INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

University of Alberta

INTEGRATED PROCESS DESIGN AND CONTROL OF CHEMICAL PROCESSES

by

Jianye Zhu 

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Doctor of Philosophy

in

Chemical Engineering

Department of Chemical and Materials Engineering

Edmonton, Alberta

Fall 1998

University of Alberta

Library Release Form

Name of Author: Jianye Zhu

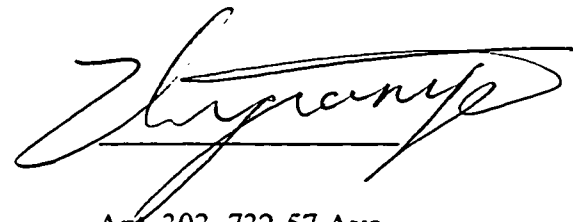
Title of Thesis: Integrated Process Design and Control of Chemical Processes

Degree: Doctor of Philosophy

Year this Degree Granted: 1998

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly, or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.



Apt. 303, 732-57 Ave.

Calgary, Alberta

Canada T2V 0H4

Sept. 28, 1998

Date

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled Integrated Process Design and Control of Chemical Processes submitted by Jianye Zhu in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Chemical Engineering.

Dr. M. Rao
Professor, Co-Supervisor

Dr. K. T. Chuang
Professor, Co-Supervisor

Dr. J.F. Forbes
Assistant Professor

Dr. R. Hayes
Professor

Dr. R. Toogood
Associate Professor

September 24, 1998

Dr. Jules Thibault, Professor
External Examiner

ABSTRACT

This thesis consists of two parts: integrated process design and plant-wide control, and simplification of heat exchanger networks to achieve the minimum number of units and the minimum energy penalty. The second part is intended to provide a useful technique that can be applied to the integrated design and control of heat exchanger networks based on evolutionary methods.

The main contributions made in the first part are:

(a) A simple and reliable method is developed to find excess control objectives from individual maximum dilation submatrices by exploring properties of maximum dilation submatrices. Rules for finding additional excess control objectives from the combinations of different maximum dilation submatrices are also proposed.

(b) A systematic method is presented to determine new manipulated variables that need to be introduced by plant redesign for restoring the controllability of the plant.

(c) A hierarchical approach for integration of process design and plant-wide control is proposed based on Douglas' hierarchical procedure of process design. Each subsystem is designed separately, and only its new control objectives introduced at the current level are considered. If local control is used, the controllability test of the composite system needs to be made after all subsystems are controllable. Process designs of the next level will not be carried out until systems of the current level are all controllable.

(d) As a case study, the integrated process design and control of the process of toluene hydrodealkylation for producing benzene is presented.

The second part includes the following contributions:

(a) It is shown that to achieve small energy penalties, split streams need to be treated separately in identifying independent loops. Consequently, a new equation is derived to determine the number of independent loops of heat exchanger networks.

(b) Well-defined procedures are proposed to find a maximum set of independent loops, to generate all dependent loops by loop additions, and to locate downstream paths between disturbances and controlled variables. An alternative representation of heat exchanger networks is presented to facilitate loop identification.

(c) A systematic method is developed to determine the best units for breaking given loops. The method is reliable and efficient, because it considers effects of both loads of units and structures of heat exchanger networks on energy penalties, but does not need calculations of accurate energy penalties of all units.

To my parents and

to Jinhong

Acknowledgements

I would like to thank my co-supervisors Dr. K. T. Chuang and Dr. M. Rao for their enthusiastic supervision, support and guidance leading to this thesis.

I would also like to thank Dr. Z. Han, a visiting professor from Shanghai Jiaotong University, for many useful discussions and his contributions to this thesis, and Dr. X. Sun of the Intelligence Engineering Laboratory for his valuable comments on Chapter 2 and his assistance in preparing this thesis.

Thanks are also due to Dr. X. Zhao of DB Robinson Research Ltd. for his useful comments on the design of heat exchanger networks, and Dr. E. Feng of Celanese Canada Inc. for his reviews of the earlier versions of Chapters 2 and 3. Dr. W. Lu and Dr. J. Jezowski of Hyprotech Ltd. also deserve thanks for helpful discussions on Chapter 3 and suggestions about Chapter 6, respectively.

I am also grateful to Drs. J.F. Forbes, R.E. Hayes, J. Thibault and R. Toogood for their careful reviews of my thesis.

Finally, I would like to express my gratitude to my mother, my wife, my brothers and sisters for their constant encouragement, love and support throughout the years of my Ph.D. program.

Table of Contents

List of Tables

List of Figures

CHAPTER 1. INTRODUCTION

1.1. Influences of Process Design on Process Control

Historically, and probably still commonly today, process and control system designs are carried out in separated phases. That is, the process is first designed and then the control system is added (McAvoy, 1987; Fisher *et al.*, 1988a).

The lack of integration between process design and control can result in a design that satisfies the objective and requirement of operation from a steady-state point of view, but does not possess the satisfactory dynamic performance (Elliott and Luyben, 1996; Bahri *et al.*, 1997). This is because process design not only determines the steady-state performance of the process, but also determines its dynamic performance. We can not expect that control systems will always achieve good dynamic performance. In fact, the capabilities of a control system are actually limited by the inherent properties of processes. One simple example to explain this idea is that "the acceleration of a family sedan will never equal that of a Formula I race car regardless of the quality of the fuel injection control system" (Morari, 1992).

The effects of process design on control have been discussed for a long time. As far back as in 1943, Zieger and Nichols devoted an entire paper to the topic of controllability and realized that the controller and process should be considered simultaneously. In recent years, more and more attentions are being paid to the integration of process design and control. The several trends of modern chemical processes make it necessary to consider control issues at the early stages of process design (Bouwens *et al.*, 1992):

(i) "The growing need to reduce capital and utility costs and to improve product purity necessitates a high level of heat and mass integration. An increasing level of integration implies increasing integration of control loops. Process controllability analysis aims at minimizing these interactions, either by establishing the best control strategy for a given design (Control Strategy Design), or by altering the design in order to improve controllability."

(ii) "Plant operability has to be maintained under non-steady state conditions, e.g. start up and shut down, switch over to different operating conditions and situations resulting from process disturbances. A simultaneous design and control study might, for instance, reveal that design changes are necessary in order to achieve or maintain control under various circumstances."

(iii) "There is a trend towards fully automated plants. In this respect, an ideal control strategy might consist of only one master control loop being the desired production of end product, whereby all other control loops are slave loops of the master control."

So far, many papers on the integration of process design and control have been published. Morari (1992) and Perkins (1989) have extensively reviewed many of the important topics and previous efforts concerning the analysis of the interactions between design and control. The work on the integration of process design and control can be classified into two categories: controllability analysis and evaluation, and the incorporation of the concept of controllability in the early stages of process design. This thesis only deals with the latter category. Some published work in that category is reviewed in the following section.

1.2. Integration of Process Design and Control

Various approaches to the incorporation of controllability requirement into the stages of process design have been developed. The approaches can be broadly classified into two groups: algorithmic and systematic evolutionary methods.

An algorithmic method first establishes an optimization problem (single- or multi-objectives) for a superstructure (Papalexandri and Pistikopoulos, 1996) which is constructed for a process, and then solves the optimization problem to simplify the superstructure and generate the final design. The established optimization problem includes expressions for economic and controllability objectives in terms of unknown design variables. The trade-off between economics and controllability requirements is automatically achieved by solving the optimization problem.

Systematic evolutionary methods apply engineering design procedures and heuristic rules to generate initial designs based upon targets (economics and controllability) which are set prior to the design phase. The initial designs are then evolved to achieve the trade-off between economics and controllability requirements.

Brief comparisons between algorithmic and systematic evolutionary methods are made here. Algorithmic methods can easily take account of design constraints and the trade-off between economics and controllability requirements simultaneously. Automatic design can be easily implemented by algorithmic methods. But the problem size which can be solved by algorithmic methods is limited and global optimal solutions are difficult to obtain (local optimal solutions are often resulted). Systematic evolutionary methods make full use of physical insights and the skills of engineers in an interactive design

environment. They are easy to comprehend and implement. But systematic evolutionary methods may fall into topology traps caused by poor initial designs, and are easy to get solutions just close to local optimal ones (Gundersen *et al.*, 1991).

1.2.1. Complete plants

So far, most of methods are developed for the integrated process design and control of specific processes (Luyben and Floudas, 1994a,b; Russo and Bequette, 1995). Very few papers have been published about the integrated process design and control of complete plants. This is because that the design of a specific process is more well-defined than the design of a complete plant, current design and implementation of controllers are focused on the control of individual units, and most of controllability measurements are developed for specific process units (Fisher *et al.*, 1988). The basic assumption behind this approach is that "if each unit is properly controlled, the control of the total plant will be satisfactory" (Fisher *et al.*, 1988a).

In theory, the algorithmic methods developed by Bahri *et al.* (1997) and Mohideen *et al.* (1997) can be applied to the integrated process design and plant-wide control. But, the two methods rely on a given set of process design alternatives (process superstructure). For a complete plant, its superstructure can be very complicated. To solve such a large size problem, it is not an easy task.

A series of papers by Fisher *et al.* (1988a, b, c) presented systematic methods for the integration between design and control of complete plants based on Douglas' hierarchical decision procedure for conceptual process design (1985). They stated that at the preliminary stage of a process design, the optimum steady-state designs of various

process alternatives are often uncontrollable and not operable. The basic idea of their methods is that for each level of the hierarchical process design procedure rather than for complete flow sheet, the controllability evaluation, operability examination and selection of controlled variables are performed, and design modifications are made if necessary. Various rules are proposed to implement their methods.

If a process is found uncontrollable, the process controllability can be restored by (i) modifying the flow sheet to include more manipulated variables; or (ii) modifying the design so that some of the process constraints never become active over the complete (reasonable) range of disturbances. Operability problems can be overcome by an appropriate amount of flexibility or by developing the alternative operating policy with the smallest cost penalty.

Barton *et al.* (1992) incorporated operability measures into process design. They also used the hierarchical procedure of Douglas (1985) as the framework of their method. At the decision level of input-output structure of the flowsheet, external disturbances that are expected to affect the process are identified, and their magnitude and time-scales estimated. At other subsequent decision levels, the most economically profitable operating point is first calculated by solving steady-state optimization problems, and the identified disturbances are classified into slow and fast disturbances based on their time scales. Then, flexibility and controllability analyses are carried out.

The flexibility analyses decide if slow disturbances are economically significant by considering the effects of the disturbances on the change of optimum operating profit. For the economically significant slow disturbances, the process flowsheet is reoptimised

at the lower and upper bounds of the disturbances. The economically insignificant slow disturbances are omitted from the flexibility analyses, but treated in the same way as the fast disturbance in the controllability analyses.

The controllability analyses include (a) open-loop stability, (b) output selection, (c) open-loop economic penalty analysis, (d) input selection/RGA analysis, (e) right-half-plane zeros and (f) minimum condition number. If the process is found to be open-loop unstable, then a redesign of the process or a change of operating point needs to be made. Other analyses are made to ensure that all constraints can be satisfied and good regulatory control of the process can be achieved.

1.2.2. Heat exchanger networks (HENs)

Integrated process design and control of HENs is one of the research areas which is paid the most attentions as the problem is characterized by a reasonable degree of complexity.

Georgiou and Floudas (1990) proposed a systematic framework for the simultaneous consideration of process synthesis and minimization of disturbance propagation in HENs based on mathematical programming techniques. They incorporated the conditions of the disturbance rejection into the superstructure of HENs. By solving the MINLP problem, a HEN which features optimal economical objective and disturbance rejection can be generated.

Papalexandri and Pistikopoulos (1994a,b) presented methods for the synthesis and/or retrofit of operable HENs by formulating a MINLP problem. The objective of the

optimization problem is the minimization of total annualized cost (operating and investment). Process design requirements, flexibility, total disturbance rejection, dynamic and control structure requirements are treated as constraints of the optimization problem.

As to systematic evolutionary methods, Linnhoff and Kotjabasakis (1986) and Kotjabasakis and Linnhoff (1986) used the concept of downstream path and sensitivity tables for the integrated design of HENs. They found that for HENs, a disturbance (D) can affect a controlled variable (C) only if there is a path between D and C, and if the path is completely downstream. There are three alternatives to protect a controlled variable from being affected by a disturbance: (i) breaking downstream paths, or (ii) inserting upstream elements, or (iii) adding by-passes. To achieve the trade-off between energy, capital and controllability, sensitivity tables are adopted in their work.

