P_t m_t/\rho_t + \Delta P_s m_s/\rho_s)$, where ΔP = Pa, m = kg/s, and $\rho = kg/m^3$. CU cost = \$6/kW year and HU costs = \$60/kW year. Overall heat-transfer coefficients of utility streams = 444 W/m^2K.

Table 4.10: Stream data for Example 3

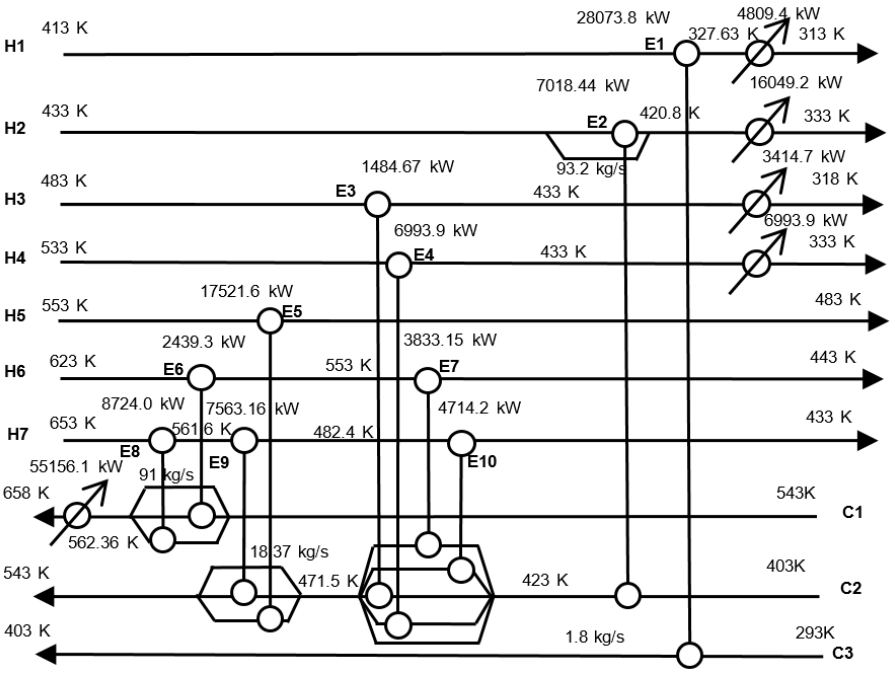
4.5.3.1 Hybrid Strategy

The hybrid strategy solution for both the network scenarios: without and with bypass has the same utility cost but with different number of utility exchangers. Both networks look very similar as shown in Figure 4.11 except the minor difference in exchangers between hot stream H2 and cold streams C2, C3 and the utility exchanger on H2 in the network with bypass.

Due to the larger size of the problem, the hybrid strategy requires a lot more computational time and does not converge for the network with bypass within 24hrs. The best network solution was found in iteration 28 which took about 7hrs of CPU time with each MINLP solution taking more than 5 CPU min. The details about the exchanger design are shown in Appendix section .5.



(a) Without Bypass



(b) With Bypass

Figure 4.11: Exchanger Network for Example 3 without bypasses using Hybrid Strategy

Unlike in previous examples, the network without bypass has a lower TAC value ($3,679,782/yr) than the network with bypass ($3,701,540/yr). This is most probably because the hybrid strategy was not able to converge the MINLP solution and the solution with detailed exchangers within 2% tolerance.

4.5.3.2 TRF based Strategy

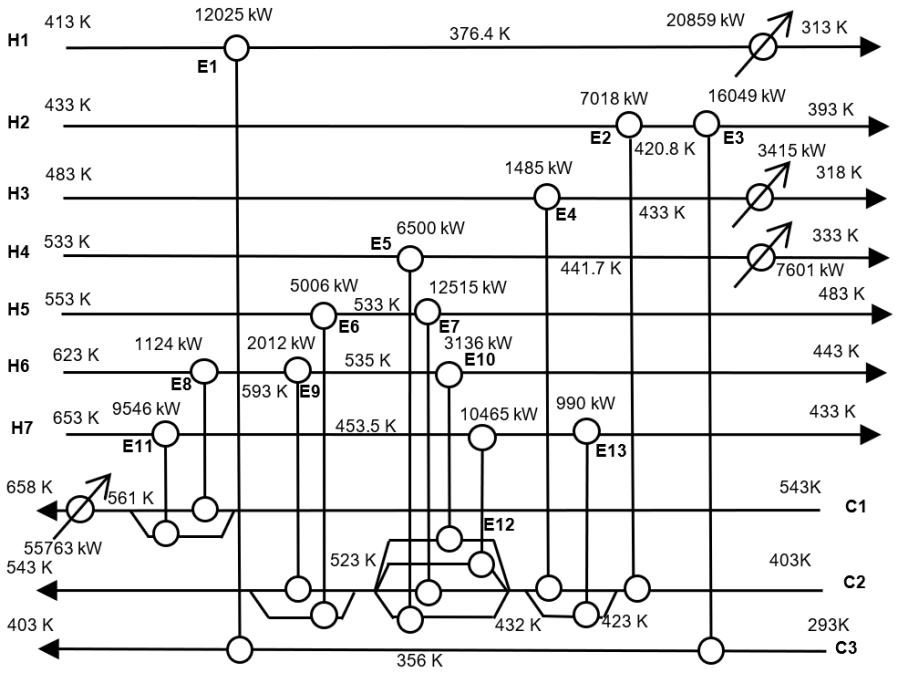
The TRF based strategy solves the Example 3 with much better performance in terms of solution time. The algorithm terminates after 2 MINLP evaluations for the cases with and without bypasses; the trust region algorithm terminates in 60 and 35 iterations respectively. The algorithm converges in 3,888 CPUs (1.08 hours) without bypasses and in 6,159 CPUs with bypasses (1.71 hours).

The Table 4.11 compares the solution of both strategies with other literature results.

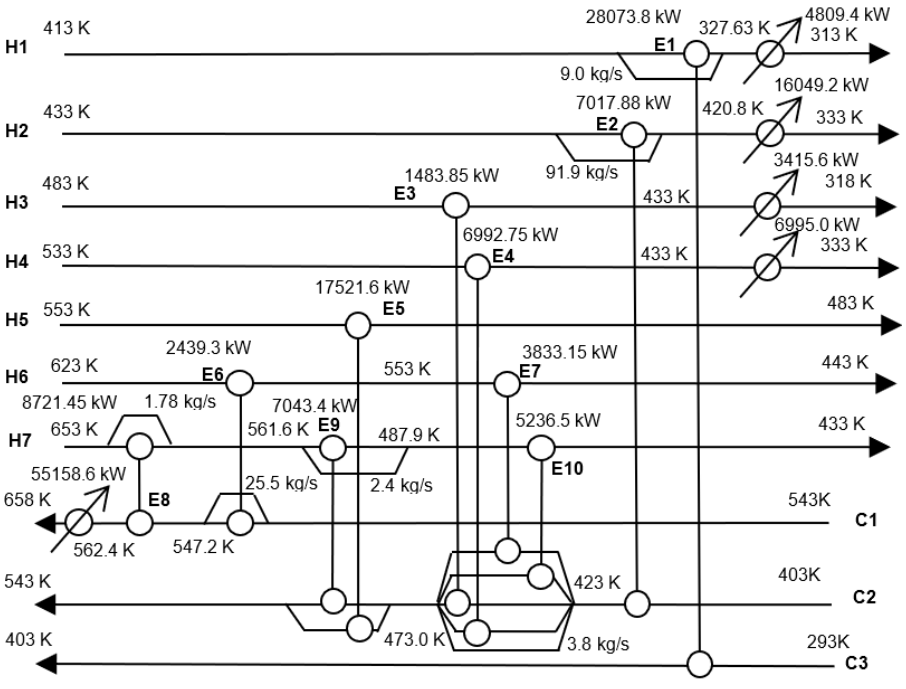
	TRF based Strategy		Hybrid Strategy			
	Without Bypass	With Bypass	Without Bypass	With Bypass	Mizutani et al.(2003b)	Short et al.(2016b)
Total Annual Cost ($/a)	3,766,112	3,764,984	3,679,782	3,701,540	5,183,221	4,203,057
Utility Costs ($/a)	3,529,527	3,497,138	3,496,972	3,496,972	5,154,291	4,091,975
Area Costs ($/a)	76,620	60,874	77,118	88,697	11,123	42,981.97
Pumping Costs ($/a)	113,965	163,972	67,692	73,871	4,807	46,099
Fixed Costs ($/a)	46,000	43,000	38,000	42,000	13,000	22,000
Number of matches	13	10	11	10	8	8
Number of exchangers	46	43	38	42	13	22

Table 4.11: Summary of solutions obtained for Example 3 in comparison with other studies

Note that although it may seem that the TRF based strategy solution is worse than the Hybrid strategy solution, the convergence of the solution and the use of sensitivity information in the TRF strategy makes its solution more feasible and closer to the optimal solution. The TRF strategy solution network for both without bypass and with bypass are presented in Figure 4.12 and the exchanger design details are given in the Appendix .5.



(a) Without Bypass



(b) With Bypass

Figure 4.12: Exchanger Network for Example 3 without bypasses using TRF based Strategy

4.6 Conclusions

The HEN synthesis is an old classical problem in the systems engineering community with wide applications in industry. Over the 20^{th} and 21^{st} century, many publications and books have introduced different methods and models to solve the HENS problem. Unfortunately, only a handful of papers have addressed the need to incorporate the exchanger design (geometry and size) inside HENS models.

Previous papers have limited to using single shell exchangers [55, 57, 34] and updating heat transfer coefficients. In this chapter, a two-step hybrid strategy originally introduced in [37] was extended to incorporate the DAE model for exchanger design inside HENS model using correction factors. Alternatively, a trust region based method using filter condition and sensitivity information from the DAE solution is developed and combined with an integer cut based strategy to include exchanger details inside the HENS solution.

Both the strategies are used to solve three standard examples from literature on networks with and without bypass and their results are discussed and compared with other studies. The results show that both hybrid strategy and trust region based strategy perform better than the other methods and obtain more feasible network solution with detailed exchanger design. The trust region based strategy is much faster than the hybrid strategy in solving the HENS problem and obtains better results except in the last example, where the algorithm gets stuck in a local minimum.

Both strategies can also be implemented with different HENS MINLP models including the model proposed by Ponce-Ortega et al. [66] for process streams undergoing isothermal phase change. The detailed exchanger design for phase change exchangers can be determined using the flooded exchanger models described in the previous chapter. The performance of both strategies provides a motivation to apply them in other heat exchanger based network problems like WHENS or mass and heat exchanger network synthesis (MHENS).

Chapter 5

Multi-Stream Heat Exchanger Design

This chapter describes the modeling and design of a different type of heat exchanger: Multi-Stream Heat Exchanger (MHEX). A brief introduction to MHEX and its application in process industry is presented. A brief overview on the literature study of MHEX design and optimization is discussed.

A novel DAE-based design model for spiral wound heat exchanger (SWHX), a common type of MHEX is introduced. The model is extended to incorporate phase change for multi-component mixture using complementarity based flash equations. The discretization for the DAE mode based on geometric structure with correlation for design variables is presented. The MHEX DAE model is used inside a natural gas liquefaction flowsheet and solved as a simultaneous design optimization problem.

5.1 Introduction

Multi-stream heat exchangers (MHEX) are exchangers with more than two streams flowing inside them and are common in industries with cryogenic processes. An air separation process to produce high purity oxygen and nitrogen using thermal distillation is a very energy intensive cryogenic process. Air separation units (ASUs) use MHEX to exchange heat between multiple streams of nitrogen-oxygen mixture at very low temperatures. The MHEX is the most important unit with the highest heat duty as it acts as both the condenser and the reboiler for the air separation column.

Another application of MHEX is the natural gas liquefaction process, where the natural gas is liquefied using a thermodynamic cycle and refrigerants. Natural gas liquefaction is

required for transporting natural gas in ships and vessels and can contribute upto 52% of the cost of liquefied natural gas (LNG) [67]. Due to the increase in natural gas demand and production, the optimization of natural gas liquefaction process has become an important topic in energy industry.

There are mainly three types of natural gas liquefaction (NGL) processes i.e. a) cascade liquefaction process, b) mixed refrigerant liquefaction process and c) expander based liquefaction process. Cascade liquefaction processes consist of multiple independent pure refrigeration cycles where the natural gas is cooled using propane, ethylene and methane as refrigerants sequentially. The cascade liquefaction process has high energy efficiency compared to other type of liquefaction process but its capital costs are high because of its complex design and additional individual units required.

Mixed refrigerant (MR) liquefaction processes use a nitrogen and hydrocarbon mixture (methane, ethane, propane, i-butane and n-butane). They require fewer units than cascade refrigeration process and the energy consumption is significantly lower. MR liquefaction process can have single (SMR) (eg. PRICO process) or dual (DMR) refrigeration loop cycles. Another type of MR liquefaction process (C3MR) has a pre-cooling stage where liquid propane (C3) is used as a refrigerant to cool the natural gas before liquefying using the MR. This increases the energy efficiency of the process but requires more heat transfer and pressure control units.

The expander based liquefaction process uses pure nitrogen or methane as the refrigerant in a single stage refrigeration cycle with multiple compressor stages. This process is the simplest method for NG liquefaction and requires the least number of thermodynamic units in it. Unfortunately the method has lower energy efficiency than MR processes and is not suitable for high LNG yield.

Review papers (Austbø et al. [68] and Qyyum et al. [69]) on optimization of natural gas liquefaction processes provide an extensive analysis of different methods and studies done on the topic. Most studies used process simulator software like Aspen Plus or Hysys

combined with either stochastic or deterministic optimization solvers in MATLAB. Some studies have also used equation-oriented (EO) modeling in GAMS, AMPL or gPROMS with continuous and mixed-integer optimization tools like IPOPT [5], CONOPT [6] and DICOPT [65].

Previous studies [70, 71, 72] have used simple enthalpy balances to model the MHEX in the NGL process and neglect the effect of the exchanger design on the thermodynamic performance of the refrigerant cycles. This could severely effect the optimization results as the MHEX design has strong correlations with the process variables and the energy efficiency of the process. To date, only Tsay et al. [73] have presented an equation oriented model of natural gas liquefaction process with detailed multi-stream heat exchanger design. They used heat transfer coefficient and pressure drop correlations to model temperature and pressure variation inside the MHEX in the PRICO process flowsheet. Their MHEX design model was embedded inside the PRICO flowsheet and solved using a pseudo-transient EO approach which converts system of nonlinear equations into a nonlinear differential equation system. They used shooting methods in gPROMS to integrate the differential equations and the SQP algorithm as the optimization solver to solve the problem.