Calandranis and Stephanopoulos (1986 and 1988) addressed uncertain network parameters as disturbances and examined the influence of the structure of a HEN on its operability and the design of its control system. Their approach is mostly designer-driven, whereas various guidelines are provided to reduce the problem dimensionality and to handle the nonconvexities.

Huang and Fan (1992) proposed a distributed strategy for designing HENs and mass exchanger networks (MENs) which are cost-effective and highly controllable in terms of disturbance rejection. They constructed a controllability assessment table based on the classification of process information such as intensity of disturbances, control precision

and disturbance propagation, and used it to evaluate the structural controllability of HENs and MENs. Various rules are proposed to achieve the design objectives.

1.3. Structural Modeling of Chemical Processes

The hierarchical design method proposed by Douglas (1985) provides a framework for the conceptual design of a complete plant. At the early stages of this method, only the structural information of the process can be obtained. Thus, as shown below, structural models are suitable for the integrated design and control of complete plants based on the Douglas' method. In this section, some concepts and definitions of structural modeling of chemical processes are reviewed.

For a linear system with the following state differential equation:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \tag{1-1}$$

where, $\boldsymbol{x} \in \boldsymbol{R}^n$, $\boldsymbol{u} \in \boldsymbol{R}^m$, and $\boldsymbol{A}$, $\boldsymbol{B}$ are compatible matrices.

The Kalman's criterion for the controllability of system (1-1) is based on the following rank test:

$$rank(\boldsymbol{B}, \boldsymbol{AB}, \boldsymbol{A}^2\boldsymbol{B}, \cdots, \boldsymbol{A}^{n-1}\boldsymbol{B}) = n \tag{1-2}$$

The above rank test mainly depends on the numerical values of entries of $\boldsymbol{A}$ and $\boldsymbol{B}$. For the same systems, different selections of the parameter values could give totally different conclusions about the controllability of the systems, and also an arbitrarily small variation in some of the parameters could make an actually controllable system uncontrollable or vice versa. The accidental selection of the parameter values often

occurs in reality because precise values of some of the system parameters are hard to know with the exceptions of zeros that are fixed by absence of physical connections between different units (Morari and Stephanopoulos, 1980; Georgiou and Floudas, 1989). Therefore, the invariant structural information of dynamic systems could be employed to overcome the limitation of its numerical information. Structural matrices of systems are used to express such invariant structural information. A *structural matrix* is the one whose entries are either fixed to zero values or arbitrary (non-zero) elements which are expressed by "x".

Different from numerical models, structural models do not need detailed expressions of relations among variables but only require information related to whether a variable is involved in a particular equation of the system or not. The non-existence of a variable in the equation is denoted by a zero entry in the structural matrix while the existence of the variable is denoted by an arbitrary entry.

The advantage of structural models is that they only need very limited information about processes. Such information can be available even before the details of processes are developed. Thus, structural models can be employed at the early stages of conceptual process designs.

Although structural models cannot give quantitative conclusions, they can be used in prescreening design alternatives. If a process is uncontrollable in structural sense, the process will also be uncontrollable in numerical sense. But, if a process is controllable in structural sense, the process may or may not be controllable in numerical sense depending on the values of the process parameters (Georgiou and Floudas, 1989).

Structural models have already received wide applications in various areas. For example, they have been applied to:

(a) generate all feasible control configurations (Morari and Stephanopoulos, 1980; Johnston *et al.*, 1985; Georgiou and Floudas, 1989);

(b) diagnose difficult control problems at the early stages of process design (Papalexandri and Pistikopoulos, 1994a);

(c) evaluate control configurations (Daoutidis and Kravaris, 1992).

1.4. Overview of Thesis

The main objectives of this thesis are two fold: to study (a) integrated process design and plant-wide control, and (b) integrated design and control of a specific process. The design of a complete plant generates the general conceptual framework for the complete processes, but it does not consider the design of individual processes. Therefore, special methods are required for the designs of specific processes. As an example, design of HENs is addressed.

This thesis includes five chapters that mainly cover three areas: control system synthesis, integration of process design and control, and HEN design. The work includes both theoretical and application studies. While focusing on the main theme of the thesis, each chapter deals with one topic and may be read independently.

The objective of integrated process design and plant-wide control is generating structurally controllable processes at the stages of conceptual designs. To design a controllable system, its manipulated variables and control objectives should satisfy some

criteria. When such criteria are not satisfied, it means some control objectives of the control system cannot be maintained or new manipulated variables should be introduced. Thus, systematic methods are necessary to design feasible control systems. Chapter 2 deals with how to systematically find excess control objectives and introduce new manipulated variables based on the concept of structural controllability. The developed methods provide effective techniques for the integrated process design and plant-wide control.

In Chapter 3, the incorporation of structural controllability into each design level is discussed. In the first part of this chapter, a systematic framework for the integration of process design and plant-wide control is proposed, which considers not only controllability requirements (structurally state or functional controllability) but also the requirement of control system implementations (local or nonlocal control). The structural controllability of composite systems is also addressed. In the second part, the proposed procedures are applied to the design of the process of hydrodealkylation of toluene for producing benzene.

The other three chapters of this thesis are focused on the simplification of HENs because the integrated process design and control of HENs by evolutionary methods needs to simplify HENs in order to achieve certain objectives (e.g., minimum energy cost and disturbance rejection). In simplifying HENs, the concept of loops is widely applied.

In Chapter 4, based on graphic theory, a method is proposed to locate a maximum set of independent loops. To effectively implement the method and account for the nonlinear property of HENs, an alternative representation of HENs is presented. When determining

independent loops of HENs with split streams, the current practice often ignores split streams. The work in this chapter discusses the necessity of treating the split stream as separated ones in determining independent loops.

Chapter 5 gives detailed procedures to find independent loops based on the method presented in the previous chapter. From a maximum set of independent loops, dependent loops can be obtained by loop additions. Thus, procedures for such loop additions are also developed. The concept of downstream paths is useful when examining the disturbance propagation in HENs. By slightly modifying the procedure of constructing a maximum tree, the downstream paths can be easily found.

In Chapter 6, a new method is proposed to simplify HENs by breaking loops. The objective which needs to be achieved is the minimum energy penalty. It is known that the energy penalties incurred by deleting units depend not only on the loads of all units but also on the structures of the HENs. The proposed method considers these two effects but does not need accurate calculations of energy penalties of all units, so that it is reliable and efficient.

1.5. Literature Cited

Bahri, P.A., J.A. Bandoni, and J.A. Romagnoli, "Integrated Flexibility and Controllability Analysis in Design of Chemical Processes," *AIChE J.*, **43**, 997 (1997).

Barton, G.W., M.A. Padley, and J.D. Perkins, "Incorporating Operability Measures Into the Process Synthesis Stage of Design," in *IFAC Workshop on Interactions between Process Design and Process Control*, J.D. Perkins, eds., London, UK, p.95 (1992).

Bouwens, S.M.A.M., P.H. Kosters, and N.V. DOW Benelux, "Simultaneous Process and System Control Design: An Actual Industrial Case," in *IFAC Workshop on Interactions between Process Design and Process Control*, J.D. Perkins, eds., London, UK, p.75 (1992).

Calandranis, J., and G. Stephanopoulos, "Structural Operability Analysis of Heat Exchanger Networks," *Chem. Eng. Res. Des.*, **64**, 347 (1986).

Calandranis, J., and G. Stephanopoulos, "A Structural Approach to the Design of Control Systems in Heat Exchanger Networks," *Comput. Chem. Eng.*, **12**, 651 (1988).

Douglas, J.M., "A Hierarchical Decision Procedure for Process Synthesis," *AIChE J.*, **31**, 353 (1985).

Daoutidis P., and C. Kravaris, "Structural evaluation of control configurations for multivariable nonlinear processes," *Chem. Eng. Sci.*, **47**, 1091 (1992).

Elliott, T.R., and W.L. Luyben, "Capacity-based Economic Approach for the Quantitative Assessment of Process Controllability during the Conceptual Design Stage," *Ind. Eng. Chem. Res.*, **34**, 3907 (1996).

Fisher, W.R., M.F. Doherty, and J.M. Douglas, "The Interface Between Design and Control. 1. Process Controllability," *Ind. Eng. Chem. Res.*, **27**, 597 (1988a).

Fisher, W.R., M.F. Doherty, and J.M. Douglas, "The Interface Between Design and Control. 2. Process Operability," *Ind. Eng. Chem. Res.*, **27**, 606 (1988b).

Fisher, W.R., M.F. Doherty, and J.M. Douglas, "The Interface Between Design and Control. 3. Selecting a Set of Controlled Variables," *Ind. Eng. Chem. Res.*, **27**, 611 (1988c).

Georgiou A., and C.A. Floudas, "Structural analysis and synthesis of feasible control systems — theory and applications," *Chem. Eng. Res. Des.*, **67**, 600 (1989).

Georgiou, A., and C.A. Floudas, "Simultaneous Process Synthesis and Control: Minimization of Disturbance Propagation in Heat Recovery Systems," in *Foundations of Computer-Aided process Design*, J.J. Siirola, I.E. Grossmann, and G. Stephanopoulos, eds., Elsever, New York, p.435 (1990).

Gundersen, T., B. Sagli, N. Hydro, and K. Kiste, "Problems in Sequential and Simultaneous Strategies for Heat Exchanger Network Synthesis," in *Computer-Oriented Process Engineering*, L. Puigjaner and A. Espuna, eds., p. 105 (1991).

Huang, Y.L., and L.T. Fan, "Distributed Strategy for Integration of Process Design and Control — A Knowledge Engineering Approach to the Incorporation of the Controllability Into Exchanger Network Synthesis," *Comput. Chem. Eng.*, **16**, 497 (1992).

Johnston R.D., G.W. Barton, and M.L. Brisk, "Single-input-single-output control system synthesis. Part 1: Structural analysis and the development of feedback control schemes," *Comput. Chem. Eng.*, **9**, 547 (1985).

Kotjabasakis, E., and B. Linnhoff, "Sensitivity Tables for the Design of Flexible Processes (1) — How Much Contingency in Heat Exchanger Networks Is Cost-effective?" *Chem. Eng. Res. Des.*, **64**, 197 (1986).

Linnhoff, B., and E. Kotjabasakis, "Downstream Path for Operable Process Design," *Chem. Eng. Prog.*, .23, (May, 1986).

Luyban, M., and C.A. Floudas, "Analyzing the Interaction of Design and Control — 1. A Multiobjective Framework and Application to Binary Distillation Synthesis," *Comput. Chem. Eng.*, **18**, 933 (1994a).

Luyban, M., and C.A. Floudas, "Analyzing the Interaction of Design and Control — 2. Reactor-Separator-Recycle System," *Comput. Chem. Eng.*, **18**, 971 (1994b).

McAvoy, T.J., "Integration of Process Design and Process Control," in *Recent Development in Chemical Process and Plant Design*, Y.A. Liu, H.A. McGee, Jr., and W.R. Epperly, eds., Wiley, New York, p.289 (1987).

Mohidden, M.J., J.D. Perkins, and E.N. Pistikopoulos, "Optimal Design of Systems under Uncertainty," *AIChE J.*, **42**, 2251 (1996).

Morari, M., and G. Stephanopoulos, "Studies in the synthesis of control structures for chemical process. Part II. Structural aspects and the synthesis of alternative feasible control schemes," *AIChE J.*, **26**, 232 (1980).

Morari, M., "Effect of Design on the Controllability of Chemical Plants", in *IFAC Workshop on Interactions between Process Design and Process Control*, J.D. Perkins eds., London, UK, p.3 (1992).

Papalexandri, K.P., and E.N. Pistikopoulos, "Synthesis and Retrofit Design of Operable Heat Exchanger Networks. 1. Flexibility and Structural Controllability Aspects," *Ind. Eng. Chem. Res.*, **33**, 1718 (1994a).

Papalexandri, K.P., and E.N. Pistikopoulos, "Synthesis and Retrofit Design of Operable Heat Exchanger Networks. 2. Dynamics and Control Structure Considerations," *Ind. Eng. Chem. Res.*, **33**, 1738 (1994b).

Papalexandri, K.P., and E.N. Pistikopoulos, "Generalized Modular Representation Framework for Process Synthesis," *AIChE J.*, **42**, 1010 (1996).