In this chapter, a new DAE-based design model is developed for multi-stream spiral wound heat exchanger (SWHX), the type of MHEX used in LNG processes. The DAE model is derived using the heat equation representing the heat transfer between the streams. The phase change for inside the exchanger is modeled using complementarity constraints. Design equations relating the geometric design of SWHX and the heat transfer area are derived. Finally the DAE model is solved within a NG liquefaction flowsheet model as a simultaneous design optimization of the complete process.

5.2 Multi-Stream Heat Exchanger DAE model

MHEX(s) have more complex design and geometry than single stream heat exchangers, which makes it difficult to build an optimization model for MHEX design. The SWHX consists of a central rod-like structure around which multiple tubes are coiled in a circular

helix shape as shown in Figure 5.1. The concentric coils have different radii and appear to form a spiral shape around the central rod.

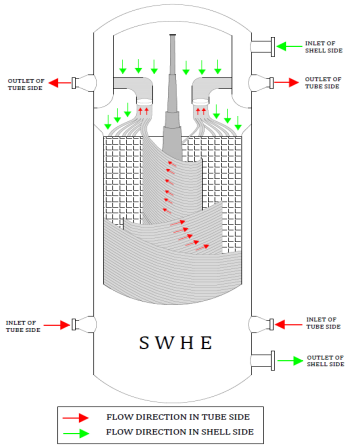


Figure 5.1: Inside view of spiral wound heat exchanger

The tubeside and shellside streams enter the exchanger from opposite directions and exchange heat over the tube curved surface. Multiple streams in the tubeside (usually hot) and single stream on the shellside (usually cold) flow in a counter-current cross-flow pattern inside the MHEX as shown in Figure 5.1. The internal design of SWHX as a cross-section is shown in Figure 5.2.

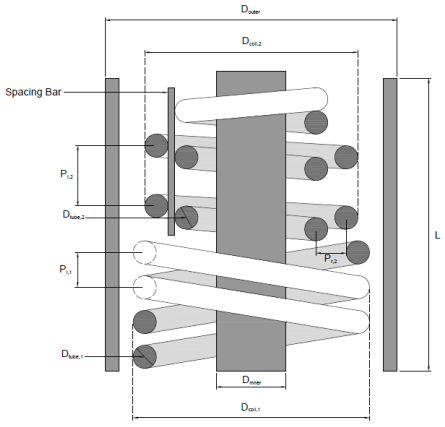


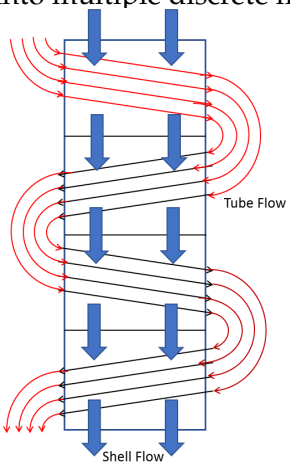
Figure 5.2: Cross-section view of SWHX

Multiple studies have developed SWHX design model with differential heat transfer equations with different methods. Fredheim [74], Neeraas [75] and Fossas [76] used a simplified rectangular geometry to discretize the differential equations and solved the system with thermodynamic equations and heat transfer correlations using a nested approach. Other studies have used CFD methods [77] or developed analytical solutions [78] to model and design SWHX. These models are non-smooth in nature as they contain non-differentiable equations or for-loops to integrate the heat equations, which makes them unsuitable for optimization.

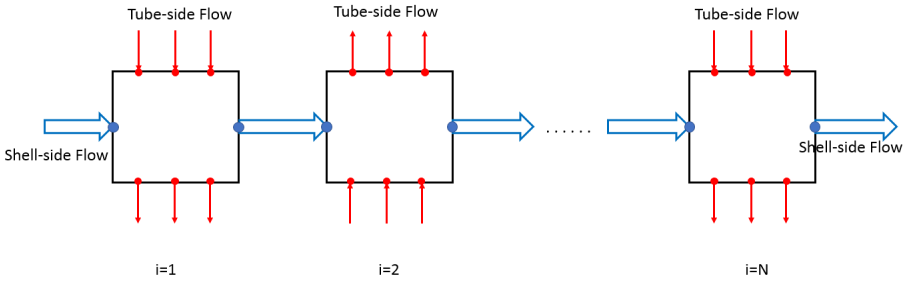
Veerabhadrappa et al. [79] proposed a finite element based strategy to model MHEX where they connect and relate the heat equation of the tubeside and shellside streams using a common independent variable: heat exchanger area (A). A similar approach is used to model the heat transfer inside SWHX as described in following sections.

5.2.1 Heat Equation

The heat equation for the streams inside SWHX is derived by discretizing the exchanger into multiple discrete finite elements as shown in Figure 5.3.



(a) Flow pattern inside SWHX



(b) SWHX Discrete Elements

Figure 5.3: Discretization of SWHX into discrete finite elements

The steady-state heat equation for tubeside and shellside streams can be written in similar way as shown in Chapter 3:

$$\rho_t C_t^p u_x \frac{dT_t}{dx} = -q_v \qquad \text{(Tubeside)} \tag{5.1a}$$

$$\rho_s C_s^p u_y \frac{dT_s}{dy} = q_v \qquad \text{(Shellside)} \tag{5.1b}$$

where ρ is molar density, C_p is specific heat capacity, u is velocity, k is thermal conductivity, T_t, T_s is tube and shell temperature, and q_v is the volumetric heat.

The following assumptions are made:

- Zero or negligible heat transfer by conduction
- Shell and coil side fluid flow are unidirectional

Since the streams are mixtures and are undergoing phase change, it is more meaningful to use enthalpy as the differential variable. Also substituting flow rate $F_t = \rho_t u_x dydz$ $(F_s = \rho_s u_y dxdz)$ and volumetric heat flux $q_v = \frac{UdA(T_t - T_s)}{dxdydz}$.

$$F_t dH_t + UdA(T_t - T_s) = 0 \tag{5.2a}$$

$$F_s dH_s - UdA(T_t - T_s) = 0 \tag{5.2b}$$

The coupled ODE system Eq.5.2 is discretized using the smaller discrete elements (shown in Figure 5.4) and FEM with hat functions as the basis functions.

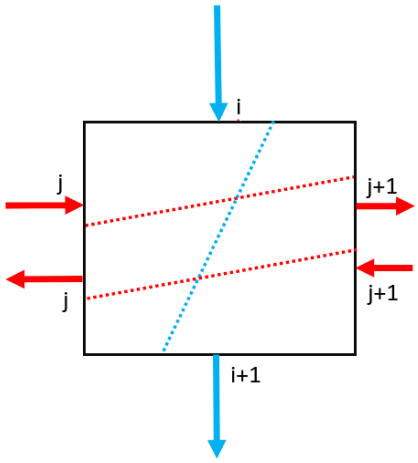


Figure 5.4: Discrete Element inside SWHX

As there are multiple tubeside streams as hydrocarbon mixture and multiple discrete elements inside the SWHX, the heat equation (Eq. 5.2) for each element is reformulated. The following sets are defined for a SWHX model with N discrete elements.

Definition

- S_c - set of tubeside streams
- C - set of components
- N_f = {1,2,3,...N} - set of discrete elements
- N_s = {1,2,3,...N+1} - set of nodes on shell side
- N_c = {(1,1),(1,2),(2,1),(2,2)....(N,1),(N,2)} - set of nodes on coil side

The heat equation for both tubeside and shellside streams inside a discrete element is written as:

$$F_j dH_j + U_{j,s} dA_{j,s}(T_j - T_s) = 0 \qquad \forall j \in S_c \tag{5.3a}$$

$$F_s dH_s - \sum_{k}^{k \in S_c} U_{k,s} dA_{k,s}(T_k - T_s) = 0 \tag{5.3b}$$

Note that it is easy to extend the model for multiple shellside streams as well with concentric shells. Discretizing the coupled ODE-system using the FEM (described in Chapter 3) results in the following system of difference equations.

For $i = 1, 2, 3, ...N$

$$F_j \frac{(H_j^{i,2} - H_j^{i,1})}{2} + \frac{U_{j,s}\Delta A_{j,s}}{3}(T_j^{i,2} - T_s^{i+1}) + \frac{U_{j,s}\Delta A_{j,s}}{6}(T_j^{i,1} - T_s^{i}) = 0 \qquad \forall j \in S_c \tag{5.4a}$$

$$F_s \frac{(H_s^{i+1} - H_s^{i})}{2} - \sum_{k}^{k \in S_c} \Big[\frac{U_{k,s}\Delta A_{k,s}}{3}(T_k^{i,2} - T_s^{i+1}) + \frac{U_{k,s}\Delta A_{k,s}}{6}(T_k^{i,1} - T_s^{i})\Big] = 0 \tag{5.4b}$$

In the above set of equations, the coil side streams are discretized as separate streams inside each discrete element with the output of one stream connected to the input of the next stream. These connections across elements for coil side streams are represented by following constraints.

$$T^{i,2} - T^{i+1,1} = 0 \qquad i = 1, ...N - 1 \tag{5.5a}$$

$$H^{i,2} - H^{i+1,1} = 0 \qquad i = 1, ...N - 1 \tag{5.5b}$$

Note that the composition variables are not used in connecting constraints since they are evaluated using the complementarity based phase change equations. For a given enthalpy and temperature, the liquid and vapor composition can be uniquely calculated from the flash equations.

Initial and Boundary Conditions

The initial conditions for both tubeside and shellside temperatures are needed whereas only the outlet temperature for tubeside streams is required, since the exit temperature for shellside stream can be implicitly calculated using overall enthalpy balance.

Inlet Temperature

$$T_j^{1,1} = T_j^{in,c} \qquad j \in S_c \tag{5.6a}$$

$$T^1 = T^{in,s} \tag{5.6b}$$

Outlet Temperature

$$T_j^{N,2} = T_j^{out,c} \qquad j \in S_c \tag{5.6c}$$

5.2.2 Thermodynamics

The stream enthalpies and thermophysical properties are calculated as a function of the stream temperature and its composition. The functional form depends on the type of equation of state (EOS) model used and its associated parameters. The IDEAL gas EOS model is the simplest thermodynamics model and can be used to calculate stream state variables as follows:

For $i \in C$ and $j \in S_c \cup \{s\}$

$$H_{ij}^{vap} = H_a + H_b T_j + H_c T_j^2 + H_d T_j^3 \tag{5.7}$$

The liquid enthalpy is determined by subtracting latent enthalpy of vaporization from the vapor enthalpy.

$$H_{ij}^{liq} = H_{ij}^{vap} - \Delta H_{ij}^{vap} \tag{5.8a}$$

$$\Delta H_{ij}^{vap} = \Delta H_a (1 - T_{ij}^r)^{\Delta H_b + \Delta H_c T_{ij}^r} \tag{5.8b}$$

The mixture enthalpy is then calculated as a combination of individual liquid and vapor enthalpies.

$$F_j H_j = V_j \sum_i y_{ij} H_{ij}^{vap} + L_j \sum_i x_{ij} H_{ij}^{liq} \tag{5.9}$$

Notation

- y_{ij}, x_{ij} - liquid and vapor composition of component i in stream j
- L_j, V_j - liquid and vapor molar flow rate of stream j

Although, cubic equation of state (CEOS) thermodynamic models are more accurate and nonlinear than IDEAL gas model, the model can be easily derived with CEOS package as shown in Appendix section .6 .

5.2.3 Phase Change

The condensation (liquefaction) and evaporation process for mixed component streams can be modeled using complementarity constraints (similar to Chapter 3). The pure vapor, liquid and two-phase (mix of liquid and vapor) streams are in a state of vapor-liquid equilibrium (VLE). As the streams exchange heat between them, they can be imagined to pass a flash column with given heat duty and pressure. The flash equations for VLE for a mixed component stream are written as:

Flash Equations

For $i \in C$, $j \in S_s \cup \{s\}$

$$F_j = L_j + V_j \tag{5.10a}$$

$$z_{ij} F_j = x_{ij} L_j + y_{ij} V_j \tag{5.10b}$$

$$y_{ij} = \beta_j K_{ij}(T_j, P_j) x_{ij} \tag{5.10c}$$

$$\sum_i (y_{ij} - x_{ij}) = 0 \tag{5.10d}$$

$$\beta_j - 1 \leq sV_j \tag{5.10e}$$

$$\beta_j - 1 \geq -sL_j \tag{5.10f}$$

$$0 \leq sV_j \perp V_j \geq 0 \tag{5.10g}$$

$$0 \leq sL_j \perp L_j \geq 0 \tag{5.10h}$$

Notation

- K_{ij} - vapor-liquid equilibrium constant
- β_j, sL_j, sV_j - auxiliary variables for phase change
- z_{ij} - molar composition of component i in stream j

The formulation uses the auxiliary variables to navigate between liquid, vapor and two-phase regions. The formulation was developed by Kamath et al. [70] and has been used multiple studies since. When the stream is pure liquid ($L > 0, V = 0$) or pure vapor ($V > 0, L = 0$), one of the auxiliary variables sL and sV is non-zero positive and the VLE equation (Eq.5.10c) is relaxed using the auxiliary variable ($\beta > 1$ or $\beta < 1$). But when the stream is in two-phase ($L > 0, V > 0$), the auxiliary variables are equal to zero ($sL = sV = 0$) and the VLE equation is active with $\beta = 1$.

5.2.4 Design Equation

Design equations for the SWHX are a bit more complex than the regular shell and tube heat exchanger because of the geometrical structure and fluid flow pattern. The design equations are relations between the heat exchanger area (ΔA) and the geometric design variables of SWHX.

A SWHX has multiple design variables which are related to the heat exchange area (A_{he}) inside the exchanger. Following is the list of design variables used to design a SWHX:

Shell design variables:

- Total Length of the heat exchanger (L)
- Shell inner diameter (D_{core})

- Shell outer diameter (D_{shell})

Coil design variables:

- Coil diameter (D_{coil})
- Longitudinal pitch (p_L)
- Radial pitch (p_R)
- Coil angle (θ)
- Number of tubes per layer (N_t)

Other design variables like number of layers per stream (N_l), number of coils or turns (Nc) and tube diameters (d_{in}, d_{out}) can be assumed constant since there are discrete variables.