Perkins, J.D., "Interactions Between Process Design and Process Control", in *International Federation of Automatic control Symposium*, Mastricht, Netherlands (1989).

Russo, L.P., and B.W. Bequette, "Impact of Process Design on the Multiplicity Behavior of a Jacketed Exothermic CSTR," *AIChE J.*, **41**, 135 (1995).

Ziegler, J.G., and N.B. Nichols, "Process Lags in Automatic-Control Circuits," *Trans. A.S.M.E.*, **65**, 433 (1943).

CHAPTER 2. DETECTING EXCESS CONTROL OBJECTIVES AND REDESIGNING PLANT FOR STRUCTURAL SYNTHESIS OF FEASIBLE CONTROL SYSTEMS[1]

2.1. Introduction

In control system synthesis, structural models of processes, structurally state and functional controllability have already received wide applications (Morari and Stephanopoulos, 1980; Johnston *et al.*, 1985; Daoutidis and Kravaris, 1992; Papalexandri and Pistikopoulos, 1994). They only need very limited information about processes, and thus are efficiently used at the early stages of conceptual process designs.

For a linear time invariant structural system described by the following equations:

$$\begin{aligned} \dot{x} &= \boldsymbol{A}x + \boldsymbol{B}u \\ y &= \boldsymbol{C}x + \boldsymbol{D}u \end{aligned} \tag{2-1}$$

where,

$\boldsymbol{A}$ = structural plant matrix;

$\boldsymbol{B}$ = structural input matrix;

$\boldsymbol{C}$ = structural output matrix for the state variables;

$\boldsymbol{D}$ = structural output matrix for the manipulated variables;

x = vector of state variables;

---

[1] A version of this chapter has been submitted for publication:
J.Y. Zhu, M. Rao, K.T. Chuang and Z.Z. Han, "A New Approach to Detecting Excess Control Objectives for Structural Synthesis of Feasible Control Systems," *Int. J. Control* (1998).

$\boldsymbol{u}$ = vector of input variables;

$\boldsymbol{y}$ = vector of output (measured) variables.

The controllability test can be made by checking if the generic rank of the compound matrix ($\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$ or $\begin{pmatrix} \boldsymbol{A}^* & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$) satisfies certain conditions (Morari and Stephanopoulos, 1980; Georgiou and Floudas, 1989a). $\boldsymbol{A}^*$ is the structural matrix with the following two features: (a) its diagonal elements are all arbitrary (non-zero) entries, and (b) its non-diagonal elements have the same values as those of $\boldsymbol{A}$ (Georgiou and Floudas, 1989a). In case where the system is not structurally controllable, one or more (noncritical) control objectives should be eliminated, or plant redesign (i.e., adding more manipulated variables to the plant) must be considered (Johnston *et al.*, 1984a) in order to restore the controllability.

Although detecting excess control objectives is required for the synthesis of control systems, few systematic methods have been published (Johnston *et al.*, 1984b; Georgiou and Floudas, 1989b). Johnston *et al.* (1984b) used an enumeration method to find excess control objectives. Starting from the first row, every row and various combinations of all other rows above the row are examined separately. For big structural matrices, the implementation of their method could be tedious. Georgiou and Floudas (1989b) detected excess control objectives by solving the optimization models of the generic rank, and the method is not simple and straightforward.

Because control objectives are represented as rows in the compound structural matrix $\begin{pmatrix} A & B \\ C & D \end{pmatrix}$ or $\begin{pmatrix} A^* & B \\ C & D \end{pmatrix}$, detecting excess control objectives is equivalent to detecting specific excess rows of the compound structural matrix. In this work, excess rows of a structural matrix are found by detecting excess rows of separated maximum dilation submatrices and their various combinations. Each maximum dilation submatrix is divided into two submatrices, and properties of each submatrix are presented. It is also revealed that some excess rows could be missed by only detecting excess rows from separated maximum dilation submatrices. Rules are proposed to find excess rows from separated maximum dilation submatrices and their combinations. Several examples are presented to demonstrate the proposed method.

In practice, some control objectives cannot be eliminated, and thus, new manipulated variables (i.e., plant redesign) should be introduced. A rule is presented for determining new manipulated variables such that the original uncontrollable structural system will become structurally controllable after the new manipulated variables are introduced.

2.2. Determination of Generic Rank of Structural Matrices

The generic rank of a structural matrix is the maximal possible rank achievable by its admissible matrices. The determination of the generic rank of a structural matrix plays a critical role in evaluating structural properties and eliminating excess control objectives.

In general, it is not easy to calculate the generic rank of a structural matrix. So far, many methods have been developed (Burrows and Sahinkaya, 1981; Johnston *et al.*, 1984b; Georgiou and Floudas, 1989b). Among them, the method proposed by Johnston

et al. (1984b) is simple and reliable, and also keeps structural information valuable in control system synthesis (e.g., identify excess control objectives). It is based on the concept of dilation presented by Lin (1974). In a directed graph, if there exist two sets $\boldsymbol{P}$ and $\boldsymbol{T}$ with the properties that there is an oriented edge going directly from a node in $\boldsymbol{T}$ to a node in $\boldsymbol{P}$ and the number of nodes involved in $\boldsymbol{P}$ is greater than the number of nodes in $\boldsymbol{T}$, then the graph contains a dilation. The existence of dilations is equivalent to the rank deficiency of the structural matrix corresponding to the graph (Johnston *et al.*, 1984b).

The method proposed by Johnston *et al.* (1984b) for determining the generic rank of a structural matrix $\boldsymbol{S}$ of size $n \times m$ consists of two procedures: reordering the structural matrix into the form of block triangle and determining the generic rank of the reordered matrix. The detailed descriptions of the two procedures are reviewed in Appendices 2-A and 2-B, respectively.

The total rank deficiency of $\boldsymbol{S}$ is equal to $\sum_{i=1}^{l} p_i$, and thus the generic rank of $\boldsymbol{S}$ is determined as follows:

$$g.r.(\boldsymbol{S}) = n - \sum_{i=1}^{l} p_i \tag{2-2}$$

where,

n = number of rows of $\boldsymbol{S}$;

l = number of dilation blocks of $\boldsymbol{S}$;

p_i = max(number of tagged rows – number of tagged columns) of the ith dilation block.

If the generic rank of $\boldsymbol{S}$ is less than the number of its rows, i.e., $g.r.(\boldsymbol{S}) < n$, then some rows can be eliminated while maintaining the generic rank of $\boldsymbol{S}$. The rows that can be eliminated are called excess rows of $\boldsymbol{S}$, and the rows which cannot be eliminated are called active rows of $\boldsymbol{S}$ in this work.

2.3. Detecting Excess Rows from Single Maximum Dilation Submatrices

Because the deficiency of generic rank is due to the existence of dilations, excess rows must come from submatrices containing dilations. To locate these submatrices, the concept of maximum dilation submatrices was introduced by Johnston *et al.* (1984a).

In a reordered structural matrix $\boldsymbol{S}$ generated by the algorithm given in Appendix 2-A, there could exist several blocks $\boldsymbol{S}_i$ with $(SI)_i < 0$, where $(SI)_i$ = number of columns of $\boldsymbol{S}_i$ - number of rows of $\boldsymbol{S}_i$. Dilations only exist in submatrices containing blocks $\boldsymbol{S}_i$. The maximum dilation submatrix associated with the block $\boldsymbol{S}_i$ is the submatrix with the value of p_i $(p_i = max\{p_{i1}, p_{i2}, \cdots\})$, where p_{ij} = number of tagged rows - number of tagged columns in the jth tagging of rows and columns $(j = 1, 2, \cdots)$ by the algorithm of Appendix 2-B. If more than one submatrix with the value of p_i exists, the one with the greatest number of rows is taken to be the maximum dilation submatrix (Johnston *et al.*, 1984a).

When the first maximum dilation submatrix is found, the rows and columns involved are removed from the original reordered matrix $\boldsymbol{S}$. Other maximum dilation submatrices

can be found in the remainder by applying the same procedure. Obviously, excess rows exist in maximum dilation submatrices.

2.3.1. Partition of maximum dilation submatrices

Let S_i of size $n_i \times m_i$ be a dilation block of the reordered structural matrix S, and M_i be the maximum dilation submatrix of size $a \times b$ associated with S_i. It is assumed that M_i has a rank deficiency, then $p_i > 0$.

To examine properties of M_i, the following algorithm is proposed to divide M_i into two submatrices based on the evaluation of index p_i during the tagging of rows and columns of M_i.

Algorithm 1. (partition of M_i)

Step 1. Let $j = 1$, $r_j = n_i$ and $q_j = m_i$.

Step 2. Tag all rows spanned by the dilation block S_i and number these rows from 1 to n_i. Tag all non-zero columns of S_i and number them from 1 to m_i.

Step 3. Calculate index p_{ij} as follows:

$$p_{ij} = \text{number of tagged rows} - \text{number of tagged columns} = r_j - q_j.$$

Step 4. If all rows and columns are tagged, then stop. Otherwise, go to Step 5.

Step 5. Untagged rows with entries in tagged columns are scanned. The row with the minimum number of untagged columns is selected. If choice exists, the

row with the minimum number of total entries is tagged. Associate the tagged row with index $(r_j + 1)$.

Step 6. Tag untagged non-zero columns in the tagged row. If k_j columns are tagged, associate these columns with indexes q_j, $(q_j + 1)$, ..., $(q_j + k_j)$.

Step 7. Let $r_j = r_j + 1$, $q_j = (q_j + k_j)$ and $j = j + 1$. Go to Step 3.

Let $p_i = max\{p_{i1}, p_{i2}, \cdots, p_{ij}\}$, then p_i is equal to the rank deficiency of $\boldsymbol{M}_i$ (Johnston *et al.*, 1984b). There must exist an integer g $(g \leq j)$ in the partition of $\boldsymbol{M}_i$ by Algorithm 1 such that $p_{ig} = p_i$. Let $\boldsymbol{E}_i$ be the submatrix consisting of rows and columns tagged from the first to the gth tagging of rows and columns by Algorithm 1, and $\boldsymbol{H}_i$ be the submatrix consisting of other rows and the non-zero columns involved in these rows. Thus, $\boldsymbol{M}_i$ can be reordered into the following form:

$$\boldsymbol{M}_i = \begin{bmatrix} \boldsymbol{H}_i, & \boldsymbol{0} \\ \boldsymbol{0}, & \boldsymbol{E}_i \end{bmatrix} \tag{2-3}$$

$\boldsymbol{E}_i$ and $\boldsymbol{H}_i$ have no zero rows and no zero columns, and could share some common columns.

Let

$$p_{E_i} = \text{number of rows of } \boldsymbol{E}_i - \text{number of columns of } \boldsymbol{E}_i \tag{2-4}$$

Obviously, we have $p_{E_i} = p_{ig} = p_i$.

It will be shown that the division of M_i into the combination of H_i and E_i will play a key role in detecting excess rows of M_i by the proposed method.

2.3.2. Detection of excess rows

Since $p_{E_i} = p_i$, the rank deficiency of M_i is equal to that of E_i, and H_i has no rank deficiency. Therefore, H_i and E_i should have some properties which are important to detecting excess rows of M_i.

Property 1. If $H_i \neq 0$, then H_i has a full row generic rank, i.e., $g.r.(H_i)$ = number of rows of H_i.

Proof. From Algorithm 1, we have $p_{E_i} = p_i$. Thus, E_i contains all rank deficiency of M_i and H_i has no rank deficiency. If one of rows of H_i is deleted from M_i, the generic rank of M_i will also decrease by one. Thus, H_i has a full row generic rank. ■

Property 2. E_i has a full column generic rank, i.e., $g.r.(E_i)$ = number of columns of E_i.

Proof. Because the rank deficiency of M_i is equal to that of E_i, it can be obtained that:

$$g.r.(E_i) = \text{number of rows of } E_i - p_{E_i}$$

from Algorithm 1, we also have:

$$p_{E_i} = \text{number of rows of } E_i - \text{number of columns of } E_i.$$

thus, it can be concluded that $g.r.(\boldsymbol{E}_i)$ = number of columns of $\boldsymbol{E}_i$, i.e., $\boldsymbol{E}_i$ has a full column generic rank. ■

Based on Properties 1 and 2, the following rules can be proposed.

Rule 1. All excess rows of $\boldsymbol{M}_i$ are only from $\boldsymbol{E}_i$.