The design variables can be calculated using geometric relations between them. The coil diameters are related to each other as follows:

$$D^i_{coil,k} = D^i_0 + p^i_R + 2(k-1)p^i_R, \qquad i \in S_c, \;\; k = 1...N^i_l \tag{5.11a}$$

$$D^i_0 = D^{i-1}_{coil,N_l}, \qquad i \in S_c \setminus S_c[1] \tag{5.11b}$$

$$D^i_0 = D_{core}, \qquad i = S_c[1] \tag{5.11c}$$

$$D_{shell} = D^i_{coil,N_l} + p^i_R, \qquad i = S_c[NC] \tag{5.11d}$$

The D^i_0 is the reference diameter for each tube-side stream and is equal to the coil diameter of the previous layer.

The heat exchange area inside each element (ΔA) can be calculated as:

$$\Delta A_i = N^i_t \Big(\sum_k \pi d^i_{out} \frac{D^i_{coil,k}}{cos(\theta^i)} \Big) . \Big(\frac{Nc^i_k}{N} \Big), \qquad i \in S_c, j = 1...N \tag{5.12}$$

The total length of the exchanger can then be calculated from the coil diameter and the number of coil turns:

$$L = D^i_{coil,k} . tan(\theta^i) . Nc^i_k, \qquad \forall i \in S_c, k = 1....N^i_l \tag{5.13}$$

As can be seen in Eq.5.13, the number of coil turns (Nc) and the coil angle (θ) are not free and are connected to the coil diameters and the total length of the exchanger.

Overall Heat Transfer Coefficient

The overall heat transfer coefficient (H.T.C) (U) is the performance variable and affects the heat exchanger design as related to the discrete heat equations (Eq.5.4). The overall H.T.C for each pair of tube and shellside streams is calculated separately using the individual H.T.Cs. These individual coefficients depend on multiple factors: composition, flow rates, thermal properties etc. Most accurate correlations for HTCs are complex, non-smooth and proprietary. Neeraas et al. [80] have proposed correlations based on Gnielinski's method for natural gas mixtures. These correlations require stream velocities and flow area calculations which are derived below for the SWHX model.

The shellside and tubeside flow cross-section area can be calculated as:

$$Afl_c^i = N_l^i N_t^i \left(\pi \tfrac{d_{in,i}^2}{4}\right), \qquad i \in S_c \tag{5.14a}$$

$$Afl_s = \pi \left(\tfrac{D_{core}+D_{shell}}{2}\right) \textstyle\sum_{i \in S_c} N_l^i S_{ref}^i \tag{5.14b}$$

$$S_{ref}^i = \sqrt{(p_L^i/2)^2 + (p_R^i)^2} - d_{out}^i, \qquad i \in S_c \tag{5.14c}$$

The stream velocities (v) can also be calculated as ($v = \frac{\dot{V}}{Afl}$), where $\dot{V}$ is the stream volumetric flowrate equal to molar flow rate divided by molar density ($\dot{V} = \frac{L}{\rho}$).

The complete SWHX model consists of (Eq.5.4-5.14) which can be solved as a nonlinear system of equations at each node of the SWHX discretization as shown in Figure 5.3b. The pressure drop equations are omitted since they depend on more mechanical factors and are more complex equations. Neeraas et al. [80] also proposed pressure drop correlations which can be easily appended to the SWHX model.

5.3 Natural Gas Liquefaction Flowsheet

As mentioned earlier, there are multiple types of natural gas liquefaction flowsheets depending on the type of refrigerant and number of thermodynamic cycles in it. A MR based liquefaction process is shown in Figure 5.5, where the MHEX is divided into two separate

sections: warm and cold bundle.

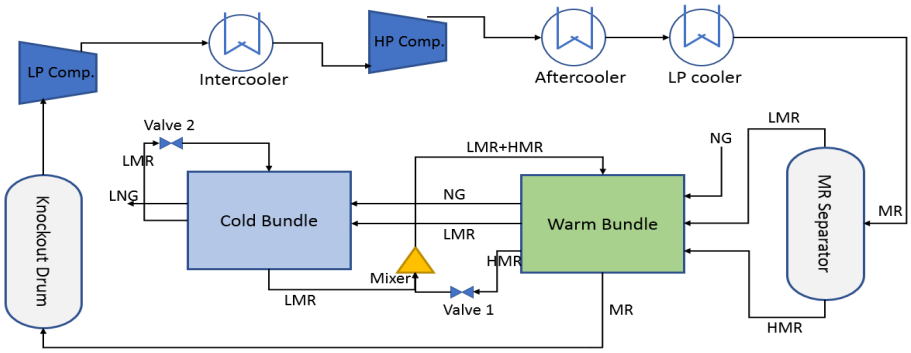


Figure 5.5: Natural Gas Liquefaction Flowsheet

Other components of the flowsheet include a MR separator, expanders and valves, a low pressure (LP) cooler and a two-stage compressor with inter-stage cooling. The natural gas stream enters the MHEX in the warm bundle as two-phase stream at $-36°C$ temperature and 40 bar pressure. It gets condensed and subcooled to $-153°C$ by the MR before being sent to a storage tank. The MR is circulated in a refrigeration cycle where the MHEX is acting as the evaporator.

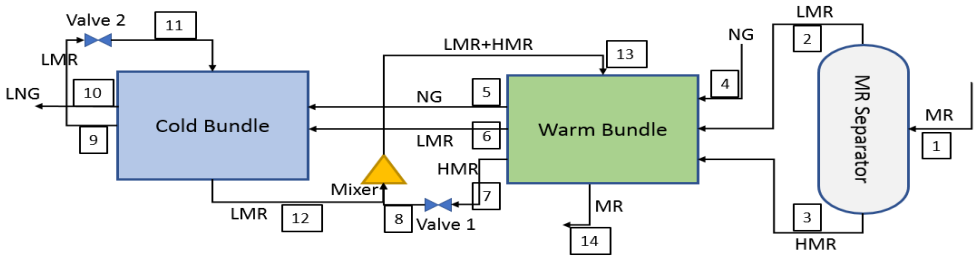


Figure 5.6: Reduced Natural Gas Liquefaction Flowsheet

Since the focus of the study is the design of MHEX, the process flowsheet is pruned as shown in Figure 5.6. Note that this doesn't limit the scope, as the compressors and inter-coolers can be easily modeled as a polytropic process.

5.3.1 Mass & Enthalpy Balance

The mass and energy balances inside different process units of the flowsheet are given below:

MR Separator

$$F_1 = F_2 + F_3 \tag{5.15a}$$

$$V_1 = V_2, \;\; L_2 = V_3 = 0 \tag{5.15b}$$

$$y_{2j} = y_{1j} \qquad \forall j \in C \tag{5.15c}$$

$$x_{3j} = x_{1j} \qquad \forall j \in C \tag{5.15d}$$

Warm Bundle, Cold Bundle and Valves

$$F_{s_{out}} = F_{s_{in}} \qquad \forall s_{in} \in S_{in}, s_{out} \in S_{out} \tag{5.16a}$$

$$z_{s_{out}j} = z_{s_{in}j} \qquad \forall s_{in} \in S_{in}, s_{out} \in S_{out} \tag{5.16b}$$

where $j \in C$, $S_{in} = \{2, 3, 6, 7, 11, 13\}$ and $S_{out} = \{6, 7, 8, 9, 12, 14\}$

Mixer

$$F_{13} z_{13j} = F_{12} z_{12j} + F_8 z_{8j} \qquad \forall j \in C \tag{5.17a}$$

$$\sum_{j \in C} z_{13j} - \sum_{j \in C} z_{12j} = 0 \tag{5.17b}$$

Enthalpy Balance

$$F_{13}(H_{14} - H_{13}) + F_2(H_6 - H_2) + F_3(H_7 - H_3) + F_4(H_5 - H_4) = 0 \tag{5.18a}$$

$$F_{11}(H_{12} - H_{11}) + F_5(H_{10} - H_5) + F_6(H_9 - H_6) = 0 \tag{5.18b}$$

$$H_8 - H_7 = 0 \tag{5.18c}$$

$$H_{11} - H_9 = 0 \tag{5.18d}$$

Miscellaneous Equations

Similar to the MHEX model, the flash equations (Eq:5.10) are used to determine the vapor-liquid molar composition at each flowsheet node. The stream enthalpies are related to the stream temperatures (same as in Eq. 5.7). For simplicity, the pressure drops are considered constant with the following values in Table 5.1.

Warm Bundle	Delta P (kPa)
Coil(s)	300
Shell	50

Cold Bundle	Delta P (kPa)
Coil(s)	200
Shell	10

Table 5.1: Pressure drop inside the MHEX

Additional constraints are added to connect the flowsheet stream temperatures and pressure. An approximation of the min function is used in Eq. 5.19h for the outlet pressure.

Connecting Equations

$$P_{2,3} - P_1 = 0 \tag{5.19a}$$

$$T_{2,3} - T_1 = 0 \tag{5.19b}$$

$$P_{6,7} - (P_2 - \Delta P_{coil}^{WB}) = 0 \tag{5.19c}$$

$$P_8 - (P_7 - \Delta P_1) = 0 \tag{5.19d}$$

$$P_9 - (P_6 - \Delta P_{coil}^{CB}) = 0 \tag{5.19e}$$

$$P_{11} - (P_9 - \Delta P_2) = 0 \tag{5.19f}$$

$$P_{12} - (P_{11} - \Delta P_{shell}^{CB}) = 0 \tag{5.19g}$$

$$P_{13} - \tilde{min}(P_{12}, P_8) = 0 \tag{5.19h}$$

$$P_{14} - (P_{13} - \Delta P_{shell}^{WB}) = 0 \tag{5.19i}$$

5.3.2 Process Constraints

There are multiple process constraints used in the flowsheet model, to ensure that the solution is feasible (no temperature cross etc.) and physically realizable.

Minimum approach temperature

A minimum approach temperature constraint between hot and cold fluid streams at terminal and internal points of both bundles is used.

$$T_i - T_j \geq EMAT \tag{5.20}$$

where $i \in S_c$ and $j \in S_s$. Default value of the EMAT parameter is 1.5 K

Superheat condition

The outgoing MR stream (S-14) should be vapor with the degree of superheat between 5 and 20 degree celsius.

$$5 \leq T_{14} - T_{14}^{DP} \leq 20 \tag{5.21}$$

MR split ratio

The HMR-LMR ratio separated by the flash vessel should be between bounds.

$$2 \leq \frac{F_{HMR}}{F_{LMR}} \leq 4 \tag{5.22}$$

5.3.3 Objective Function

The objective function of the flowsheet model is to minimize the compressor work required to recycle the MR stream. The expression for compressor work using a polytropic coefficient is described in Eq. 5.23, where the volumetric flowrate is calculated using IDEAL gas equation.

$$\Delta E_{loss} = \frac{(P_{14}\dot{v}_{14} - P_1\dot{v}_1)}{\gamma - 1} \tag{5.23a}$$

$$\dot{v}_i = \frac{V_i R T_i}{P_i} \tag{5.23b}$$

Degrees of Freedom

The multiple degrees of freedom (DOF) in this model are stated below:

- MR inlet temperature, pressure and composition
- Pressure drop across LMR and HMR valves
- Bundle break temperature (temperature between warm and cold bundles)

Variable	Initial value	Lower bound	Upper bound
T_1	-35 C	-38 C	-30 C
P_1	4600 kPa	4300 kPa	4700 kPa
$z_1(N_2)$	0.06	0	0.15
$z_1(CH_4)$	0.42	0.30	0.55
$z_1(C_2H_6)$	0.30	0.15	0.50
$z_1(C_3H_8)$	0.12	0.05	0.30
ΔP_1	3870 kPa	3700 kPa	4400 kPa
ΔP_2	3715 kPa	3700 kPa	4400 kPa
$T_{5,6,7}$	-128 C	-135 C	-120 C
$T_{8,9,11,12,13}$		-165 C	-120 C

Table 5.2: Initial values and bounds for decision variables

5.4 Initialization and Numerical Results

The optimization of natural gas liquefaction process is performed using the described MHEX DAE design model for SWHX and the NGL flowsheet model. Since the problem is a large scale optimization problem with nonlinearities and complementarity constraints, it is imperative to provide a good initial point for convergence of the NLP solver. For this reason, the problem is solved in a step-by-step procedure described below:

Step 1: Solve NGL Flowsheet model

The first step involves solving only the NGL flowsheet model separately and ignoring the heat equations inside the MHEX. This requires solving Eq.(5.15) - (5.22), enthalpy relations (Eq.5.7) and flash equations (Eq.5.10) with objective function (Eq.5.23). The min. approach temperature constraint are applied only at the terminal points of the MHEX. The complementarity constraints in the flash equations are reformulated using Reg(ϵ) (Eq. 2.55). The resultant NLP model is solved in Pyomo using CONOPT with default options.

Step 2: Solve MHEX DAE model

In the second step, the MR flowrates and composition along with inlet and outlet temperature at warm and cold bundles obtained in the previous step are used as inputs to the MHEX DAE model. The DAE model consists of Eq.(5.4)-(5.10), with min. approach temperature constraint applied at the internal nodes of the exchanger. The warm and cold bundle are each discretized with $N = 20$ discrete elements. Similar to flowsheet model, the complementarity constraints are reformulated using Reg(ϵ). For now, the design equations are omitted from the model since it requires equations for overall heat transfer coefficients which are either unavailable or unsuitable for optimization. The discretized DAE model is also modeled as a NLP in Pyomo and solved using CONOPT.

Step 3: Solve the combined NGL Flowsheet with MHEX DAE model

Final step includes concatenating the two models into a combined NGL Flowsheet and MHEX Design model. The two models are connected through the inlet conditions (temperature, MR composition, flow rates etc.) of the bundles. The solution of the flowsheet model and the DAE model from the previous steps are used as the initial value for the variables in the combined flowsheet plus DAE model and is solved using CONOPT.

This way of initializing the model aids in solving the model robustly and identifying any possible errors. The initialization method would also make it easy to include non-

ideal CEOS and design equations, as it keeps and solves the complex models separately before solving the combined NGL process model.