This rule is obvious from Property 1. Thus, rows of $\boldsymbol{H}_i$ need not to be considered in detecting excess rows of $\boldsymbol{M}_i$, which decreases the search space of excess rows of $\boldsymbol{M}_i$.

Rule 2. Any k rows $(k \leq p_{E_i})$ from $\boldsymbol{E}_i$ are excess rows of $\boldsymbol{E}_i$ if and only if after these rows are deleted from $\boldsymbol{E}_i$, the remainder of $\boldsymbol{E}_i$ can be reordered into a square submatrix with all non-zero diagonal entries and with the size of the number of columns of $\boldsymbol{E}_i$.

Proof. By definition, the generic rank of a structural matrix is the maximal possible rank which can be achieved by its admissible matrices. Thus, for a square structural matrix, if and only if the structural matrix can be reordered into the one whose all diagonal entries are not zero, then its generic rank can be equal to its size.

When any k rows $(k \leq p_{E_i})$ of $\boldsymbol{E}_i$ are deleted, we can always find a square matrix in the remainder of $\boldsymbol{E}_i$. Therefore, if and only if we can find a square matrix in the remainder of $\boldsymbol{E}_i$ such that the square matrix can be reordered into the one with all non-zero diagonal entries, the deleted k rows are excess rows of $\boldsymbol{E}_i$. ■

It should be pointed out that Rule 2 cannot be simply applied to $\boldsymbol{M}_i$ because $\boldsymbol{M}_i$ does not have the property of the full column generic rank.

2.4. Detecting Excess Rows from the Combination of Different Maximum Dilation Submatrices

Excess rows of a structural matrix not only can come from each individual maximum dilation submatrix, but also can come from the combination of several separated maximum dilation submatrices. The reasons are that firstly, the reordering of a structural matrix by the algorithm of Appendix 2-B can produce different reordered matrices, thus different maximum dilation submatrices can be obtained. Secondly, a row which cannot be eliminated from an individual maximum dilation submatrix could become an excess row when several maximum dilation submatrices are considered together. Consider the following reordered structural matrix:

$$
\boldsymbol{S} = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{matrix} \\ \begin{bmatrix} x & x & & & & & & \\ x & x & & & & & & \\ & x & x & & & & & \\ & x & x & & & & & \\ & & & x & x & & & \\ x & x & & & & x & & \\ x & x & & & & x & & \\ & & & & & x & x & \\ & & & & & & x & x \\ & x & x & & x & & x & x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{matrix}
$$

There exist the following two maximum dilation submatrices:

$$
\boldsymbol{M}_1 = \begin{array}{c} \begin{matrix} 1 & 2 & 3 \end{matrix} \\ \begin{bmatrix} x & x & \\ x & x & \\ & x & x \\ & x & x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} \quad \text{and} \quad \boldsymbol{M}_2 = \begin{array}{c} \begin{matrix} 6 \end{matrix} \\ \begin{bmatrix} x \\ x \end{bmatrix} \end{array} \begin{matrix} 6 \\ 7 \end{matrix}
$$

From M_1, it is obtained that $E_1 = M_1 = \begin{matrix} 1 & 2 & 3 \\ \end{matrix}$
$\begin{bmatrix} x & x & \\ x & x & \\ & x & x \\ & x & x \end{bmatrix}$
$\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix}$ and $p_{E_1} = p_{M_1} = 1$. By Rule 2, it can be easily determined that one of rows 1, 2, 3 and 4 is an excess row of M_1.

From M_2, we have $E_2 = M_2 = \begin{matrix} 6 \\ \end{matrix}$
$\begin{bmatrix} x \\ x \end{bmatrix}$
$\begin{matrix} 6 \\ 7 \end{matrix}$ and $P_{E_2} = P_{M_2} = 1$. The excess row is either row 6 or 7. Thus, by examining M_1 and M_2 independently, the detected excess rows of S are one of rows 1, 2, 3 and 4, and one of rows 6 and 7.

In fact, there still exist other excess rows which cannot be detected if M_1 and M_2 are taken into account separately. The additional excess rows of S are both rows 1 and 2, or one of rows 3 and 4 and one of rows 1 and 2.

2.4.1. Detecting excess rows from the combination of two maximum dilation submatrices

As the simplest case, the combination of two maximum dilation submatrices is considered first.

2.4.1.1. Constitution of additional excess rows

Let M_{12} be the combination of two individual maximum dilation submatrices M_1 and M_2, then M_{12} can be written as:

$$M_{12} = \begin{pmatrix} M_1 & 0 \\ L_{12} & M_2 \end{pmatrix} \tag{2-5}$$

where, L_{12} is a submatrix of S whose indices of rows are the same as those of M_2 and indices of columns are the same as those of M_1.

The number of excess rows of M_{12} is determined by the following equation:

$$p_{M_{12}} = p_{M_1} + p_{M_2} \tag{2-6}$$

where, $p_{M_{12}}$, p_{M_1} and p_{M_2} are rank deficiencies of M_{12}, M_1 and M_2, respectively.

When M_1 and M_2 are considered together, we have the following different cases to take into account.

Case (i) One or more active rows of M_1 could become excess row(s) of M_{12}.

Obviously, the only case is that the active rows of M_1 can replace some excess rows of M_2 to be eliminated from M_{12}. The reason is that the rows of L_{12} associated with the excess rows of M_2 could replace active rows of M_1 when M_{12} is considered. Under this case, additional excess rows of M_{12} consist of one or more active rows of M_1 which can be replaced by some excess rows of M_2 and other excess rows of M_2.

Case (ii) One or more active rows of M_2 become excess row(s) of M_{12}.

This case will not happen because M_2 does not share common columns with M_1 and no active rows of M_2 can be replaced by excess rows of M_1.

2.4.1.2. Detection of additional excess rows

From the above analyses, it can be concluded that when M_{12} is considered, additional excess rows of M_{12} can be found by only examining if some active rows of M_1 can replace some excess rows of M_2. Both H_1 and E_1 contain active rows of M_1, thus they need to be examined individually.

A. Rows of H_1

To determine what rows of H_1 can replace some excess rows of E_2, the submatrix (denoted as $\tilde{H}_1$) generated by eliminating common columns of H_1 and E_1 from H_1 only needs to be examined because from Algorithm 1, we know it is $\tilde{H}_1$ that makes H_1 has a full row generic rank.

Property 3. Let n_{H_1} be the number of rows of H_1, and $m_{\tilde{H}_1}$ be the number of columns of $\tilde{H}_1$, then $n_{H_1} \le m_{\tilde{H}_1}$.

Proof. Let n_{E_1} and m_{E_1} be the numbers of rows and columns of E_1. Then, we have

$$p_{E_1} = n_{E_1} - m_{E_1}$$

$$p_{M_1} = (n_{E_1} + n_{H_1}) - (m_{E_1} + m_{\tilde{H}_1})$$

Because $p_{E_1} = p_1 = max(p_{11}, p_{12}, \cdots)$ and $p_{M_1} \le p_1$, we can obtain that $p_{M_1} \le p_{E_1}$ which implies $n_{H_1} \le m_{\tilde{H}_1}$. ■

Because $\tilde{H}_1$ has a full row generic rank and $n_{H_1} \le m_{\tilde{H}_1}$, we can at least find a submatrix $\tilde{H}_1^D$ which consists of n_{H_1} rows and n_{H_1} columns of $\tilde{H}_1$ such that $\tilde{H}_1^D$ has a full generic rank. Thus, $\tilde{H}_1^D$ can be reordered such that its diagonal entries are all non-zero.

Rule 3. A row of $\tilde{H}_1^D$ can be replaced by an excess row of E_2 if and only if the row of L_{12} associated with the excess row of E_2 has a non-zero entry in the same column as the diagonal entry of the row of $\tilde{H}_1^D$.

Proof.

Sufficient

Let e be the row of submatrix $[L_{12}, M_2]$ associated with an excess row of E_2 and $[L_{12}^r, M_2^r]$ be the matrix generated by eliminating e from $[L_{12}, M_2]$. Let a be a row of H_1 and M_1^r be the matrix generated by deleting a from M_1. Obviously, $[L_{12}, M_2]$ and M_1 can be reordered as $[L_{12}, M_2] = \begin{bmatrix} e \\ [L_{12}^r, M_2^r] \end{bmatrix}$ and $M_1 = \begin{bmatrix} a \\ M_1^r \end{bmatrix}$.

Since e is an excess row of $[L_{12}, M_2]$, and a and e have the same entry on the diagonal of $\tilde{H}_1^D$, then

$$g.r.(M_{12}) = g.r.\left(\begin{array}{c:c} M_1 & 0 \\ \hdashline L_{12} & M_2 \end{array}\right) = g.r.\left(\begin{array}{c:c} M_1 & 0 \\ \hdashline L_{12}^r & M_2^r \end{array}\right) = g.r.\left(\begin{array}{c:c} a & 0 \\ M_1^r & \\ \hdashline L_{12}^r & M_2^r \end{array}\right)$$

$$= g.r.\begin{pmatrix} e \\ \hdashline M_1^r \quad 0 \\ \hdashline L_{12}^r \quad M_2^r \end{pmatrix} = g.r.\begin{pmatrix} M_1^r \quad 0 \\ \hdashline e \\ \hdashline L_{12}^r \quad M_2^r \end{pmatrix} = g.r.\begin{pmatrix} M_1^r & 0 \\ L_{12} & M_2 \end{pmatrix}$$

Therefore, a can replace e to be an excess row of M_{12}.

Necessary

Let $(\tilde{H}_1^D)^r$ be the matrix obtained by eliminating a from $\tilde{H}_1^D$. $\tilde{H}_1^D$ can be reordered as $\tilde{H}_1^D = \begin{pmatrix} a \\ (\tilde{H}_1^D)^r \end{pmatrix}$. Since a can be replaced by e, we have

$$g.r.\left(\tilde{H}_1^D\right) = g.r.\begin{pmatrix} a \\ (\tilde{H}_1^D)^r \end{pmatrix} = g.r.\begin{pmatrix} e \\ (\tilde{H}_1^D)^r, \ 0 \end{pmatrix}$$

Because $\tilde{H}_1^D$ has a full row rank, e must have a non-zero entry in the same column as the diagonal entry of a in $\tilde{H}_1^D$. ■

By Rule 3, we can easily know if a row in H_1 can be an excess row of M_{12} or not.

B. Active rows of E_1

E_1 can be reordered as $E_1 = \begin{pmatrix} E_1^a \\ E_1^e \end{pmatrix}$, where E_1^e consists of p_{M_1} excess rows of M_1, and E_1^a consists of all other rows of E_1. Obviously, E_1^a has a full generic rank and is square. Similarly, a simple rule can be obtained to examine if a row of E_1^a can be an excess row of E_2.

Rule 4. A row of E_1^a can be replaced by an excess row of E_2 if and only if the row of L_{12} associated with the excess row of E_2 has a non-zero entry in the same column as the diagonal entry of the row of E_1^a.

Rule 4 can be proved in the way similar to the proof of Rule 3. By Rules 3 and 4, additional excess rows of M_{12} can be found systematically.

2.4.2. Detecting excess rows from the combination of more than two maximum dilation submatrices

Detecting additional excess rows from the combination of more than two individual maximum dilation submatrices can be performed based on the method for detecting additional excess rows from the combination of two separated maximum dilation submatrices.

As an example, determining all excess rows of a structural matrix involving three maximum dilation submatrices M_1, M_2 and M_3 can be conducted as follows:

(i) detecting excess rows of each maximum dilation submatrix;

(ii) finding additional excess rows from the combinations of any two maximum dilation submatrices;

(iii) locating additional excess rows from the combination of three maximum dilation submatrices. One set of additional excess rows of the combination of M_1, M_2 and M_3 can be generated in the way that excess rows of M_2 replace active rows of M_1, and active rows of M_2 are replaced by excess rows of M_3. Obviously,

the rules for the detection of excess rows from two individual maximum dilation submatrices can be applied, but only various combinations should be examined.

All rules proposed for detecting all excess rows have sound theoretic bases. Therefore, the method developed in this work is reliable.

2.5. Plant Redesign

It may be considered that all control objectives are "critical" and must be retained, which may thereby necessitate plant redesign. Plant redesign is to introduce new manipulated variables so that the plant will become structurally controllable while all control objectives can be accomplished.