Results

The optimization results for the combined natural gas liquefaction process model is shown below

Variable	Value
Pressure Energy Loss	33,854.4 kW
ΔP_1, ΔP_2	3,700 & 3,840 kPa
Heat Duty (WB)	2,01,095.2kW
Heat Duty (CB)	97,375.7kW

MR Composition	Value
Nitrogen	0.046
Methane	0.404
Ethane	0.5
Propane	0.05

Table 5.3: Natural Gas Liquefaction Optimization Results

Table 5.3 shows the minimum objective value or pressure energy loss in the flowsheet equal to 33,854.4 kW. The pressure drops across the valves are 3,700 kPa and 3,840 kPa respectively. The heat duty of warm bundle is much higher than cold bundle which means that most of NG liquefaction occurs in the warm bundle ($> 99\%$). The optimal composition of MR shows that ethane and propane mole fraction values reach their upper and lower bound respectively.

The detailed results for the streams in the flowsheet is reported below in Table 5.4 with flowrates, temperature, pressure and saturation variables. The HMR and LMR flowrates show that the MR split ratio constraint Eq.5.22 is active at it lower bound.

The MR pressure at the inlet of MR separator and outlet of warm bundle reach their lower and upper bounds respectively. The MR superheat inequality is also active with difference between actual temperature and dew point temperature for S-14 is equal to 5 K. The bundle break temperature (between warm and cold bundles) for NG and LMR streams are -130.65°C and -128.65°C, respectively.

Stream label	F (kmol/s)	L (kmol/s)	V (kmol/hr)	$T(^{\circ}C)$	P(kPa)	$T_{BP}(^{\circ}C)$	$T_{DP}(^{\circ}C)$
1	10.2814	6.854	3.427	-32.15	4700	-61.35	19.9
2	3.427	0	3.427	-32.15	4700	-89.6	-32.15
3	6.854	6.854	0	-32.15	4700	-32.15	34.3
4	7.485	0.671	6.791	-36	4000	-85.3	3.43
5	7.485	7.485	0	-130.65	3700	-87.9	0.97
6	3.427	3.427	0	-128.65	4400	-92	-34.5
7	6.854	6.854	0	-128.8	4400	-35.5	30.9
8	6.854	6.854	0	-129.8	700	-105.1	-37.7
9	3.427	3.427	0	-151.5	4200	-93.7	-36.1
10	7.485	7.485	0	-153.5	3500	-89.8	-0.75
11	3.427	3.373	0.054	-155.1	360	-156	-94.8
12	3.427	2.753	0.674	-147.8	350	-156.5	-95.25
13	10.2814	9.565	0.716	-134.8	350	-141.7	-62.6
14	10.2814	0.0	10.2814	-60.8	300	-144.8	-65.8

Table 5.4: Natural Gas Liquefaction Stream Table

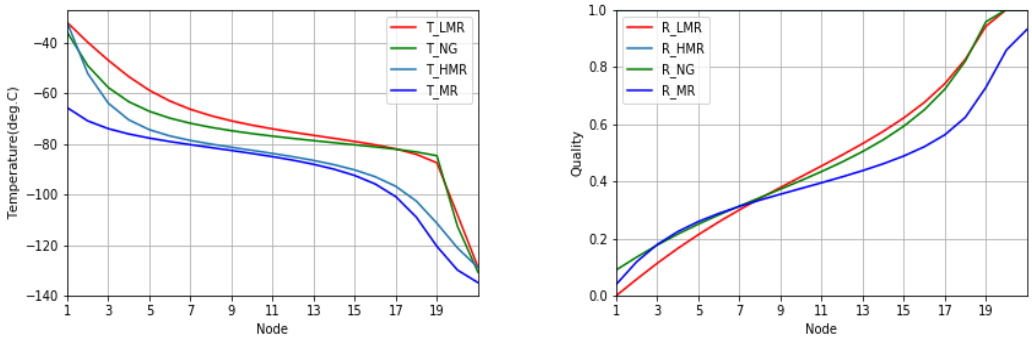
MHEX Design and Figures

In Table 5.5, the design variables (UA) are reported for all combinations of one hot and one cold stream. Figures 5.7 and 5.8 represent the variation of temperature and liquid fraction (quality) of the streams inside the warm and cold bundle.

Warm Bundle	
$UA_{NG,MR}$	6,542 kW/K
$UA_{HMR,MR}$	4,367 kW/K
$UA_{LMR,MR}$	2,662 kW/K

Cold Bundle	
$UA_{NG,LMR-C}$	1,230 kW/K
$UA_{LMR-H,LMR-C}$	348 kW/K

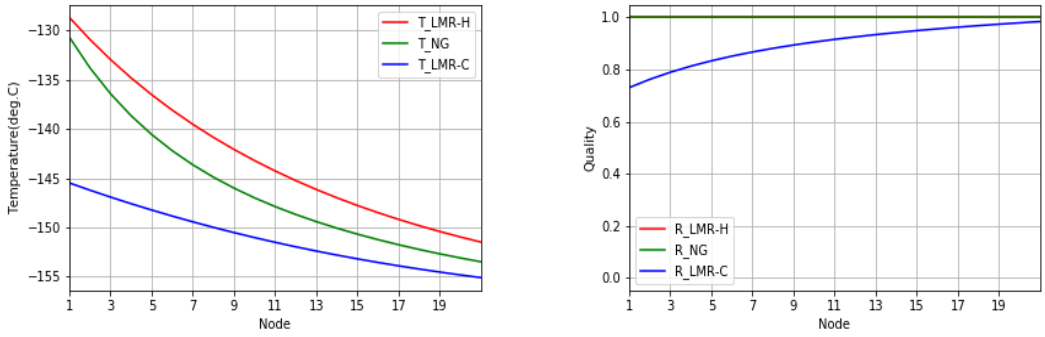
Table 5.5: MHEX Design Values



(a) Temperature profile inside Warm Bundle (b) Liquid Mole Fraction inside Warm Bundle

Figure 5.7: Temperature and Liquid Mole Fraction inside Warm Bundle

As the heat duty of warm bundle is higher, the required UA values are also higher than those in the cold bundle. Also note that the UA value between NG and MR stream is higher than between two different MR streams. This shows that most of the heat exchange in the MHEX is used to liquefy natural gas and the exchanger design is highly efficient.



(a) Temperature profile inside Cold Bundle (b) Liquid Mole Fraction inside Cold Bundle

Figure 5.8: Temperature and Liquid Mole Fraction inside Cold Bundle

The temperature plots in Figure 5.7a and 5.8a show that the min. approach temperature constraint is active at both inside and boundary points of the MHEX. The plots for liquid fraction (quality) show the phase change phenomenon occurring inside the exchanger.

Another interesting observation is the non-linear behavior in temperature plot of NG and MR streams while undergoing phase change as compared to almost linear behavior when it is exchanging only sensible heat and not being condensed or evaporated.

The flowsheet model consists of 2,453 variables and 2,465 constraints, whereas the DAE model(s) for warm (and cold bundle) consist of 13,579 (9,718) variables and 13,897 (9,937) constraints. The complete algorithm including loading, initializing and solving the models takes about 300 CPUs or 5 min.

5.5 Conclusions

This chapter describes the MHEX Design problem and its application in chemical engineering processes. An overview on different types of natural gas liquefaction process and design methods for MHEX is presented. A DAE model for SWHX-type MHEX is developed based on first principles heat equation and the phase change phenomenon is modeled using complementary based flash equations. Design equations for SWHX geometric design variables are derived with a discussion on methods to calculate overall heat transfer coefficient(s).

A natural gas liquefaction flowsheet model is developed with suitable objective function and degrees of freedom. An initialization/solution strategy is presented to robustly and simultaneously solve the combined natural gas liquefaction flowsheet with detailed MHEX design model. The results for the optimization problem are presented with detailed summary of stream variables and profiles for temperature and liquid fraction variation inside the MHEX. The results show that the inclusion of detailed MHEX design inside process flowsheet models is imperative to obtain optimal solutions which can be achieved in actual process performance.

Chapter 6

Conclusions

6.1 Summary and Contributions

The thesis attempts to address an important aspect of process modeling and optimization: incorporating equipment design in process flowsheet models. The focus is on modeling heat exchanger design and using it to optimize chemical process flowsheets. The heat exchanger design problem is studied in terms of mathematical modeling and optimization perspective. Multiple models are developed for different types of heat exchanger with detailed geometric design. Advanced optimization methods and algorithms are developed to incorporate these exchanger design models inside process flowsheet optimization models. The application of these models for optimization of refrigeration cycle, heat exchanger network and natural gas liquefaction process show the significance of incorporating the equipment design within process optimization.

In Chapter 1, the thesis describes the importance of heat exchanger design for efficient building HVAC operation and process intensification. The increasing energy demand and effects of global warming will require redesign of heat exchanger units in chemical process plants. Chapter 2 provides a brief overview on nonlinear and dynamic optimization. Basic notions of optimality and necessary (and sufficient) conditions are presented along with description on different algorithms to solve large NLPs. Parametric sensitivity and the method to derive it using KKT conditions is also presented. A brief introduction to dynamic optimization, optimality conditions and numerical discretization schemes is discussed. Lastly, complementarity constraints and MPECs are introduced with NLP based reformulations and applications in modeling hybrid dynamic systems.

In Chapter 3, the heat exchanger design problem is investigated in extensive detail by using shell and tube heat exchanger as the model type. The traditional LMTD formula with its assumptions is discussed along with the methods (Kern and Bell-Delaware) based on it. The chapter derives a novel DAE based design model derived from the PDE heat equations and the geometric design equations for exchanger design. The advantage of accounting for fluid flow dynamics and temperature dependent thermophysical property is showcased using several examples. The main contributions from this chapter are listed as follows:

- Developed a first principles based DAE model without the LMTD assumptions for heat exchanger design using exchanger geometry and numerical integration.
- Formulated complementarity based formulations for pure component phase change and included it in the DAE model to design flooded evaporator and condenser.
- Implemented a solution strategy based on the DAE method to solve examples with single shell, multiple shell and phase change exchangers and compared with LMTD solutions.

Chapter 4 focuses on the classical heat exchanger network synthesis problem with a focus on the exchanger design and its effects on the optimal network solution. The two different types of HENS models are used with hybrid strategy [58] and a newly developed trust region filter based strategy with DAE models used for detailed exchanger design. Results from both approaches are compared to other studies and shows the importance of including details about the exchanger design in the optimal HENS network. The major contributions in this chapter include:

- Implementation of the two step hybrid strategy on the simultaneous HENS with exchanger design using DAE based models and correction factors.
- Developed an integer cut based trust region strategy to incorporate detailed exchanger design models using reduced modeling and trust region filter.

- Reduced the computational times and improved the network results of previously solved examples solved in other studies.

In Chapter 5, the design of multi-stream heat exchanger is discussed in relation with natural gas liquefaction process. The heat exchange equations and flow dynamics inside a spiral wound heat exchanger are used to develop a DAE model. The phase change equations for multi-component mixture are derived by non-smooth flash equations. The DAE model is included into the natural gas liquefaction flowsheet model and solved using an step-wise initialization strategy. The main contributions of this chapter are listed as follows:

- A DAE based design model for SWHX multi-stream heat exchanger is developed with discretized heat equations and geometry based design equations.
- The phase change of multi-component natural gas and mixed refrigerant are modeled using non-smooth flash equations and then reformulated using complementarity constraints.
- A robust initialization strategy is presented to solve the overall natural gas flowsheet with detailed SWHX DAE design model.

Moreover, the work presented in this thesis is published (or under review) in following journal articles and conference papers:

Journal Articles

1. S. R. Kazi, M. Short and L. T. Biegler, "Heat exchanger network synthesis with detailed ex-changer designs: Part 1. a discretized differential algebraic equation model for shell and tube heat exchanger design," *AIChE Journal*, vol. 67, no. 1, p. e17056, 2021a

2. S. R. Kazi, M. Short, A. J. Isafiade and L. T. Biegler, "Heat exchanger network synthesis with detailed exchanger designs—2. hybrid optimization strategy for synthesis of heat exchanger networks," *AIChE Journal*, vol. 67, no. 1, p. e17057, 2021b

3. S. R. Kazi, M. Short and L. T. Biegler, "A Trust Region Framework for Heat Exchanger Network Synthesis with Detailed Individual Heat Exchanger Designs," *Computers & Chemical Engineering*, *Computer Aided Chemical Engineering*, 2021, **In Press**

Conference Papers

1. S. R. Kazi, L. T. Biegler, "Nonlinear Optimization of Detailed Heat Exchanger Models with Phase Change," *Computer Aided Chemical Engineering*, 2019, vol. 47, p. 151–156

2. S. R. Kazi, M. Short and L. T. Biegler, "Heat Exchanger Network Optimization including Detailed Heat Exchanger Models using Trust Region Method," *Computer Aided Chemical Engineering*, 2020, vol. 48, p. 1051–1056

6.2 Recommendations for Future Work

Although the thesis covers major topics regarding heat exchanger design modeling and methods to include it in process optimization models, listed below are few possible directions for future work based on the methodologies presented in this thesis.

- **Plate Heat Exchanger Design**

  Plate heat exchangers (PHE) are another type of heat exchangers used very commonly for specific purposes in process industry. There is an increase in demand of these types of exchangers in aerospace and cryogenic industry as they require less space and can conduct heat exchange with low temperature differences. A DAE based design model for PHE with improved accuracy than the current LMTD based design methods is an interesting scope for future work.
- **Work and Heat Exchanger Network Synthesis (WHENS)**

  The thesis talks about heat exchanger networks and the integration of heat exchanger design inside the optimal synthesis of HENs. Another type of network consisting of compressors and heat exchangers, called WHENS are used in heat pumps and other

process intensification systems. The water cooled chiller and natural gas liquefaction process are examples of WHEN with fixed network. There is an increasing interest in optimizing WHENS models and incorporating the DAE based models in WHENS with detailed equations for pumping and piping costs (as shown in Rathjens and Fieg [81]) is a possible line of future work [82].

- **Mass and Heat Exchanger Network Synthesis (MHENS)**

  The alternate type of networks known as MHENS consists of separators (distillation columns, absorption units etc.) and heat exchangers. The MHENS problem has been solved previously using detailed design of both mass exchanger (DE model) and heat exchangers (LMTD model) [83, 84, 85]. A similar approach as in Chapter 4 can also be applied to MHENS problem with trust region algorithm and DAE models for heat exchanger design.

- **Trust Region methods for MHEX Design**

  In the thesis, the MHEX DAE design model is solved inside the natural gas liquefaction with the IDEAL gas thermodynamics. Since natural gas and refrigerant mixture are non-ideal gases in sub-ambient and cryogenic temperatures, more complex thermodynamic models like SRK or custom package need to be used for accurate modeling of the process (Vikse et al. [86], Neeraas et al. [80]). Trust region methods presented in the thesis can be used to incorporate these CEOS based thermodynamic models as black box models [87].

- **MHEX Design in Air Separation Units (ASUs)**

  As mentioned in Chapter 5, multi-stream heat exchangers are also used in energy intensive air separation process where air is distilled as liquid nitrogen and oxygen at cryogenic temperatures (close to -190°C). The MHEX design has a strong effect on the optimization of ASU operation [88]. Therefore a possible application of the DAE design model developed in the ASU optimization model is recommended as another future work option.

Although the focus of this thesis is on heat exchanger design, the modeling tools discussed in Chapter 2 should also be studied to improve the heat exchanger design models and the solution methods to solve them. The heat exchanger design is modeled as a DAE with boundary conditions and discretized using FEM or FDM. In special cases, where the mass flow rates are low or exchangers are very small in size (like inside electronic appliances), the hyperbolic heat equation can become unstable with the mentioned numerical methods. For these instances, either the heat equation be solved as PDE using the second order diffusion term or a finite volume method (FVM) can be used with suitable delimiters for solving the hyperbolic equation system. This aspect of heat exchanger should be analyzed for general application of heat exchanger design as DAE models.

Another aspect of heat exchanger design included modeling phase change using complementarity constraints. The resultant MPECs were solved as a sequence of reformulated NLPs, but the reformulation results in creation of spurious stationary points which are not avoidable for standard NLP solvers such as IPOPT. A robust method to solve MPECs to strongly stationary or B-stationary points using the combination of NLP based reformulations, NLP sensitivity and the LPEC (Eq. 2.54) should be explored in detail to obtain global convergence to local minimum of the MPEC.

Bibliography

[1] L. Biegler, I. Grossmann, and A. Westerberg, *Systematic Methods of Chemical Process Design*. Prentice Hall, 1997.

[2] I. E. Grossmann and I. Harjunkoski, "Process systems engineering: Academic and industrial perspectives," *Computers & Chemical Engineering*, vol. 126, pp. 474–484, 2019.

[3] U.S. Energy Information Administration, "Annual Energy Outlook 2021," Tech. Rep., 2021.

[4] L. T. Biegler, *Nonlinear Programming: concepts, algorithms and applications to chemical processes*. Philadelphia, PA: SIAM, 2010.

[5] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[6] A. S. Drud, "CONOPT—A Large-Scale GRG Code," *ORSA Journal on Computing*, vol. 6, no. 2, pp. 207–216, 1994.

[7] D. Thierry and L. T. Biegler, "Dynamic real-time optimization for a CO_2 capture process," *AIChE Journal*, vol. 65, no. 7, p. e16511, 2019.

[8] C. A. J. Fletcher, *Computational Galerkin Methods*. Springer Berlin Heidelberg, 1984.

[9] D. Ralph and S. J. Wright, "Some properties of regularization and penalization schemes for MPECs," *Optimization Methods and Software*, vol. 19, no. 5, pp. 527–556, 2004.

[10] D. Ralph and H. Xu, "Implicit Smoothing and Its Application to Optimization with Piecewise Smooth Equality Constraints," vol. 124, no. 3, pp. 673–699, 2005.

[11] B. Baumrucker, J. Renfro, and L. Biegler, "MPEC problem formulations and solution strategies with chemical engineering applications," *Computers & Chemical Engineering*, vol. 32, no. 12, pp. 2903–2913, 2008.

[12] A. S. Ambekar, R. Sivakumar, N. Anantharaman, and M. Vivekenandan, "CFD simulation study of shell and tube heat exchangers with different baffle segment configurations," *Applied Thermal Engineering*, vol. 108, pp. 999–1007, 2016.

[13] J. Yang, L. Ma, J. Bock, A. M. Jacobi, and W. Liu, "A comparison of four numerical modeling approaches for enhanced shell-and-tube heat exchangers with experimental validation," *Applied Thermal Engineering*, vol. 65, no. 1-2, pp. 369–383, 2014.

[14] R. K. Sinnott, Coulson, and Richardson, *Coulson and Richardson's chemical engineering-Chemical Engineering Design*. Oxford:Pergamon, 1993, vol. 6.

[15] R. Smith, *Chemical Process Design and Integration*. John Wiley & Sons, Ltd, 2005.

[16] J. J. J. Chen, "Comments on improvements on a replacement for the logarithmic mean," *Chem. Eng. Sci.*, vol. 42, no. 10, pp. 2488–2489, 1987.

[17] W. Paterson, "A replacement for the logarithmic mean," *Chemical Engineering Science*, vol. 39, no. 11, pp. 1635–1636, 1984.

[18] F. T. Mizutani, F. L. P. Pessoa, E. M. Queiroz, S. Hauan, and I. E. Grossmann, "Mathematical Programming Model for Heat-Exchanger Network Synthesis Including Detailed Heat-Exchanger Designs. 1. Shell-and-Tube Heat-Exchanger Design," *Industrial & Engineering Chemistry Research*, vol. 42, no. 17, pp. 4009–4018, 2003.

[19] M. A. Ravagnani and J. A. Caballero, "A MINLP Model for the Rigorous Design of Shell and Tube Heat Exchangers Using the Tema Standards," *Chem. Eng. Res. Des.*, vol. 85, no. 10, pp. 1423–1435, 2007.

[20] V. C. Onishi, M. A. Ravagnani, and J. A. Caballero, "Mathematical programming model for heat exchanger design through optimization of partial objectives," *Energy Conversion and Management*, vol. 74, pp. 60–69, 2013.

[21] C. d. O. Gonçalves, A. L. H. Costa, and M. J. Bagajewicz, "Alternative Mixed-Integer Linear Programming Formulations for Shell and Tube Heat Exchanger Optimal Design," *Industrial & Engineering Chemistry Research*, vol. 56, no. 20, pp. 5970–5979, 2017.

[22] C. d. O. Goncalves, A. Costa, and M. J. Bagajewicz, "Shell and Tube Heat Exchanger Design Using Mixed-Integer Linear Programming," *AICHE Journal*, vol. 63, no. 6, pp. 1907–1922, 2017.

[23] C. O. Goncalves, A. H. Costa, and M. J. Bagajewicz, "Linear method for the design of shell and tube heat exchangers using the Bell-Delaware method," *AIChE Journal*, vol. 65, no. 8, p. e16602, 2019.

[24] A. H. Costa† and M. J. Bagajewicz, "110th Anniversary: On the Departure from Heuristics and Simplified Models toward Globally Optimal Design of Process Equipment," *Ind. Eng. Chem. Res.*, vol. 58, no. 40, pp. 18 684–18 702, 2019.

[25] M. A. S. S. Ravagnani, A. P. Silva, E. C. Biscaia, and J. A. Caballero, "Optimal Design of Shell-and-Tube Heat Exchangers Using Particle Swarm Optimization," *Industrial & Engineering Chemistry Research*, vol. 48, no. 6, pp. 2927–2935, 2009.

[26] P. D. Chaudhuri, U. M. Diwekar, and J. S. Logsdon, "An Automated Approach for the Optimal Design of Heat Exchangers," *Industrial & Engineering Chemistry Research*, vol. 36, no. 9, pp. 3685–3693, 1997.

[27] J. M. Ponce-Ortega, M. Serna-González, and A. Jiménez-Gutiérrez, "Use of genetic algorithms for the optimal design of shell-and-tube heat exchangers," *Applied Thermal Engineering*, vol. 29, no. 2-3, pp. 203–209, 2009.

[28] V. H. Iyer, S. Mahesh, R. Malpani, M. Sapre, and A. J. Kulkarni, "Adaptive Range Genetic Algorithm: A hybrid optimization approach and its application in the design and economic optimization of Shell-and-Tube Heat Exchanger," *Eng. Appl. Artif. Intell.*, vol. 85, pp. 444–461, 2019.

[29] S. G. Ravikumaur, K. N. Seetharamu, and P. A. Aswathanarayan, "Application of finite elements in heat exchangers," *Communications in Applied Numerical Methods*, vol. 2, no. July 1985, pp. 229–234, 1986.

[30] S. Ravikumaur, K. Seetharamu, and P. Aswatha Narayana, "Finite element analysis of shell and tube heat exchanger," *International Communications in Heat and Mass Transfer*, vol. 15, no. 2, pp. 151–163, 1988.

[31] C. M. Moorthy, S. G. Ravikumaur, and K. N. Seetharamu, "FEM application in phase change exchangers," *Wärme - und Stoffübertragung*, vol. 26, no. 3, pp. 137–140, 1991.

[32] D. Q. Kern, *Process heat transfer*, ser. International student edition. McGraw-Hill, 1950.

[33] K. J. Bell, "Heat exchangers: Thermal-hydraulic fundamentals and design," *Delaware Method for Shell Side Design*, 1981.

[34] W. Xiao, K. Wang, X. Jiang, X. Li, X. Wu, Z. Hao, and G. He, "Simultaneous optimization strategies for heat exchanger network synthesis and detailed shell-and-tube heat-exchanger design involving phase changes using GA/SA," *Energy*, vol. 183, pp. 1166–1177, 2019.

[35] W. E. Hart, J.-P. Watson, and D. L. Woodruff, "Pyomo: modeling and solving mathematical programs in Python," *Mathematical Programming Computation*, vol. 3, no. 3, pp. 219–260, 2011.

[36] U. V. Shenoy, *Heat exchanger network synthesis: process optimization by energy and resource analysis*. Gulf Professional Publishing, 1995.

[37] M. Short, A. J. Isafiade, D. M. Fraser, and Z. Kravanja, "Synthesis of heat exchanger networks using mathematical programming and heuristics in a two-step optimisation procedure with detailed exchanger design," *Chemical Engineering Science*, vol. 144, pp. 372–385, 2016.

[38] H. T. Broeck, "Economic Selection of Exchanger Sizes," *Industrial & Engineering Chemistry*, vol. 36, no. 1, pp. 64–67, 1944.

[39] A. H. Masso and D. F. Rudd, "The synthesis of system designs. ii. heuristic structuring," *AIChE Journal*, vol. 15, no. 1, pp. 10–17, 1969.

[40] K. C. Furman and N. V. Sahinidis, "Computational complexity of heat exchanger network synthesis," *Computers & Chemical Engineering*, vol. 25, no. 9-10, pp. 1371–1390, 2001.

[41] B. Linnhoff and E. Hindmarsh, "The pinch design method for heat exchanger networks," *Chemical Engineering Science*, vol. 38, no. 5, pp. 745–763, 1983.

[42] B. Linnhoff, "Pinch analysis : a state-of-the-art overview : Techno-economic analysis," *Chemical Engineering Research & Design*, vol. 71, pp. 503–522, 1993.

[43] J. Cerda, A. W. Westerberg, D. Mason, and B. Linnhoff, "Minimum utility usage in heat exchanger network synthesis A transportation problem," *Chemical Engineering Science*, vol. 38, no. 3, pp. 373–387, 1983.

[44] S. A. Papoulias and I. E. Grossmann, "A structural optimization approach in process synthesis—II," *Computers & Chemical Engineering*, vol. 7, no. 6, pp. 707–721, 1983.

[45] C. A. Floudas, A. R. Ciric, and I. E. Grossmann, "Automatic synthesis of optimum heat exchanger network configurations," *AIChE Journal*, vol. 32, no. 2, pp. 276–290, 1986.

[46] T. Yee and I. Grossmann, "Simultaneous optimization models for heat integration—II. Heat exchanger network synthesis," *Computers & Chemical Engineering*, vol. 14, no. 10, pp. 1165–1184, 1990.

[47] K. F. Huang, E. M. Al-mutairi, and I. Karimi, "Heat exchanger network synthesis using a stagewise superstructure with non-isothermal mixing," *Chemical Engineering Science*, vol. 73, pp. 30–43, 2012.

[48] K. F. Huang and I. A. Karimi, "Simultaneous synthesis approaches for cost-effective heat exchanger networks," *Chemical Engineering Science*, vol. 98, pp. 231–245, 2013.

[49] K. C. Furman and N. V. Sahinidis, "A Critical Review and Annotated Bibliography for Heat Exchanger Network Synthesis in the 20th Century," *Industrial & Engineering Chemistry Research*, vol. 41, no. 10, pp. 2335–2370, 2002.

[50] F. Liporace, F. Pessoa, and E. Queiroz, "Automatic Evolution of Heat Exchanger Networks with Simultaneous Heat Exchanger Design," *Brazilian Journal of Chemical Engineering*, vol. 16, no. 1, pp. 25–40, 1999.

[51] F. Liporace, F. Pessoa, and E. Quieiroz, "The influence of heat exchanger design on the synthesis of heat exchanger networks," *Brazilian Journal of Chemical Engineering*, vol. 17, no. 4-7, pp. 735–750, 2000.

[52] G. T. Polley and M. H. Panjeh Shahi, "Interfacing heat exchanger network synthesis and detailed heat exchanger design," *Trans. Inst. Chem. Eng.*, vol. 69, pp. 445–457, 1991.

[53] S. Frausto Hernandez, V. Rico Ramırez, A. Jimenez Gutierrez, and S. Hernandez Castro, "MINLP synthesis of heat exchanger networks considering pressure drop effects," *Computers & Chemical Engineering*, vol. 27, no. 8-9, pp. 1143–1152, 2003.

[54] L. Sun, B. S. Zhao, and X. L. Luo, "Optimal heat exchanger network synthesis with detailed shell and tube heat exchanger design based on superstructure model," *Chemical Engineering Transactions*, vol. 61, pp. 193–198, 2017.

[55] F. T. Mizutani, F. L. P. Pessoa, E. M. Queiroz, S. Hauan, and I. E. Grossmann, "Mathematical Programming Model for Heat-Exchanger Network Synthesis Including Detailed Heat-Exchanger Designs. 2. Network Synthesis," *Industrial & Engineering Chemistry Research*, vol. 42, no. 17, pp. 4019–4027, 2003.