In the sense of the manipulation of the structural compound matrix $\boldsymbol{S}$ ($\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$ or $\begin{pmatrix} \boldsymbol{A}^{*} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$), plant redesign means some new columns should be added to $\boldsymbol{S}$ so that $\boldsymbol{S}$ can achieve a full row generic rank. The new columns should be added in some systematic ways. In this section, an approach to adding columns is presented.

It is obvious that if $\boldsymbol{S}$ of size $n \times m$ has v excess rows (i.e., $g.r.(\boldsymbol{S}) = n - v$), then at least v new columns (i.e., a new matrix $\boldsymbol{Q}$ of size $n \times v$) should be added to $\boldsymbol{S}$ in order that the composite matrix $(\boldsymbol{S}, \boldsymbol{Q})$ has no excess rows.

Thus, the smallest number of new columns which must be added to the original structural matrix $\boldsymbol{S}$ is equal to the number of excess rows which cannot be deleted.

Property 4. Let v excess rows of $\boldsymbol{S}$ be rows $e_1, e_2, \cdots, e_v$ and $\boldsymbol{Q}_v$ be the submatrix of $v \times v$ consisting of rows $e_1, e_2, \cdots, e_v$ of $\boldsymbol{Q}$. Then, if the matrix $\boldsymbol{Q}_v$ satisfy the following condition:

$$g.r.(\boldsymbol{Q}_v) = v$$

the compound matrix $(\boldsymbol{S}, \boldsymbol{Q})$ has a full row generic rank, i.e., $g.r.(\boldsymbol{S}, \boldsymbol{Q}) = n$.

Proof. $\boldsymbol{S}$ of size $n \times m$ can be reordered as $\boldsymbol{S} = \begin{bmatrix} \boldsymbol{S}_1 \\ \boldsymbol{S}_v \end{bmatrix}$, where $\boldsymbol{S}_v$ of size $v \times m$ consists of rows $e_1, e_2, \cdots, e_v$, and $\boldsymbol{S}_1$ of size $(n-v) \times m$ consists of other rows of $\boldsymbol{S}$. Thus, $\boldsymbol{Q}$ of size $n \times v$ can also be reordered as $\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{Q}_1 \\ \boldsymbol{Q}_v \end{bmatrix}$, where, $\boldsymbol{Q}_1$ has size of $(n-v) \times v$ and $\boldsymbol{Q}_v$ has size of $v \times v$.

$\boldsymbol{S}_1$ has a full row generic rank, i.e., $g.r.(\boldsymbol{S}_1) = n - v$, then we have $m \geq (n-v)$ because $g.r.(\boldsymbol{S}_1) = \min(n-v, m)$. We also have

$$g.r.\begin{bmatrix} \boldsymbol{S}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_v \end{bmatrix} = g.r.(\boldsymbol{S}_1) + g.r.(\boldsymbol{Q}_v) = (n-v) + v = n$$

Since $m \geq (n-v)$, it can be obtained that $(m+v) \geq n$, i.e., the number of columns of $(\boldsymbol{S}, \boldsymbol{Q})$ is not less than the number of its rows. From the above equation, $\begin{bmatrix} \boldsymbol{S}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_v \end{bmatrix}$ has the maximal rank of $(\boldsymbol{S}, \boldsymbol{Q})$. By the definition of generic rank, we can conclude that $g.r.(\boldsymbol{S}, \boldsymbol{Q}) = n$. ■

Property 4 means that plant redesign should introduce the new manipulated variables so that they affect all of excess control objectives. The simplest case is that one-to-one effects between the new manipulated variables and the excess control objectives. Whether or not the newly introduced manipulated variables affect other control objectives has no effects on the controllability of the new systems. Thus, a rule for introducing new manipulated variables can be presented as follows:

Rule 5. To make v uncontrollable control objectives controllable, v new manipulated variables should be introduced and there should exist at least one-to-one effects between the new manipulated variables and the uncontrollable control objectives.

2.6. Examples

To illustrate and demonstrate the proposed rules for detecting excess rows and for plant redesign, several examples are presented.

2.6.1. Detecting excess rows from single maximum dilation submatrix

Example 1.

$$M = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} \\ \begin{bmatrix} x & & & & & \\ & x & & & & \\ & & x & & & \\ & & & x & & \\ & & & & x & x \\ & & & & x & x \\ & & & x & x & x \\ & & & x & x & x \\ x & & x & x & x & x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{matrix}$$

By Algorithm 1, it is determined that $p_E = 3$ and

$$\boldsymbol{E} = \begin{array}{c} \begin{matrix} 1 & 3 & 4 & 5 & 6 \end{matrix} \\ \begin{bmatrix} x & & & & \\ & x & & & \\ & & x & & \\ & & & x & x \\ & & & x & x \\ & & x & x & x \\ & & x & x & x \\ x & x & x & x & x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{matrix}$$

According to Rule 2, rows 1 and 3, or 1 and 9, or 3 and 9 cannot be excess rows. The excess rows should be any three rows from $\boldsymbol{E}$ except those including the combination of rows 1 and 3, or 1 and 9, or 3 and 9.

Example 2.

$$\boldsymbol{S} = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} \\ \begin{bmatrix} x & & & & & \\ x & & & & & \\ & x & & & & \\ & x & & & & \\ x & x & x & x & & \\ & & & x & x & x \\ x & & & x & x & x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{matrix}$$

Since $g.r.(\boldsymbol{S}) = 5$, then there exist two excess rows. The maximum dilation submatrices are:

$$\boldsymbol{M}_1 = \begin{array}{c} \begin{matrix} 1 \end{matrix} \\ \begin{bmatrix} x \\ x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 2 \end{matrix} \text{ and } \boldsymbol{M}_2 = \begin{array}{c} \begin{matrix} 2 \end{matrix} \\ \begin{bmatrix} x \\ x \end{bmatrix} \end{array} \begin{matrix} 3 \\ 4 \end{matrix}$$

Obviously, $p_{M_1} = 1$, $p_{M_2} = 1$, $\boldsymbol{E}_1 = \boldsymbol{M}_1$ and $\boldsymbol{E}_2 = \boldsymbol{M}_2$. It is easy to find that two excess rows of $\boldsymbol{S}$ are one of rows 1 and 2, and one of rows 3 and 4.

This example shows that the proposed method also works for structural matrices whose generic rank are not equal to the number of its column.

2.6.2. Detecting excess rows from two maximum dilation submatrices

Example 3.

$$\boldsymbol{S} = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} \\ \left[\begin{array}{cccccc} x & x & & & & \\ & x & x & & & \\ & & & x & x & \\ & & & x & x & \\ x & & & x & x & \\ x & & & x & x & \\ \hdashline x & & & x & x & x \\ x & & x & x & x & x \end{array}\right] \end{array} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{matrix}$$

The maximum dilation submatrices involved in $\boldsymbol{S}$ are:

$$\boldsymbol{M}_1 = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \left[\begin{array}{ccccc} x & x & & & \\ & x & x & & \\ & & & x & x \\ & & & x & x \\ x & & & x & x \\ x & & & x & x \end{array}\right] \end{array} \begin{matrix} 1 \\ 2 \\ 3, \\ 4 \\ 5 \\ 6 \end{matrix} \quad \boldsymbol{M}_2 = \begin{array}{c} \begin{matrix} 6 \end{matrix} \\ \left[\begin{array}{c} x \\ x \end{array}\right] \end{array} \begin{matrix} 7 \\ 8, \end{matrix} \quad \boldsymbol{L}_{12} = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \left[\begin{array}{ccccc} x & & & x & x \\ x & & x & x & x \end{array}\right] \end{array} \begin{matrix} 7 \\ 8 \end{matrix}$$

We can also obtain that:

$$\boldsymbol{H}_1 = \begin{array}{c} \begin{matrix} 1 & 2 & 3 \end{matrix} \\ \left[\begin{array}{ccc} x & x & \\ & x & x \end{array}\right] \end{array} \begin{matrix} 1 \\ 2, \end{matrix} \quad \boldsymbol{E}_1 = \begin{array}{c} \begin{matrix} 1 & 4 & 5 \end{matrix} \\ \left[\begin{array}{ccc} & x & x \\ & x & x \\ x & x & x \\ x & x & x \end{array}\right] \end{array} \begin{matrix} 3 \\ 4 \\ 5 \\ 6 \end{matrix}$$

$$\boldsymbol{E}_2 = \boldsymbol{M}_2 = \begin{array}{c} \begin{matrix} 6 \end{matrix} \\ \left[\begin{array}{c} x \\ x \end{array}\right] \end{array} \begin{matrix} 7 \\ 8 \end{matrix}$$

When M_1 and M_2 are considered separately, it can be easily found that one of rows 3, 4, 5, 6 and one of rows 7 and 8 are excess rows of S. To find additional excess rows of S, we need to examine if any active rows of H_1 and E_1 can replace excess rows of M_2.

$$\tilde{H}_1^D = \tilde{H}_1 = \begin{matrix} \\ \text{x} \\ \text{x} \end{matrix}\!\!\begin{matrix} 2 & 3 \\ \left[\begin{matrix} \text{x} & \\ \text{x} & \text{x} \end{matrix}\right. & \begin{matrix} 1 \\ 2 \end{matrix} \end{matrix}$$

$\tilde{H}_1^D$ can be reordered into the following two diagonal forms:

$$\tilde{H}_1^D \sim \begin{matrix} \\ \text{x} \\ \text{x} \end{matrix}\!\!\begin{matrix} 2 & 3 \\ \left[\begin{matrix} \text{x} & \\ \text{x} & \text{x} \end{matrix}\right. & \begin{matrix} 1 \\ 2 \end{matrix} \end{matrix} \text{ and } \tilde{H}_1^D \sim \begin{matrix} \\ \text{x} \\ \end{matrix}\!\!\begin{matrix} 2 & 3 \\ \left[\begin{matrix} \text{x} & \text{x} \\ & \text{x} \end{matrix}\right. & \begin{matrix} 2 \\ 1 \end{matrix} \end{matrix}.$$

From Rule 3, if row 1 or row 2 can replace row 7 (or 8) to be an excess row of S, entry (7,2) or (7,3) (or (8,2) or (8,3)) of L_{12} should not be zero. Therefore, row 1 or 2 can replace row 8 to become an excess row of S.

Now, we see if some active rows of E_1 can be replaced by excess rows of M_2. When row 3 or 4 or 5 or 6 is selected as an excess row of M_1, E_1^a can be one of the follows:

$$E_1^a = \begin{matrix} \\ \\ \text{x} \\ \text{x} \\ \text{x} \end{matrix}\!\!\begin{matrix} 1 & 4 & 5 \\ \left[\begin{matrix} & \text{x} & \text{x} \\ \text{x} & \text{x} & \text{x} \\ \text{x} & \text{x} & \text{x} \end{matrix}\right. & \begin{matrix} 4 \\ 5 \\ 6 \end{matrix} \end{matrix}, \text{ or } E_1^a = \begin{matrix} \\ \\ \text{x} \\ \text{x} \\ \text{x} \end{matrix}\!\!\begin{matrix} 1 & 4 & 5 \\ \left[\begin{matrix} & \text{x} & \text{x} \\ \text{x} & \text{x} & \text{x} \\ \text{x} & \text{x} & \text{x} \end{matrix}\right. & \begin{matrix} 3 \\ 5, \\ 6 \end{matrix} \end{matrix}$$

or $$E_1^a = \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} 1 & 4 & 5 \\ \end{matrix} \begin{bmatrix} & x & x \\ & x & x \\ x & x & x \end{bmatrix} \begin{matrix} 3 \\ 4 \\ 6 \end{matrix}$$, or $$E_1^a = \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} 1 & 4 & 5 \\ \end{matrix} \begin{bmatrix} & x & x \\ & x & x \\ x & x & x \end{bmatrix} \begin{matrix} 3 \\ 4 \\ 5 \end{matrix}$$

From Rule 4, we can obtain that any one of rows 3, 4, 5 and 6 can replace row 7 or 8 to be an excess row of S.