[56] M. Türkay and I. E. Grossmann, "Logic-based MINLP algorithms for the optimal synthesis of process networks," *Computers & Chemical Engineering*, vol. 20, no. 8, pp. 959–978, 1996.

[57] M. Ravagnani and J. Caballero, "Optimal heat exchanger network synthesis with the detailed heat transfer equipment design," *Computers & Chemical Engineering*, vol. 31, no. 11, pp. 1432–1448, 2007.

[58] M. Short, A. J. Isafiade, D. M. Fraser, and Z. Kravanja, "Two-step hybrid approach for the synthesis of multi-period heat exchanger networks with detailed exchanger design," *Applied Thermal Engineering*, vol. 105, pp. 807–821, 2016.

[59] M. Mistry and R. Misener, "Optimising heat exchanger network synthesis using convexity properties of the logarithmic mean temperature difference," *Computers & Chemical Engineering*, vol. 94, pp. 1–17, 2016.

[60] N. M. Alexandrov, J. E. Dennis, R. M. Lewis, and V. Torczon, "A trust-region framework for managing the use of approximation models in optimization," *Structural Optimization*, vol. 15, no. 1, pp. 16–23, 1998.

[61] N. Yoshio and L. T. Biegler, "Demand-based optimization of a chlorobenzene process with high-fidelity and surrogate reactor models under trust region strategies," *AIChE Journal*, vol. 67, no. 1, 2021.

[62] J. P. Eason and L. T. Biegler, "A trust region filter method for glass box/black box optimization," *AIChE Journal*, vol. 62, no. 9, pp. 3124–3136, 2016.

[63] R. Fletcher, N. I. Gould, S. Leyffer, P. L. Toint, and A. Wächter, "Global convergence of a trust-region SQP-filter algorithm for general nonlinear programming," *SIAM Journal on Optimization*, vol. 13, no. 3, pp. 635–659, 2003.

[64] N. V. Sahinidis, "BARON: A general purpose global optimization software package," *Journal of Global Optimization*, vol. 8, no. 2, pp. 201–205, 1996.

[65] I. Grossmann, J. Viswanathan, A. Vecchietti, R. Raman, and E. Kalvelagen, "GAMS/DICOPT: A Discrete Continuous Optimization Package," *GAMS Development Corporation*, vol. 11, 2002.

[66] J. M. Ponce-Ortega, A. Jiménez-Gutiérrez, and I. E. Grossmann, "Optimal synthesis of heat exchanger networks involving isothermal process streams," *Computers & Chemical Engineering*, vol. 32, no. 8, pp. 1918–1942, 2008.

[67] A. MIT Energy Initiative, Moniz EJ, Jacoby HD, Meggs AJM, H. J. strong RC, Cohn DR, Connors SR, Deutch JM, Ejaz QJ, P. J. Kaufman GM, Kenderdine MA, O'Sullivan F, Paltsev S, and Y. Y. Perez-Arriaga I, Reilly JM, Seto C, Webster MD, "Future of Natural Gas: An Interdisciplinary MIT Study," Massachusetts Institute of Technology, Cambridge, MA, Tech. Rep., 2011.

[68] B. Austbø, S. W. Lovseth, and T. Gundersen, "Annotated bibliography-Use of optimization in LNG process design and operation," *Computers & Chemical Engineering*, vol. 71, pp. 391–414, 2014.

[69] M. A. Qyyum, K. Qadeer, and M. Lee, "Comprehensive Review of the Design Optimization of Natural Gas Liquefaction Processes: Current Status and Perspectives," *Industrial & Engineering Chemistry Research*, vol. 57, no. 17, pp. 5819–5844, 2018.

[70] R. S. Kamath, L. T. Biegler, and I. E. Grossmann, "Modeling multistream heat exchangers with and without phase changes for simultaneous optimization and heat integration," *AIChE Journal*, vol. 58, no. 1, pp. 190–204, 2012.

[71] H. Nagesh Rao and I. A. Karimi, "A superstructure-based model for multistream heat exchanger design within flow sheet optimization," *AIChE Journal*, vol. 63, no. 9, pp. 3764–3777, 2017.

[72] H. A. Watson, M. Vikse, T. Gundersen, and P. I. Barton, "Optimization of single mixed-refrigerant natural gas liquefaction processes described by nondifferentiable models," *Energy*, vol. 150, pp. 860–876, 2018.

[73] C. Tsay, R. C. Pattison, and M. Baldea, "Equation-oriented simulation and optimization of process flowsheets incorporating detailed spiral-wound multistream heat exchanger models," *AIChE Journal*, vol. 63, no. 9, pp. 3778–3789, 2017.

[74] A. O. Fredheim, "Thermal design of coil-wound LNG heat exchangers, shell-side heat transfer and pressure drop," Ph.D. dissertation, NTNU, 1994.

[75] B. O. Neeraas, "Condensation of hydrocarbon mixtures in coil-wound LNG heat exchangers. Tube-side heat transfer and pressure drop ," Ph.D. dissertation, NTNU, 1993.

[76] J. S. Fossas, "Modelling of Multistream LNG Heat Exchangers," Master's thesis, NTNU, 2011.

[77] C. Acher, Thomas and Knaup, Manuel and Göll, Stephan and Zander, Hans-Jörg and Lenz, Stefan and Kerber, "A two-dimensional CFD model for gas-liquid flows in a coil-wound heat exchanger." International Conference on Multiphase Flow, 2016.

[78] S. M. Hosseinian, A. Mostafazade Abolmaali, and H. Afshin, "An analytical method for spiral-wound heat exchanger: design and cost estimation considering temperature-dependent fluid properties," *International Journal of Numerical Methods for Heat and Fluid Flow*, vol. 31, no. 1, pp. 471–496, 2021.

[79] K. Veerabhadrappa, K. Seetharamu, C. Kembhavi, D. Dayanand, K. Vinayakaraddy, and R. Suresh, "Finite element analysis of three-fluid heat exchanger for diesel engine exhaust heat recovery system," *Applied Mechanics and Materials*, vol. 592, pp. 1607–1611, 2014.

[80] B. O. Neeraas, A. O. Fredheim, and B. Aunan, "Experimental shell-side heat transfer and pressure drop in gas flow for spiral-wound LNG heat exchanger," *International Journal of Heat and Mass Transfer*, vol. 47, no. 2, pp. 353–361, 2004.

[81] M. Rathjens and G. Fieg, "Cost-optimal heat exchanger network synthesis based on a flexible cost functions framework," *Energies*, vol. 12, no. 5, 2019.

[82] T. Gundersen and L. Naess, "The synthesis of cost optimal heat exchanger networks," *Heat Recovery Systems and CHP*, vol. 10, no. 4, pp. 301–328, 1990.

[83] O. Azeez, A. Isafiade, and D. Fraser, "Supply-based superstructure synthesis of heat and mass exchange networks," *Computers & Chemical Engineering*, vol. 56, no. 2, pp. 184–201, 2013.

[84] A. J. Isafiade and D. M. Fraser, "Interval based MINLP superstructure synthesis of combined heat and mass exchanger networks," *Chem. Eng. Res. Des.*, vol. 87, no. 11, 2009.

[85] A. J. Isafiade and M. Short, "Synthesis of Renewable Energy Integrated Combined Heat and Mass Exchange Networks," *Process Integration and Optimization for Sustainability*, vol. 3, no. 4, pp. 437–453, 2019.

[86] M. Vikse, H. A. Watson, D. Kim, P. I. Barton, and T. Gundersen, "Optimization of a dual mixed refrigerant process using a nonsmooth approach," *Energy*, vol. 196, p. 116999, 2020.

[87] J. P. Eason, J. Kang, X. Chen, and L. T. Biegler, "Surrogate Equations of State for Equation-Oriented Optimization of Polymerization Processes," vol. 44, pp. 781–786, 2018.

[88] A. W. Dowling, C. Balwani, Q. Gao, and L. T. Biegler, "Optimization of sub-ambient separation systems with embedded cubic equation of state thermodynamic models and complementarity constraints," *Computers & Chemical Engineering*, vol. 81, pp. 323–343, 2015.

Appendix

.1 Heat Exchanger Design Equations

Nomenclature

F_T	correction factor for multiple tube passes
C_p	Specific heat capacity ($kJ/kg.K$)
C	Heat Capacity (kW/K)
ρ	Stream Density (kg/m^3)
μ	Stream Viscosity ($Pa.s$)
k	Stream thermal conductivity ($W/m.K$)
Q_{duty}	Exchanger Heat Duty (kW)
A_{shell}	Heat exchange area of each shell (m^2)
ΔA	Heat exchange area inside each finite element (m^2)
NS	Number of shells
U	Overall heat transfer coefficient ($W/m^2.K$)
h_t	Tube side heat transfer coefficient ($W/m^2.K$)
h_s	Shell side heat transfer coefficient ($W/m^2.K$)
R_i	Tube inside resistance ($W/m^2.K$)
R_o	Tube outside resistance ($W/m^2.K$)
N_t	Number of tubes
N_{tp}	Number of tube-passes
N_b	Number of baffles
D_s	Shell Diameter (m)
L_t	Tube length (m)
L_b	Baffle spacing (m)
d_i	Tube inside diameter (mm)
d_o	Tube outside diameter (mm)
p_t	Tube pitch (mm)
d_e	Equivalent diameter (mm)
Re	Reynolds number
Pr	Prandtl number
Nu	Nusselt number
f_f	Tube friction factor
ΔP_t	Pressure drop inside tubes (kPa)
ΔP_s	Pressure drop inside shell (kPa)

Kern Method

F_T correction factors depend only on inlet-outlet temperatures and number of shells (NS). The correction factor calculations are given in Kern: [32]

$$R = \frac{T_{in}^{shell} - T_{out}^{shell}}{T_{out}^{tube} - T_{in}^{tube}} \tag{1a}$$

$$S = \frac{T_{out}^{tube} - T_{in}^{tube}}{T_{in}^{shell} - T_{in}^{tube}} \tag{1b}$$

if $R \neq 1$

$$P_1 = \sqrt{\frac{R^2+1}{R-1}} \tag{1c}$$

$$W = \left(\frac{1 - R \cdot S}{1 - S}\right)^{1/NS} \tag{1d}$$

$$F_T = \frac{P_1.log(W)}{log\left(\frac{1+W-P1+P1 \cdot W}{1+W+P1-P1 \cdot W}\right)} \tag{1e}$$

if $R = 1$

$$P_2 = \frac{S}{NS - NS \cdot S} \tag{1f}$$

$$F_T = \frac{\sqrt{2}P_2}{log\left(\frac{1/P_2+1/\sqrt{2}}{1/P_2-1/\sqrt{2}}\right)} \tag{1g}$$

$$LMTD_{actual} = F_T \cdot LMTD \tag{1h}$$

The number of shells should be chosen such that the correction factor (F_T) value is greater than 0.75, according to Kern [32].

After determining the number of shells and actual LMTD value, the other design variables are calculated by solving the following system of nonlinear equations Eq.2.

$$A_{shell} = \frac{Q_{duty}}{NS \cdot U \cdot LMTD_{actual}} \tag{2a}$$

$$N_t = \frac{A_{shell}}{\pi d_o L_t} \tag{2b}$$

$$v_i = \frac{V_i}{\left(\frac{\pi d_i^2}{4}\right)\left(\frac{N_t}{N_{tp}}\right)} \tag{2c}$$

$$a_o = \frac{(P_t - d_o)D_s L_b}{P_t}, \quad L_b = \frac{L_t}{N_b + 1} \tag{2d}$$

$$v_o = \frac{V_o}{a_o} \tag{2e}$$

$$Re_i = \frac{\rho v_i d_i}{\mu}, \quad Re_o = \frac{\rho v_o d_e}{\mu} \tag{2f}$$

$$Nu_i = \frac{h_i d_i}{k_i} = 0.023 Re_i^{0.8} Pr_i^{0.33} \tag{2g}$$

$$Nu_o = \frac{h_o d_e}{k_o} = 0.36 Re_o^{0.55} Pr_o^{0.33} \tag{2h}$$

$$N_t = C\left(\frac{D_s}{d_o}\right)^K \tag{2i}$$

C and K are parameters which depend on the number of tube passes.

$$\frac{1}{U} = \left(\frac{d_o}{d_i}\right)\frac{1}{h_i} + \frac{d_o log\left(\frac{d_o}{d_i}\right)}{2k} + \frac{1}{h_o} + \frac{d_o}{d_i}R_i + R_o \tag{2j}$$

Pressure drops on both tube-side and shell-side are calculated along with the design equations using the following expression:

$$\Delta P_t = 2\rho_i v_i^2 (N_{tp}) NS\left(f_f \frac{L_t}{d_i} + 1\right) \tag{3a}$$

where f_f represents the friction factor and is calculated by the Haaland equation:

$$\frac{1}{f_f^{\frac{1}{2}}} = -3.6 log_{10}\left[\left(\frac{6.9}{Re_i}\right) + \left(\frac{e}{3.7}d_i\right)^{\frac{10}{9}}\right] \tag{3b}$$

where e is the tube roughness

$$\Delta P_s = 8 j_f \left(\frac{D_s}{d_e}\right)\left(\frac{L_t}{L_b}\right)\left(\frac{\rho_o v_o^2}{2}\right)(NS) \tag{3c}$$

.2 HENS Models

Nomenclature

i	index representing the hot process streams		
j	index representing the cold process streams	CF	heat exchanger fixed
k	index representing the interval boundary number		cost ($\$/shell$)
	($k = 1, 2, ..., NOK + 1$)	CHU	hot utility cost ($\$/kW$)
NOK	number of intervals in superstructure	CCU	cold utility cost ($\$/kW$)
H	hot utilities and process streams	ACC	area cost coefficient ($\$/m^2$)
C	cold utilities and process streams	AE	area cost exponent
HPS	hot process streams	γ	upper bound for
HUT	hot utilities		heat exchanged
CPS	cold process streams	Γ	upper bound for
CUT	cold utilities		temperature difference
$T_{i,s}$	supply temperature of hot stream i (K)	$EMAT$	exchanger minimum
$T_{i,t}$	target temperature of hot stream i (K)		approach temperature (K)
$T_{j,s}$	supply temperature of cold stream j (K)	PC	pumping cost ($\$/(Pa.s)$)
$T_{j,t}$	target temperature of cold stream j (K)	NSP	Number of shell passes
$t_{i,k}$	intermediate stream temperature (K)	dt	exchanger approach
$z_{i,j,k}$	binary variables representing a match		temperature
$U_{i,j,k}$	overall heat transfer coefficient ($W/m^2.K$)		

.2.1 SWS MINLP model

The HENS model, which is a modified version of the one proposed by Yee and Grossmann [46], is presented in this section.