Therefore, all excess rows of S are:

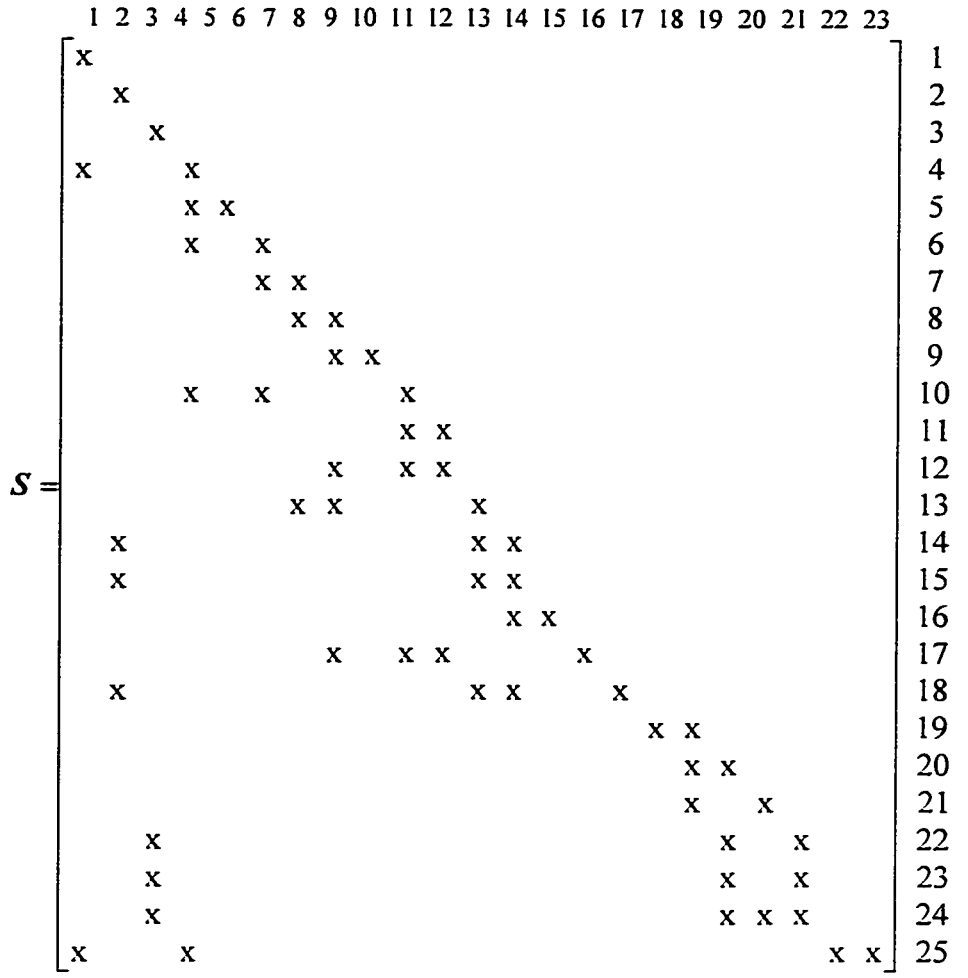
(a) one of rows 3, 4, 5 and 6, and one of rows 7 and 8;

(b) one of rows 3, 4, 5 and 6, and one of rows 1 and 2;

(c) any two rows of rows 3, 4, 5 and 6.

2.6.3. Finding all excess rows of structural matrices with more than two maximum dilation submatrices

Example 4. The Williams-Otto plant (Williams and Otto, 1960) has been widely used as a test case for control studies (Morari and Stephanopoulos, 1980; Johnston *et al.*, 1985; Georgiou and Floudas, 1989). The following structural matrix S is the reordered compound matrix $\begin{pmatrix} A & B \\ C & D \end{pmatrix}$ for the Williams-Otto plant (Johnston *et al.* 1985).



There are three individual maximum dilation submatrices M_1, M_2 and M_3. All excess rows of S can be found by examining separated maximum dilation matrices, combinations of different two maximum dilation submatrices and the combination of all three maximum dilation submatrices.

(1) Excess rows of each maximum dilation submatrix

$$M_1 = \begin{array}{c} \begin{matrix} 1 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \end{matrix} \\ \begin{bmatrix} x & & & & & & & & \\ x & x & & & & & & & \\ & x & x & & & & & & \\ & x & & x & & & & & \\ & & & x & x & & & & \\ & & & & x & x & & & \\ & & & & & x & x & & \\ & x & & x & & & & x & \\ & & & & & & & x & x \\ & & & & & x & & x & x \end{bmatrix} \end{array} \begin{matrix} 1 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \end{matrix}, \quad E_1 = M_1$$

$P_{E_1} = 1$ and $P_{M_1} = 1$. One of rows 1, 4, 6, 7, 8, 10, 11 and 12 is an excess row of M_1, and rows 5 and 9 cannot be eliminated from M_1.

$$M_2 = \begin{array}{c} \begin{matrix} 2 & 12 & 13 \end{matrix} \\ \begin{bmatrix} x & & \\ & x & \\ x & x & x \\ x & x & x \end{bmatrix} \end{array} \begin{matrix} 2 \\ 13, \\ 14 \\ 15 \end{matrix} \quad E_2 = M_2$$

$P_{E_2} = 1$ and $P_{M_2} = 1$. One of rows 2, 13, 14 and 15 can be eliminated from M_2.

$$M_3 = \begin{array}{c} \begin{matrix} 3 & 17 & 18 & 19 & 20 & 21 \end{matrix} \\ \begin{bmatrix} x & & & & & \\ & x & x & & & \\ & & x & x & & \\ & & x & & x & \\ x & & & x & & x \\ x & & & x & & x \\ x & & & x & x & x \end{bmatrix} \end{array} \begin{matrix} 3 \\ 19 \\ 20 \\ 21 \\ 22 \\ 23 \\ 24 \end{matrix}$$

$$E_3 = \begin{array}{c} \begin{matrix} 3 & 18 & 19 & 20 & 21 \end{matrix} \\ \begin{bmatrix} x & & & & \\ & x & x & & \\ & x & & x & \\ x & & x & & x \\ x & & x & & x \\ x & & x & x & x \end{bmatrix} \end{array} \begin{matrix} 3 \\ 20 \\ 21, \\ 22 \\ 23 \\ 24 \end{matrix} \qquad H_3 = \begin{array}{c} \begin{matrix} 17 & 18 \end{matrix} \\ \begin{bmatrix} x & x \end{bmatrix} \end{array} 19$$

$P_{E_3} = 1$ and $P_{M_3} = 1$. Any one of rows 3, 20, 21, 22, 23 and 24 can be an excess row of M_3, and row 19 cannot be eliminated from M_3.

(2) Additional excess rows from combinations of two maximum dilation submatrices

$$M_{12} = \begin{bmatrix} M_1 & 0 \\ L_{12} & M_2 \end{bmatrix} = \begin{array}{c} \begin{matrix} 1 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 2 & 12 & 13 \end{matrix} \\ \left[\begin{array}{cccccccccccc} x & & & & & & & & & & & \\ x & x & & & & & & & & & & \\ & x & x & & & & & & & & & \\ & x & & x & & & & & & & & \\ & & & x & x & & & & & & & \\ & & & & x & x & & & & & & \\ & & & & & x & x & & & & & \\ & x & & x & & & & x & & & & \\ & & & & & & & x & x & & & \\ & & & & & x & & x & x & & & \\ \hdashline & & & & & & & & & x & & \\ & & & & x & x & & & & & x & \\ & & & & & & & & & x & x & x \\ & & & & & & & & & x & x & x \end{array}\right] \end{array} \begin{matrix} 1 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 2 \\ 13 \\ 14 \\ 15 \end{matrix}$$

From excess rows of M_1 and M_2, excess rows of M_{12} can be one of rows 1, 4, 6, 7, 8, 10, 11, and 12 and one of rows 2, 13, 14 and 15. When M_{12} is considered, row 7, 8 or 12 can replace row 13 as an excess row. Thus, additional excess rows of M_{12} are one of rows 1, 4, 6, 10 and 11, and one of rows 7, 8 and 12.

$$M_{13}=\begin{bmatrix} M_1 & 0 \\ L_{13} & M_3 \end{bmatrix}=$$

	1	4	5	6	7	8	9	10	11	3	17	18	19	20	21	
	x															1
	x	x														4
		x	x													5
		x		x												6
				x	x											7
					x	x										8
						x	x									9
		x		x				x								10
								x	x							11
						x		x	x							12
										x						3
											x	x				19
												x	x			20
												x		x		21
										x			x		x	22
										x			x		x	23
										x			x	x	x	24

Because $L_{13}=\boldsymbol{0}$, no additional excess rows can be found from M_{13}.

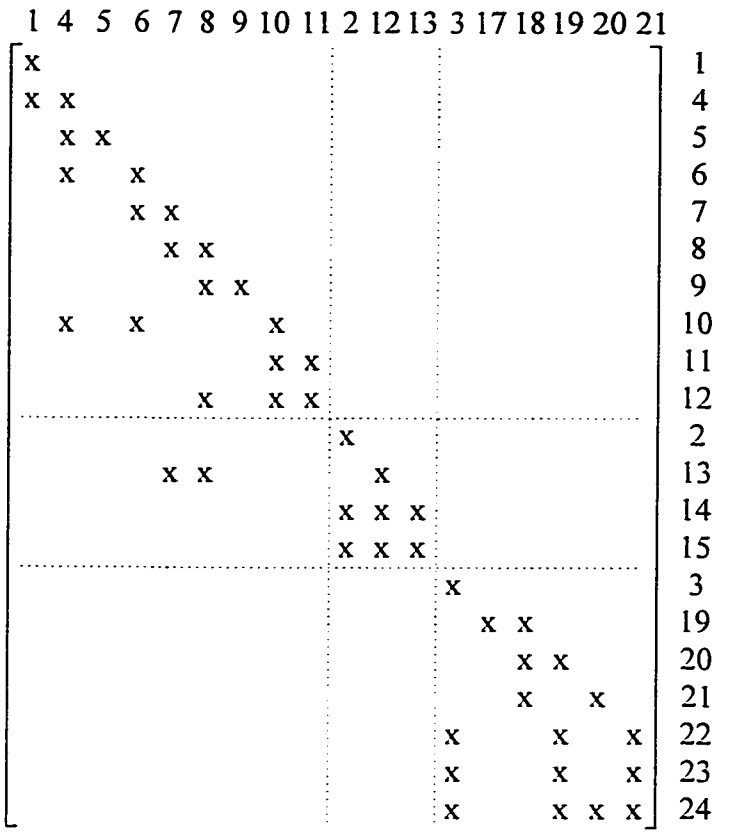
$$M_{23}=\begin{bmatrix} M_2 & 0 \\ L_{23} & M_3 \end{bmatrix}=$$

	2	12	13	3	17	18	19	20	21	
	x									2
		x								13
	x	x	x							14
	x	x	x							15
				x						3
					x	x				19
						x	x			20
						x		x		21
				x			x		x	22
				x			x		x	23
				x			x	x	x	24

$L_{23}=\boldsymbol{0}$, thus no additional excess rows exist in M_{23}.

(3) Additional excess rows from the combination of three maximum dilation submatrices

$$M_{123}=\begin{bmatrix} M_1 & 0 & 0 \\ L_{12} & M_2 & 0 \\ L_{13} & L_{23} & M_3 \end{bmatrix}=$$



Because $L_{13}=0$ and $L_{23}=0$, no additional excess rows can be detected from M_{123}.

In summary, all excess rows of S can be given as follows:

(i) One of rows 1, 4, 6, 7, 8, 10, 11 and 12, one of rows 2, 13, 14 and 15, and one of rows 3, 20, 21, 22, 23 and 24;

(ii) One of rows 1, 4, 6, 10 and 11, one of rows 7, 8 and 12, and one of rows 3, 20, 21, 22, 23 and 24.

2.6.4. Plant redesign

Example 5. Consider a system of two heating tanks in series as shown in Figure 2-1(a). The linear dynamic models of the system are (Georgiou and Floudas, 1989):

$$\begin{pmatrix} \dot{T}_1 \\ \dot{T}_2 \end{pmatrix} = \begin{bmatrix} \frac{-F_s}{V_1} & 0 \\ \frac{F_s}{V_2} & \frac{-F_s}{V_2} \end{bmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} + \begin{bmatrix} \frac{1}{V_1 c_p} & \frac{(T_0 - T_1)}{V_2 c_p} \\ 0 & 0 \end{bmatrix} \begin{pmatrix} Q \\ F \end{pmatrix}$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}$$

where:

V_1:	holdup;	T_i:	temperature;
F:	feed flow;	c_p:	heat capacity;
Q:	heat input in the first tank;	s:	steady state.

Thus, we have

$$\boldsymbol{x} = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}; \quad \boldsymbol{u} = \begin{pmatrix} Q \\ F \end{pmatrix}; \quad \boldsymbol{y} = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix};$$

$$\boldsymbol{A} = \begin{bmatrix} \mathrm{x} & 0 \\ \mathrm{x} & \mathrm{x} \end{bmatrix}; \qquad \boldsymbol{B} = \begin{bmatrix} \mathrm{x} & \mathrm{x} \\ 0 & 0 \end{bmatrix};$$

$$\boldsymbol{C} = \begin{bmatrix} \mathrm{x} & 0 \\ 0 & \mathrm{x} \end{bmatrix}; \qquad \boldsymbol{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

The compound structural matrix $\begin{pmatrix} A & B \\ C & D \end{pmatrix}$ or $\begin{pmatrix} A^* & B \\ C & D \end{pmatrix}$ of the system is described as:

$$S = \begin{array}{c} \begin{matrix} x_1 & x_2 & u_1 & u_2 \end{matrix} \\ \begin{bmatrix} x & 0 & x & x \\ x & x & 0 & 0 \\ x & 0 & 0 & 0 \\ 0 & x & 0 & 0 \end{bmatrix} \end{array} \begin{matrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{matrix}$$

It can be obtained that $g.r.(S) = 3$. Therefore, the system of two heating tanks in series is not structurally controllable and there exists an excess control objective.

(a) Determination of excess control objectives

The structural matrix S is reordered into the following form:

$$\tilde{S} = \begin{array}{c} \begin{matrix} x_1 & x_2 & u_1 & u_2 \end{matrix} \\ \begin{bmatrix} x & 0 & 0 & 0 \\ 0 & x & 0 & 0 \\ x & x & 0 & 0 \\ x & 0 & x & x \end{bmatrix} \end{array} \begin{matrix} y_1 \\ y_2 \\ x_2 \\ x_1 \end{matrix}$$

There exists only one maximum dilation submatrix in $\tilde{S}$, i.e.,

$$M_1 = \begin{array}{c} \begin{matrix} x_1 & x_2 \end{matrix} \\ \begin{bmatrix} x & 0 \\ 0 & x \\ x & x \end{bmatrix} \end{array} \begin{matrix} y_1 \\ y_2 \\ x_2 \end{matrix}$$

It is easy to obtain that one of rows y_1, y_2 and x_2 is an excess row of S. Therefore, the excess control objective is either y_1 or y_2.

(b) Plant redesign

By Rule 5, it can be seen that by introducing a new heat input Q_2 into the second tank as shown in Figure 2-1(b), the modified system will be structurally controllable. The linear models of the modified system then become:

$$\begin{pmatrix} \dot{T}_1 \\ \dot{T}_2 \end{pmatrix} = \begin{pmatrix} x & 0 \\ x & x \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} + \begin{pmatrix} x & x & 0 \\ 0 & 0 & x \end{pmatrix} \begin{pmatrix} Q \\ F \\ Q_2 \end{pmatrix}$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} x & 0 \\ 0 & x \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}$$

2.7. Conclusions

Although the concept of maximum dilation submatrices decreases the searching space of excess rows, we cannot get all excess rows by only examining individual maximum dilation submatrices. Thus, the proposed method consists of two procedures: finding excess rows from individual maximum dilation submatrices and finding additional excess rows from their various combinations.

An individual maximum dilation submatrix M_i is divided into two submatrices, i.e., a submatrix H_i without excess rows, and a submatrix E_i including all excess rows of M_i. Because E_i has a full column generic rank, excess rows involved in E_i can be detected by checking after deleting these rows, if we can find a square submatrix whose size is equal to the number of columns of E_i and whose diagonal entries are all non-zero.

Additional excess rows from the combination of separated maximum dilation submatrices can be found by examining if active rows of an upper maximum dilation submatrix can be replaced by excess rows of its lower maximum dilation submatrices.

The basic idea for determining additional excess rows is to check if such replacements can still maintain the generic rank of the upper maximum dilation submatrix.

Comparing with the method of Johnston *et al.* (1984b), the proposed method considers not only individual maximum dilation submatrices but also their combinations. The presented procedures render the inherent properties of maximum dilation submatrices and thus are more efficient, especially for big structural matrices.

In addition, plant redesign to achieve a controllable process while maintaining all control objectives is also discussed. It is found that new manipulated variable which at least have one-to-one effects on the excess control objectives of the original process must be available in the redesigned plant.

Appendix 2-A. Reordering of a structural matrix (Johnston *et al.*, 1984b)

Step 1. Let $i = 0$ and $j = 0$.

Step 2. Find the row of the structural matrix A with the least number of undeleted entries (say k). If choice exists, the row with the minimum number of total entries (deleted and undeleted) is selected.

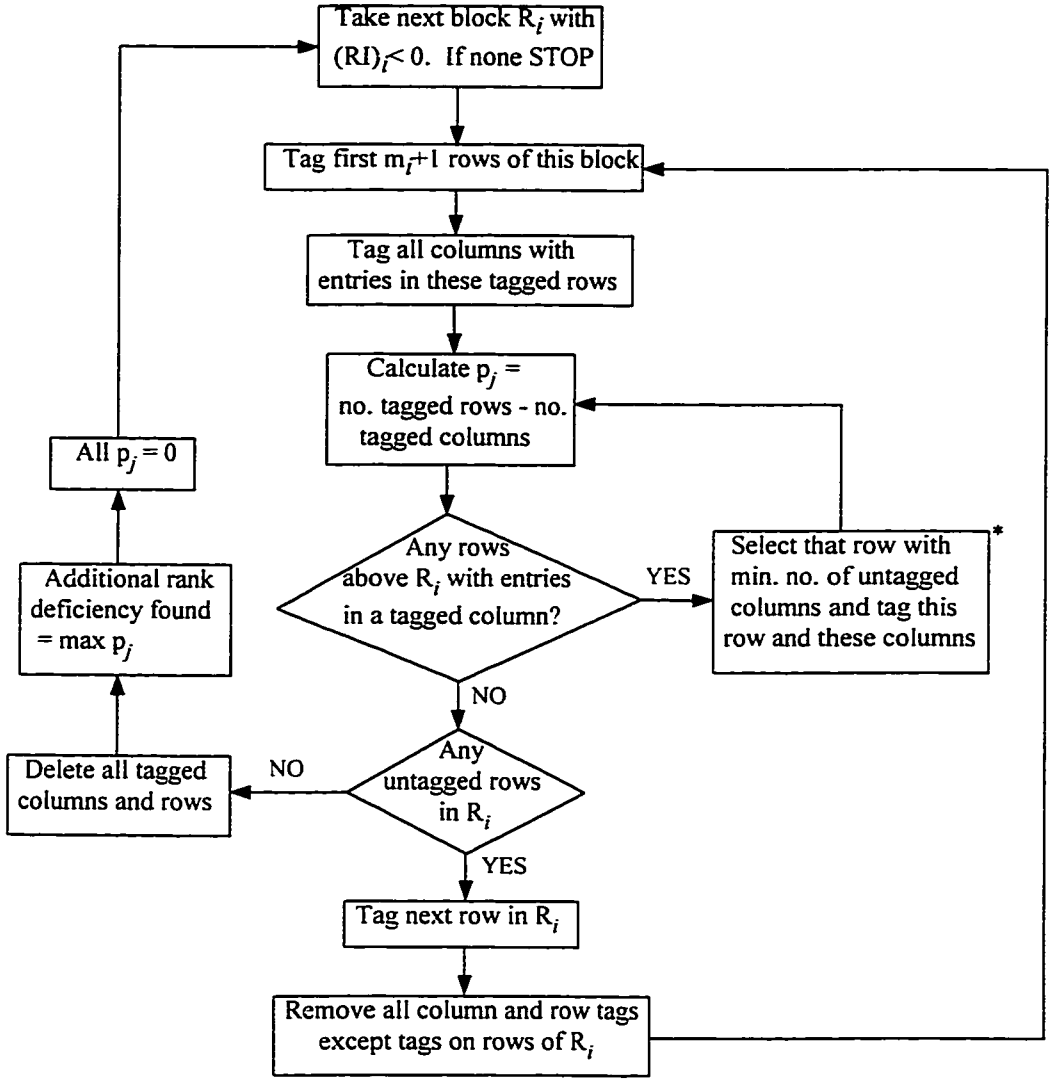
Step 3. Let $i = i + 1$ and associate this row with index i and delete it.

Step 4. Associate the columns which have entries in this row with indices j+1, j+2, ..., j+k and delete them. Let $j = j + k$.

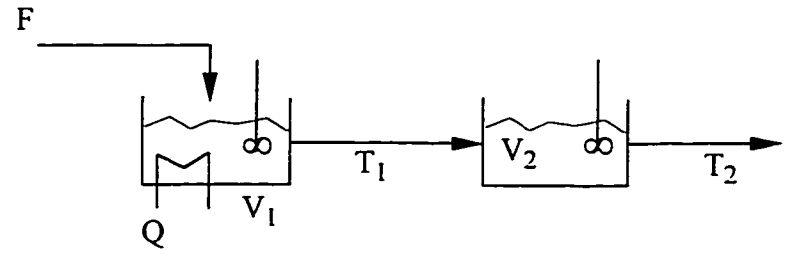
Step 5. If there are any rows left, then go to Step 2. Otherwise, go to END.

Rewrite structural matrix A with rows having increasing indices from top to bottom and with columns having increasing indices from left to right. The blocks A_i with the size of $n_i \times m_i$ $(i = 1, 2, \cdots, l)$ are the individual blocks along the upper edge of the reordered matrix and contain no fixed zero elements.

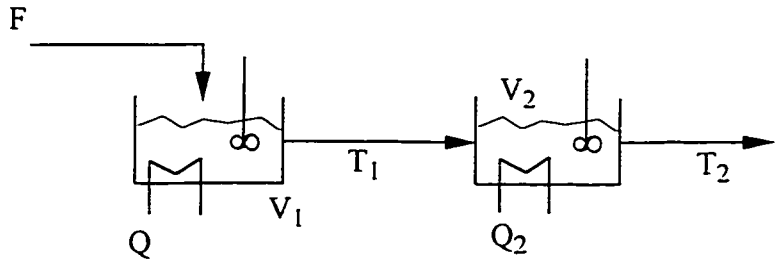
Appendix 2-B. Determination of generic rank of a structural matrix (Johnston *et al.*, 1984b)



*, if a choice exists, select that with the minimum total entries.



(a) Original design (Georgiou and Floudas, 1989a).



(b) Modified design.

Figure 2-1. Two heating tanks in series.

2.8 Literature Cited

Burrows, C.R., and M.N. Sahinkaya, "A New Algorithm for Determining Structural Controllability," *Int. J. Control*, **33**, 379 (1981).

Daoutidis, P., and C. Kravaris, "Structural Evaluation of Control Configurations for Multivariable Nonlinear Processes," *Chem. Eng. Sci.*, **47**, 1091 (1992).

Georgiou, A., and C.A. Floudas, "Structural Analysis and Synthesis of Feasible Control Systems — Theory and Applications," *Chem. Eng. Res. Des.*, **67**, 600 (1989a).

Georgiou, A., and C.A. Floudas, "Optimization Model for Generic Rank Determination of Structural Matrices," *Int. J. Control*, **49**, 1633 (1989b).

Johnston, R.D., G.W. Barton, and M.L. Brisk, "Control Objective Reduction in Single-Input Single-Output Control Schemes," *Int. J. Control*, **40**, 265 (1984a).

Johnston, R.D., G.W. Barton, and M.L. Brisk, "Determination of the Generic Rank of Structural Matrices," *Int. J. Control*, **40**, 257 (1984b).

Johnston, R.D., G.W. Barton, and M.L. Brisk, "Single-Input-Single-Output Control System Synthesis. Part 1: Structural Analysis and the Development of Feedback Control Schemes," *Comput. Chem. Eng.*, **9**, 547 (1985).

Lin, C.T., "Structural Controllability," *IEEE Trans. Automat. Contr.*, **19**, 201 (1974).

Morari, M., and G. Stephanopoulos, "Studies in the Synthesis of Control Structures for Chemical Process. Part II. Structural Aspects and the Synthesis of Alternative Feasible Control Schemes," *AIChE J.*, **26**, 232 (1980).

Papalexandri, K. P., and E. N. Pistikopoulos, "Synthesis and Retrofit Design of Operable Heat Exchanger Networks. 1. Flexibility and Structural Controllability Aspects," *Ind. Eng. Chem. Res.*, **33**, 1718 (1994).