The first equations represent the heat balance across the hot and cold streams:

$$F_i(T_i^s - T_i^t) = \sum_{k \in int, j \in C} q_{i,j,k}, \quad i \in HPS \tag{4}$$

$$F_j\left(T_j^t - T_j^s\right) = \sum_{k \in int, i \in H} q_{i,j,k}, \quad j \in CPS \tag{5}$$

where F_i and F_j are the heat capacity flowrates of hot stream i and cold stream j, respectively. $q_{i,j,k}$ is the heat duty of matches between streams i and j in interval k. T^s and T^t are the supply and target temperatures of the corresponding streams.

The second set of equations represent the temperature changes over each individual exchanger in each interval:

$$F_i(t_{i,k} - t_{i,k+1}) = \sum_{j \in C} q_{i,j,k}, \quad i \in HPS, k \in int \tag{6}$$

$$F_j\left(t_{j,k} - t_{j,k+1}\right) = \sum_{i \in H} q_{i,j,k}, \quad j \in CPS, k \in int \tag{7}$$

where $t_{i,k}$ and $t_{j,k}$ are the intermediate stream temperatures for hot stream i and cold stream j in interval k.

Monotonicity of temperatures is strictly enforced with the following constraints:

$$t_{i,k} \geq t_{i,k+1}, \quad i \in H, k \in int \tag{8}$$

$$t_{j,k} \geq t_{j,k+1}, \quad j \in C, k \in int \tag{9}$$

The amount of heat that can be exchanged is limited to be the smaller of the heat duties that is possible between the two streams through the following inequality constraints:

$$q_{ij,k} - \Omega_{i,j,k} z_{i,j,k} \leq 0, i \in H, j \in C, k \in int \tag{10}$$

where $\Omega_{i,j,k}$ is the smaller of the two heat contents of the streams and $z_{i,j,k}$ is the binary variable that denotes the existence of a match between hot stream i and cold stream j in interval k.

Approach temperatures are calculated using the following inequality constraints:

$$dt_{i,j,k} \leq t_{i,k} - t_{j,k} + \Gamma\left(1 - z_{i,j,k}\right), i \in H, j \in C, k \in int \tag{11}$$

$$dt_{i,j,k+1} \leq t_{i,k+1} - t_{j,k+1} + \Gamma\left(1 - z_{i,j,k}\right), i \in H, j \in C, k \in int \tag{12}$$

where $dt_{i,j,k}$ is the difference between hot and cold stream temperatures at the corresponding interval boundary. The parameter Γ is introduced in order to ensure that negative approach temperatures are not possible and that when a match is excluded the inequality still holds.

In order to avoid division by zero to set a minimum approach temperature the following inequality constraint is included:

$$dt_{i,j,k} \geq \varepsilon \tag{13}$$

where ε is the minimum approach temperature

The MINLP formulation is completed by including Equations 2, 4a, 4b and 6 (of main manuscript).

After the MINLP is solved, an NLP suboptimization step is solved wherein the binary variables are fixed and the majority of constraints are the same. The purpose of this stage is to remove the isothermal mixing assumption and allow for the optimization to find the optimal splits.

.2.2 Network NLP model

The first additional equation enforces monotonicity of temperatures in the split streams with the following constraints.

$$\bar{t}^{c}_{i,j,k} \geq t_{j,k+1}, \quad i \in H, \quad j \in C, k \in int - 1 \tag{14}$$

$$t_{i,k} \geq \bar{t}^{h}_{i,j,k}, \quad i \in H, \quad j \in C, k \in int \tag{15}$$

where $\bar{t}^c_{i,j,k}$ and $\bar{t}^h_{i,j,k}$ are new variables for the intermediate temperatures of the streams following an exchanger. Mixing equations are also added:

$$F_i \cdot t_{i,k+1} = \sum_{j \in C} \bar{t}^h_{i,j,k} \cdot F^h_{i,j,k} \quad i \in H, \quad j \in C, k \in int - 1 \tag{16}$$

$$F_j \cdot t_{j,k} = \sum_{i \in H} \bar{t}^c_{i,j,k} \cdot F^c_{i,j,k} \quad i \in H, \quad j \in C, k \in int - 1 \tag{17}$$

where $F^h_{i,j,k}$ and $F^c_{i,j,k}$ are the heat capacity flowrates of each branch of the splits, which are related to F_i and F_j through the following constraints:

$$F_i = \sum_{j \in C, i \in H, k \in int} F^h_{i,j,k} \quad i \in H, \quad j \in C, k \in int - 1 \tag{18}$$

$$F_j = \sum_{j \in C, i \in H, k \in int} F^c_{i,j,k} \quad i \in H, \quad j \in C, k \in int - 1 \tag{19}$$

New energy balances are also added in addition to those already in place for cases where $z_{i,j,k} = 1$:

$$F^h_{i,j,k}(t_{i,k} - \bar{t}^h_{i,j,k}) = q_{i,j,k}, \quad i \in H, \quad j \in C, k \in int - 1 \tag{20}$$

$$F^c_{i,j,k}(\bar{t}^c_{i,j,k} - t^c_{j,k+1}) = q_{i,j,k}, \quad i \in H, \quad j \in C, k \in int - 1 \tag{21}$$

And, where $z_{i,j,k} = 0$:

$$t_{i,k} = \bar{t}^h_{i,j,k} \quad i \in H, \quad j \in C, k \in int - 1 \tag{22}$$

$$\bar{t}^c_{i,j,k} = t^c_{j,k+1} \quad i \in H, \quad j \in C, k \in int - 1 \tag{23}$$

Approach temperatures are also changed to reflect the new intermediate temperatures:

$$dt_{i,j,k} \leq t_{i,k} - \bar{t}^c_{i,j,k}, \quad i \in H, j \in C, k \in int \tag{24}$$

$$dt_{i,j,k+1} \leq \bar{t}^h_{i,j,k} - t_{j,k+1}, \quad i \in H, j \in C, k \in int \tag{25}$$

The final constraints depend on whether bypassing of exchangers is included or not. The following constraints are removed when exchanger bypassing by streams is included.

$$if \quad z_{i,j,k} = 0 \quad then \tag{26}$$

$$F^h_{i,j,k} = 0, \quad i \in H, j \in C, k \in int \tag{27}$$

$$and \qquad F^c_{i,j,k} = 0, \quad i \in H, j \in C, k \in int \tag{28}$$

Finally, we replace the implicitly calculated pressure drops with pressure drops calculated more explicitly with the corrected approximation. Note that the correction factors between the MINLP and the NLP are the same, apart from those relating to pressure drops, as they are calculated using different equations. For completeness, we also include the same objective function as shown in the MINLP.

All of the correction factors are based on the solutions obtained in the NLP suboptimization and the detailed exchanger designs, obtained using the DAE model.

.3 Derivation of Correction Factors

Once the network NLP model is solved, the resulting mass and energy balances from the network are used to solve for optimal detailed exchanger designs as described before. Once these exchangers are designed, the correction factors are updated by comparing these solutions with those obtained in the topology generation section.

$$(\tilde{P}^C_{i,j,k,p+1})_{minlp} = (\tilde{P}^C_{i,j,k,p})_{minlp} \cdot Psub^C_{i,j,k}/V^C_{i,j,k}/\Delta P^C_{i,j,k} \tag{29}$$

where $(\tilde{P}^C_{i,j,k,p})_{minlp}$ is the correction factor for pressure drop in the MINLP for the cold stream for current iteration p, $Psub^C_{i,j,k}$ is the pressure drop found in the detailed model converted to the appropriate units, $V^C_{i,j,k}$ is the volumetric flowrate of the cold stream and $\Delta P^C_{i,j,k}$ is the pressure drop in the MINLP solution. The purpose of the correction is to match the solution of the detailed model to the MINLP shortcut model and provide a linear update to "correct" the shortcut model. A similar equation is used for the hot stream.

$$(\tilde{P}^H_{i,j,k,p+1})_{minlp} = (\tilde{P}^H_{i,j,k,p})_{minlp} \cdot Psub^H_{i,j,k}/V^H_{i,j,k}/\Delta P^H_{i,j,k} \tag{30}$$

Since a different formulation is used for pressure drop in the NLP suboptimization that relates the pressure drops to heat exchanger area and volumetric flowrate, a different correction factor is required:

$$(\tilde{P}^C_{i,j,k,p+1})_{nlp} = (\tilde{P}^C_{i,j,k,p})_{nlp} \cdot Psub^C_{i,j,k}/\Delta P^C_{i,j,k} \tag{31}$$

where $(\tilde{P}^C_{i,j,k,p})_{nlp}$ is the correction factor for pressure drop in the NLP suboptimization for the cold stream for current iteration p, $Psub^C_{i,j,k}$ is the pressure drop found in the detailed model converted to the appropriate units and $\Delta P^C_{i,j,k}$ is the pressure drop in the MINLP solution. For completeness we show the equation to derive the hot stream correction factor.

$$(\tilde{P}^H_{i,j,k,p+1})_{nlp} = (\tilde{P}^H_{i,j,k,p})_{nlp} \cdot Psub^H_{i,j,k}/\Delta P^H_{i,j,k} \tag{32}$$

The remaining correction factors are calculated in a similar way to this. The correction for the number of shells, $NSP_{i,j,k}$, is directly corrected as this is a parameter in the model:

$$NSP_{i,j,k,p+1} = NS_{i,j,k}/(NSP_{i,j,k,p} \cdot NSP^{0}_{i,j,k}) \tag{33}$$

where $NSP_{i,j,k,p}$ is the correction factor for the current number of shells in the network optimization, $NS_{i,j,k}$ is the number of shells found in the detailed exchanger optimization, and $NSP^{0}_{i,j,k}$ is the initial number of shells (at iteration 1). Other important correction factors include those for heat transfer coefficients, $\tilde{h}^{H}_{i,j,k}$ and $\tilde{h}^{C}_{i,j,k}$.

$$\tilde{h}^{C}_{i,j,k,p+1} = h^{C}_{i,j,k}/(\tilde{h}^{C}_{i,j,k,p} \cdot h^{C0}_{i,j,k}) \tag{34}$$

$$\tilde{h}^{H}_{i,j,k,p+1} = h^{H}_{i,j,k}/(\tilde{h}^{H}_{i,j,k,p} \cdot h^{H0}_{i,j,k}) \tag{35}$$

where $\tilde{h}^{C}_{i,j,k,p}$ is the correction factor for the current cold stream heat transfer coefficient in the network optimization, $h^{C}_{i,j,k}$ is the heat transfer coefficient for the cold stream found in the detailed exchanger optimization, and $h^{C0}_{i,j,k}$ is the initial heat transfer coefficient for the cold stream (at iteration 1). Finally, a correction factor is also included for the heat exchanger areas.

$$\tilde{C}orF_{i,j,k,p+1} = \tilde{C}orF_{i,j,k,p} \cdot Asub_{i,j,k}/Area_{i,j,k} \tag{36}$$

where $Area_{i,j,k}$ is the area obtained in the network optimization. $Asub_{i,j,k}$ is the area obtained via the detailed unit optimization and $\tilde{C}orF_{i,j,k,p}$ is the current value for the correction factor at iteration p. Note that the values for the initial correction factors are chosen as 1, however the other initial values for parameters such as heat transfer coefficients, pressure drops and number of shells should be chosen to give values that underestimate the objective function, but are also suitably close to the true values to minimize the number of iterations required.

As mentioned in the main text, there is also a filter implemented whereby the difference between the correction factor at iteration p and iteration $p+1$ is constrained to a maximum

of $\pm 5\%$. These are then implemented into the network optimization equations for the next iteration as parameters. An example of what this procedure might look like is shown in Figure 1.

Note how in this figure, it is difficult to tell a priori which values the factors might take and also that some of the factors correct the original parameters significantly. In this example the correction factors do not decrease, however in other examples, where the match is changing duty as a result of network topology changes it is possible that these corrections increase and decrease throughout the iterations.

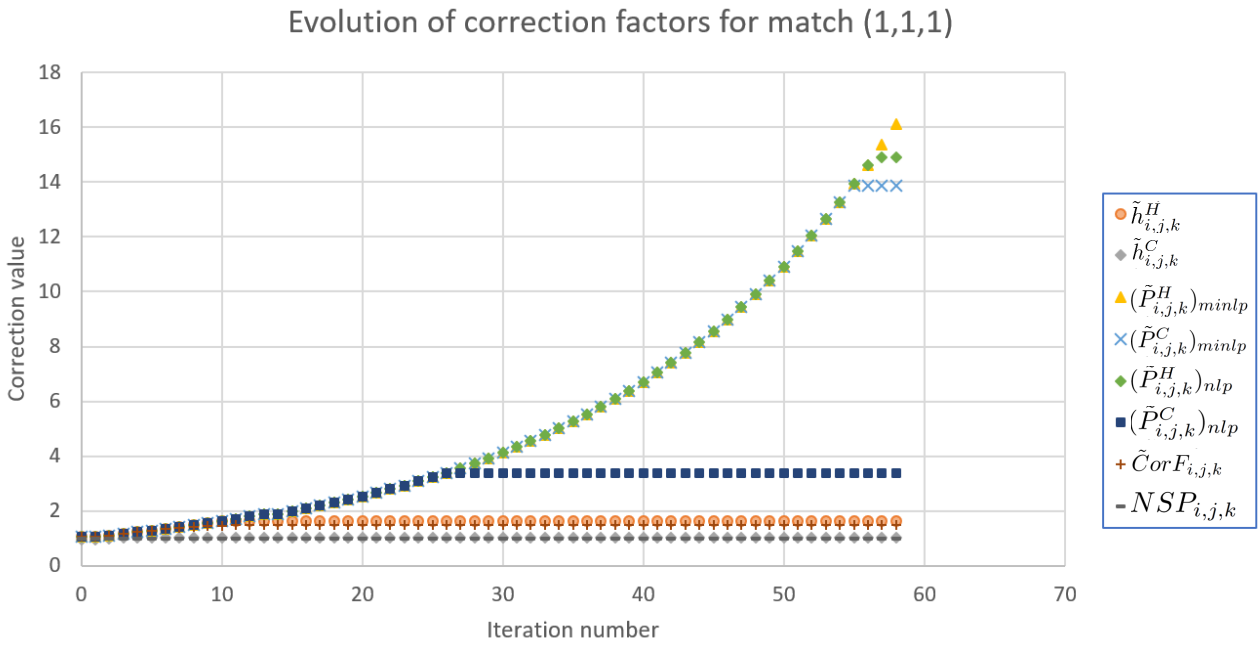


Figure 1: Demonstration of evolution of correction factors over the course of the algorithm for Example 1

.4 Trust Region Constraint and Parameters

Trust region constraints are used to bound the solution near to the approximation point such that the reduced models are a good approximation. In this study, the infinite-norm is used to model the trust region constraints.

$$||u - u_k||_\infty \leq \Delta_k \tag{37}$$

$$\text{can be reformulated as} \qquad -\Delta_k \leq u - u_k \leq \Delta_k \tag{38}$$

u,u_k are scaled variables denoting the degrees of freedom for the NLP problem

Parameters

Parameters used in the trust region strategy are described below along with their values used in this paper.

Parameter	Description	Value
γ_θ	Filter parameter for infeasibility	0.01
γ_f	Filter parameter for objective	0.01
γ_c	Trust region radius shrink factor	0.75
γ_e	Trust region radius expansion factor	1.25
γ_s	Switching condition parameter-1	2.0
κ_θ	Switching condition parameter-2	0.1
θ_{min}	Infeasibility tolerance	1e-04
θ_{init}	Initial value of infeasibility	50
Δ_{min}	Minimum value of trust region radius	1e-03
Δ_{max}	Maximum value of trust region radius	1000
η_1	Interval parameter-1 for ratio test	0.05
η_2	Interval parameter-2 for ratio test	0.20

.5 Detailed Exchanger Design Results for Example 3

The result with detailed design of each exchanger inside the heat exchanger network without bypass (Fig. 4.11a and 4.12a) for both hybrid strategy and trust region filter strategy.

Hybrid Strategy

	Exchanger 1	Exchanger 2	Exchanger 3	Exchanger 4	Exchanger 5
Area(m^2)	951.77	1041.4	739.12	387.21	611.46
Duty(kW)	12,049.16	7,018.44	16,049.2	1,484.67	6993.9
N_{tp}	4	4	4	20	6
NS	2	2	2	5	3
$D_s(m)$	1.11	1.42	1.31	0.52	0.46
N_t	2610	4284	2534	424	1118
N_b	3	1	2	6	4
$d_o(mm)$	15.88	15.88	19.05	15.88	15.88
$d_i(mm)$	13.24	13.24	15.88	13.24	13.24
$p_t(mm)$	19.84	19.84	23.81	19.84	19.84
$L(m)$	3.658	2.438	2.438	3.658	3.658
$v_s(m/s)$	0.347	0.673	0.440	0.351	0.402
$v_t(m/s)$	1.177	1.258	1.477	1.306	1.169
$h_s(W/m^2.K)$	1001.4	1441.6	1051.1	1007.9	1085.6
$h_t(W/m^2.K)$	1673.8	1765.0	1935.0	1819.3	1664.7
$U(W/m^2.K)$	472.2	563.2	505.2	486.9	489.2
$\Delta P_t(kPa)$	21.45	18.90	22.85	131.5	31.76
$\Delta P_s(kPa)$	13.65	31.87	16.24	28.73	23.71
Hot fluid allocation	Tube	Tube	Tube	Shell	Shell

Table 1: Results of E1 - E5 for Example 3 without bypasses using Hybrid Strategy

	Exchanger 6	Exchanger 7	Exchanger 8	Exchanger 9	Exchanger 10
Area(m^2)	4,982.0	203.84	220.42	569.2	1,232.53
Duty(kW)	17,521.6	2,439.3	3,833.15	8,723.97	7,042.98
N_{tp}	28	4	4	2	24
NS	7	2	2	1	4
$D_s(m)$	1.44	0.75	0.58	1.33	1.08
N_t	1464	698	604	2602	2536
N_b	4	6	6	4	2
$d_o(mm)$	25.4	19.05	15.89	19.05	15.89
$d_i(mm)$	21.18	15.89	13.24	15.89	13.24
$p_t(mm)$	31.75	23.81	19.84	23.81	19.84
$L(m)$	6.096	2.4384	3.658	3.658	2.4384
$v_s(m/s)$	0.459	0.430	0.366	0.316	0.348
$v_t(m/s)$	1.248	1.170	1.185	1.124	1.112
$h_s(W/m^2.K)$	945.2	1038.2	1032.1	876.4	1003.7
$h_t(W/m^2.K)$	1596.8	1606.2	1682.7	1555.3	1599.9
$U(W/m^2.K)$	448.7	472.1	479.8	430.9	465.4
$\Delta P_t(kPa)$	158.5	14.46	21.73	8.42	89.43
$\Delta P_s(kPa)$	84.8	20.65	14.08	7.07	20.20
Hot fluid allocation	Shell	Shell	Shell	Shell	Shell

Table 2: Results of E6 - E10 for Example 3 without bypasses using Hybrid Strategy

	Exchanger 11
Area(m^2)	1212.2
Duty(kW)	5234.38
N_{tp}	24
NS	4
$D_s(m)$	1.08
N_t	2494
N_b	2
$d_o(mm)$	15.875
$d_i(mm)$	13.24
$p_t(mm)$	19.84
$L(m)$	2.438
$v_s(m/s)$	0.351
$v_t(m/s)$	1.177
$h_s(W/m^2.K)$	1007.2
$h_t(W/m^2.K)$	1673.7
$U(W/m^2.K)$	473.5
$\Delta P_t(kPa)$	99.85
$\Delta P_s(kPa)$	20.33
Hot fluid allocation	Shell

Table 3: Results of E11 for Example 3 without bypasses using Hybrid Strategy

TRF based Strategy

	Exchanger 1	Exchanger 2	Exchanger 3	Exchanger 4	Exchanger 5
Area(m^2)	1840.6	971.7	411.9	148.3	1704.4
Duty(kW)	12,025	7,018	16,049	1,484	6,500
N_{tp}	4	2	2	2	4
NS	3	2	1	1	9
$D_s(m)$	1.44	1.48	1.35	0.88	0.77
N_t	2804	3330	1412	1016	1038
N_b	3	3	3	6	4
$d_o(mm)$	19.05	19.05	25.4	19.05	15.88
$d_i(mm)$	15.88	15.88	21.2	15.88	13.24
$p_t(mm)$	23.81	23.81	31.7	23.81	19.84
$L(m)$	3.658	2.438	3.658	2.438	3.658
$v_s(m/s)$	0.624	1.25	0.662	0.31	0.398
$v_t(m/s)$	1.52	1.12	1.49	1.34	1.258
$h_s(W/m^2.K)$	1274	1868.4	1156	868.8	1079.7
$h_t(W/m^2.K)$	1982	1554.7	1840	1793.7	1765.5
$U(W/m^2.K)$	556	971.7	516	448.9	498
$\Delta P_t(kPa)$	91.4	13.4	11.9	9.5	219.9
$\Delta P_s(kPa)$	71.7	197.6	19.0	6.4	70.4
Hot fluid allocation	Tube	Tube	Tube	Shell	Shell

Table 4: Results of E1 - E5 for Example 3 without bypasses using TRF Strategy

	Exchanger 6	Exchanger 7	Exchanger 8	Exchanger 9	Exchanger 10
Area(m^2)	1428	1260.4	44.6	203.5	697
Duty(kW)	5,006	12,515	1,124	2,012	3,136
N_{tp}	4	2	2	2	4
NS	2	3	1	2	8
$D_s(m)$	1.44	1.02	0.52	0.67	0.55
N_t	1468	1440	306	838	478
N_b	4	4	4	4	6
$d_o(mm)$	25.4	19.05	19.05	15.89	15.89
$d_i(mm)$	21.18	15.89	15.89	13.24	13.24
$p_t(mm)$	31.75	23.81	23.81	19.84	19.84
$L(m)$	6.096	4.877	2.438	2.438	3.658
$v_s(m/s)$	0.448	0.43	0.441	0.341	0.391
$v_t(m/s)$	1.244	1.13	1.29	1.12	1.319
$h_s(W/m^2.K)$	944	1041.8	1052.4	991.4	1069.6
$h_t(W/m^2.K)$	1592	1559.5	1736.7	1611.2	1834
$U(W/m^2.K)$	448	467.9	488	464	502.1
$\Delta P_t(kPa)$	45.0	30.7	8.76	15.2	214.5
$\Delta P_s(kPa)$	24.2	30.7	5.4	10.0	60.1
Hot fluid allocation	Shell	Tube	Shell	Shell	Shell

Table 5: Results of E6 - E10 for Example 3 without bypasses using TRF Strategy

	Exchanger 11	Exchanger 12	Exchanger 13
Area(m^2)	620.1	1072.8	216.7
Duty(kW)	9,546	10,465	990
N_{tp}	2	6	2
NS	2	6	1
$D_s(m)$	1.43	0.98	0.82
N_t	1594	1226	446
N_b	5	3	7
$d_o(mm)$	25.4	19.05	25.4
$d_i(mm)$	21.1	15.89	21.1
$p_t(mm)$	31.75	23.8	31.75
$L(m)$	2.438	2.438	6.096
$v_s(m/s)$	0.528	0.52	0.49
$v_t(m/s)$	1.181	1.79	1.15
$h_s(W/m^2.K)$	1021.5	1414.2	981.7
$h_t(W/m^2.K)$	1527.9	2254.6	1495.4
$U(W/m^2.K)$	458	550.6	446.3
$\Delta P_t(kPa)$	12.38	299.65	9.65
$\Delta P_s(kPa)$	38.3	65.84	12.67
Hot fluid allocation	Shell	Shell	Shell

Table 6: Results of E11 - E13 for Example 3 without bypasses using TRF Strategy

.6 Cubic Equation of State for SWHX model

The more suitable thermodynamic model for hydrocarbon mixtures are cubic equation of state (CEOS) models - Soave Redlich Kwong (SRK) or Peng-Robinson (PR) method. Both models can be represented using the cubic compressibility factor equations as follows:

$$Z^3 - (1 + B - uB)Z^2 + (A + wB^2 - uB - uB^2)Z - AB - wB - wB^3 = 0 \tag{39}$$

where the parameters u and w decide the type of model as shown below.

Model	u	w
SRK	1	0
Peng-Robinson	2	-1

Table 7: Parameters for Cubic Equation of State-I

A and B are EOS parameters which are calculated using following equations.

$$a_i = K_1 \frac{R^2 T_c^2}{P_c} \tag{40a}$$

$$b_i = K_2 \frac{RT_c}{P_c} \tag{40b}$$

$$a_m = \sum_{i \in C} \sum_{j \in C} x_i x_j \sqrt{a_i a_j}(1 - k_{i,j}) \tag{40c}$$

$$b_m = \sum_{i \in C} x_i b_i \tag{40d}$$

$$\alpha = (1 + \kappa(1 - (T/T_c)^{1/2})^2 \tag{40e}$$

$$\kappa = c_1 + c_2\omega + c_3\omega^2 \tag{40f}$$

$$A = \frac{\alpha a_m P}{R^2 T^2} \tag{40g}$$

$$B = \frac{b_m P}{RT} \tag{40h}$$

Model	K_1	K_2	c_1	c_2	c_3
SRK	0.42747	0.08664	0.48	1.574	-0.176
Peng-Robinson	0.45724	0.0778	0.37464	1.54226	-0.26992

Table 8: Parameters for Cubic Equation of State-II

In non-ideal mixtures, the fugacities also depend on the liquid and vapor phase compositions (x_{ij}, y_{ij}) along with stream temperature and pressure. The individual fugacity coefficients are given by the equation below.

$$ln(\phi_j^k) = \frac{b_j}{b_m}(Z^k - 1) - ln(Z^k - B) + \frac{A}{B\sqrt{u^2 - 4w}}\left(\frac{b_j}{b_m} - \delta_j\right) ln\left[\frac{2Z^k + B(u + \sqrt{u^2 - 4w})}{2Z^k + B(u - \sqrt{u^2 - 4w})}\right] \tag{41a}$$

$$\delta_j = \frac{2\sqrt{a_j}}{a_m}\sum_{i \in C} x_i\sqrt{a_i}(1 - k_{i,j}) \tag{41b}$$

$$K_j = \frac{\phi_j^L}{\phi_j^V} \tag{41c}$$

for $k = L, V$ and where $b_j, b_m, , \delta_j, k_{i,j}$ are parameters linked to the EOS model. Z^L and Z^V are compressibility factors which are to be calculated using the cubic equation (Eq. 39).

Z^V and Z^L are the largest and smallest root of the cubic equation respectively. To separately calculate the two compressibility factors, additional constraints are added which are characteristics of the largest and smallest root of a cubic equation.

$$f'(Z) = 3Z^2 - 2(1 + B - uB)Z + (A + wB^2 - uB - uB^2) \geq 0 \tag{42a}$$

$$f''(Z^L) = 6Z^L - 2(1 + B - uB) \leq Ms^L \tag{42b}$$

$$f''(Z^V) = 6Z^V - 2(1 + B - uB) \geq -Ms^V \tag{42c}$$

The non-ideal enthalpy is calculated using residual equations:

$$H - H^{id} = \left(a_m - T\frac{\partial a_m}{\partial T}\right)\frac{1}{b_m\sqrt{u^2 - 4w}} ln\left[\frac{2Z^k + B(u + \sqrt{u^2 - 4w})}{2Z^k + B(u - \sqrt{u^2 - 4w})}\right] + RT(Z^k - 1) \tag{43a}$$

$$\frac{\partial a_m}{\partial T} = -\frac{R}{2}\sqrt{\frac{K_1}{T}}\sum_{i \in C}\sum_{j \in C} x_i x_j(1 - k_{i,j})\left[\kappa_j\sqrt{\frac{a_i T_j^c}{P_j^c}} + \kappa_i\sqrt{\frac{a_j T_i^c}{P_i^c}}\right] \tag{43b}$$