Williams, T.J., and R.E. Otto, "A Generalized Chemical Processing Model for the Investigation of Computer Control," *AIEE Trans.*, **79** (Part 1), 458 (1960).

CHAPTER 3. INTEGRATED PROCESS DESIGN AND PLANT-WIDE CONTROL

3.1. Introduction

Different from other types of engineering problems, the design problem of translating a chemist's discovery of a new reaction into a process flowsheet is always underdefined, i.e., only a very small fraction of the information can be available from the problem statement (Douglas, 1985) to define the design problem. The space of the design problem is also very large. It is estimated that there could exist over a million process alternatives for the same design problem (Douglas, 1985). To make this problem more tractable, it is now widely accepted that a hierarchical approach is necessary. The essence of hierarchical approaches is to utilize a means for discriminating between important information and details in problem space (Sacerdoti, 1974). The hierarchical decision procedure developed by Douglas (1985) is well defined, and has been applied in designs of various processes, such as separation system (Douglas, 1995), polymer process (McKenna and Malone, 1990), waste minimization (Douglas, 1992), solid process (Rajagopal *et al.*, 1992) and batch process (Stephanopoulos and Han, 1994). The procedure has also been used in control system design (Ponton and Laing, 1993) and integrated process design and control (Fisher *et al.*, 1988).

Fisher *et al.* (1988) proposed a systematic method for assessing controllability at the preliminary stages of a process design, so that some of the economic penalties associated with control can be used as an additional criterion for screening process alternatives and conflicts between design and control can be avoided as early as possible. The method is

only based on steady-state models of processes, and the controllability incorporated in the process design is to check if the number of manipulated variables is equal to the number of constraints plus operating variables. As noted in their work, if process dynamics are considered, new controllability limitations would arise. One simple example is that the controllability criterion is not always satisfied when a process has the equal numbers of manipulated variables and controlled variables. Although they check controllability at each design level, they did not ensure that the process generated at each level is always controllable, instead they continue to the next level and see if new manipulated variables will be introduced, which could violate the principle of the hierarchical method.

In this work, a systematic method is proposed for the integration of process design and plant-wide control by incorporating structural controllability (state and functional) into each design stage of Douglas' hierarchical procedure for process design. The method guarantees that processes generated at each design level are always controllable, so that some control problems can be avoided at early stages of process design. The method also makes the control system synthesis of a complete plant simpler because it decomposes the big system into several smaller subsystems and coordinates the designs of the smaller subsystems by checking the controllability of the composite system consisting of the subsystems.

3.2. Hierarchical Decision Procedure for Process Design

The hierarchical decision procedure proposed by Douglas (1985) starts a process design from a single block with only feed and product streams, and ends the design with a full process flow sheet by sequential elaboration of the flow sheet. The approach

decomposes a very large and complex problem into a number of smaller problems that are much easier to handle, and involves the following five decision levels:

Level 1: Batch vs Continuous

Level 2: Input-Output Structure of the Flow Sheet

Level 3: Recycle Structure of the Flow Sheet and Reactor Considerations

Level 4: Separation System Specification

Level 4a: Vapor recovery system

Level 4b: Liquid recovery system

Level 5: Heat Exchanger Network

At each level, a complete process is always considered, but more detailed structure is added to the flow sheet by making decisions at the current design level. Various heuristics are proposed to guide the decision making at each level. Economic analyses made at each level will help select the best decision from generated design alternatives or discard poor projects as early as possible, which is highly desirable for new processes because less than one percent of the ideas for new designs are ever commercialized (Douglas, 1985). The decisions needed to make at each level are listed as follows (Douglas, 1985):

Level 2: input-output structure of the flow sheet

1. Is a gas recycle and purge system required?
2. How many product streams are required?

3. Should the feed streams be purified or should the impurities be processed?

4. Should reversible byproducts be recovered or recycled to extinction?

5. What selectivity losses are associated with complex reactions?

Level 3: recycle structure of the flow sheet

1. Is more than one reactor required?

2. How many recycle streams are there?

3. Do we want to use an excess of one reactant at the reactor inlet?

4. Is gas recycle compressor required?

5. Should the reactor be operated adiabatically, with direct heating or cooling, or with a diluent acting as a heat carrier?

6. If the reactor is equilibrium limited, do we want to shift the equilibrium?

Level 4: general structure of the separation system

1. Is a phase split of the reactor effluent possible?

2. Is part of the reactor effluent in the vapor phase? Can part of the vapor be condensed at ambient temperature?

Level 4a: vapor recovery system

1. Where should the vapor recovery system be located?

2. What type of vapor recovery system should be used?

Level 4b: liquid recovery system

1. How many of the separations can be accomplished by distillation?
2. What arrangement of distillation columns should be used?
3. How should the light components be removed?
4. Should the light components be vented to the atmosphere, sent to fuel, or recycled to the vapor recovery system?
5. How should we accomplish the other separation?

3.3. Control System Synthesis

Most of work in process control is focused on design performance aspects of controllers, while other important problems such as how to select measured and manipulated variables and how to design control structures are not paid much attention. Quite often, measured and manipulated variables are selected not based on theoretic systematic approaches, but based on designer's experience. The main task of control system synthesis is to "explore the question of which variables should be measured, which variables should be manipulated and how these two sets of variables should be interconnected to form the control loops" (Nishida *et al.*, 1981).

According to Nishida *et al.* (1981), a control system consists of:

(i) a set of control objectives;

(ii) a set of controlled variables;

(iii) a set of measured variables for monitoring the behavior of the process;

(iv) a set of manipulated variable;

(v) a structure interconnecting measured and manipulated variables (control loops).

There often exist many feasible control structure alternatives for the same control problem. To evaluate these alternatives, we can use techniques such as relative gain array (RGA) and single value decomposition (SVD) for numerical systems and relative order (Daoutidis and Kravaris, 1992) for structural systems.

The control system synthesis can be classified into two sub-problems (Daoutidis and Kravaris, 1992):

(i) generation of all feasible control configurations.

(ii) evaluation and selection of a control configuration.

The integrated process design and plant-wide control in this work only handles the first sub-problem. The proposed procedure will guarantee that at each design level, the control configuration is always feasible by introducing redesign or ignoring uncritical control objectives if necessary.

3.4. Local Control and Structural Controllability of Composite Systems

The proposed hierarchical approach takes into account each subsystem independently, so that the determination of manipulated variables and control objectives, detection of excess control objectives and plant redesign are much easier to handle. For a subsystem with more critical control objectives than manipulated variables available associated with the same subsystem, if its upstream subsystem has unused manipulated variables, we can use nonlocal or local control to accomplish the control system. Nonlocal control directly uses the unused manipulated variables of the upstream

subsystem to maintain the objectives; while, local control only uses the manipulated variables associated with the same subsystem. To implement local control, it is necessary to find some input variables of the subsystem that can be output variables of the upstream subsystem. In practice, local control is usually preferable to nonlocal control (Johnston 1985).

Local control causes the coupling among the subsystems. Such coupling could change the controllability of the composite system consisting of the subsystems. Therefore, under such situations, the controllability test of the composite system is required after designs of all subsystems are finished. To restore the controllability of the composite system, we may introduce new manipulated variables to one or more subsystems.

3.5. Structurally State and Functional Controllability

A system is said to be state controllable at time t_0 if it is possible to find an unconstrained control vector to transfer any initial state to the origin in a finite time interval. Stated mathematically, the system is controllable at t_0 if for any $x(t_0)$, there exist t_1 and $u_{[t_0, t_1]}$ such that $x(t_0)=0$ and $x(t_1)=0$ $(t_1 > t_0)$. If it is true for all initial time t_0 and all initial states $x(t_0)$, the system is said to be completely controllable.

When time-invariant systems are only considered, the initial time can be fixed to zero, i.e., $t_0 = 0$. If a time-invariant system is controllable at an initial time, then it is controllable at all $t \in [0,\infty]$. This is why the initial time of time-invariant systems can be fixed and we never designate the initial time.

Although the initial and final states need to be specified in the definition of state controllability, no conditions can be imposed on the trajectory between those states or on the behavior of the trajectory after the final time. This is important since the major goal of regulatory control is usually to maintain the plant at some steady state (Russell and Perkins, 1987). To overcome the limitation of state controllability, the concept of functional controllability was introduced.

Rosenbrock (1970) defined that a system is *functionally controllable*, if given any "suitable" vector $y(t)$ of output functions defined for $t > 0$, there exists a vector $u(t)$ of inputs defined for $t > 0$, which generates the output vector $y(t)$ from the initial condition $x(0) = 0$.

Structurally state and functional controllability are developed by extending the above two concepts of controllability to structural systems. Thus, a structural system is *structurally state controllable* (or *structurally functional controllable*) if there exists an admissible system which is controllable in the classical sense. An *admissible matrix* with respect to a structural matrix is defined as the matrix obtained by fixing the arbitrary (non-zero) elements of the structural matrix at some specified values.

For the following linear time invariant system:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \end{cases} \tag{3-1}$$

where,

$\boldsymbol{A}$ = plant matrix;

$\boldsymbol{B}$ = input matrix;

$\boldsymbol{C}$ = output matrix for the state variables;

$\boldsymbol{D}$ = output matrix for the manipulated variables;

$\boldsymbol{x}$ = vector of state variables;

$\boldsymbol{u}$ = vector of input variables;

$\boldsymbol{y}$ = vector of output (measured) variables.

Morari and Stephanopoulos (1980) extended the concept of controllability of the system $(\boldsymbol{A}, \boldsymbol{B})$ to the system including PI feedback control action, and showed that the pair $\left[\begin{pmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{0} \end{pmatrix}, \begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{D} \end{pmatrix}\right]$ is structurally controllable if and only if the following conditions are satisfied:

(i) $(\boldsymbol{A}, \boldsymbol{B})$ is structurally controllable;

(ii) $$g.r.\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix} = q + r\,. \qquad (3\text{-}2)$$

where, $q = \dim(\boldsymbol{x})$; $\quad r = \dim(\boldsymbol{y})$.

A necessary condition for (ii) above is that $g.r.(\boldsymbol{A}, \boldsymbol{B}) = q$. Thus, only the second generic rank test needs to be performed (Johnston *et al.*, 1985).

Georgiou and Floudas (1989) showed that system (3-1) is structurally functional controllable with respect to r control objectives and q manipulated variables if and only if:

(i) $g.r.(A, B) = q$;

(ii) $g.r.\begin{pmatrix} A^* & B \\ C & D \end{pmatrix} = q + r$. (3-3)

where, $A^* \in R^{q \times q}$ is the structural matrix with the following two features: (a) its diagonal elements are all arbitrary (non-zero) entries, and (b) its non-diagonal elements have the same values as those of A.

The above two criteria have been widely used in control system synthesis, and will also be used in this work.

3.6. Hierarchical Approach to Integration of Process Design and Plant-wide Control

The hierarchical design method for the integration of design and control proposed in this work uses the same procedure as defined by Douglas (1985). We only consider continuous processes. Thus, only the last four levels of the hierarchical design procedure are taken into account. In the integrated process design and control, controllability is used as an additional criterion for screening design alternatives. The objective of the integrated design and control is to design processes that satisfy not only economical targets but also control requirements, i.e., the generated processes at each level are structurally controllable.

3.6.1. Design procedures at each design level

The proposed approach incorporates structurally state or functional controllability into designs of each subsystem and the composite system. The integrated design and control at each level can be divided into two stages: design and evaluation.

The design stage only deals with new manipulated variables and new control objectives introduced at the current level. Other manipulated variables and control objectives of upper levels are not considered. By doing so, designs of control systems at the current level are simplified. The controllability test at each level is made for all subsystems. If a subsystem is not controllable, we find excess control objectives. Then we examine if the excess control objectives are important or not. We keep the critical control objectives but neglect uncritical control objectives. If the subsystem is still uncontrollable, process redesign must be made.

In the evaluation stage, we examine the controllability of the composite system if necessary. After all subsystems at the same level are controllable, if one or more subsystems have coupling of variables with their upstream subsystems, we examine if the composite system is still controllable. If not, we introduce new manipulated variables to subsystems and make the composite system controllable.

We don't continue the designs of the next level until all subsystems and the composite system at the current level are controllable. The above procedures can be shown in Figure 3-1.

3.6.2. Decision making for each subsystem

Same as the approach of Fisher *et al.* (1988), a preliminary screening of process design alternatives is completed before controllability analysis is incorporated. The controllability evaluation and plant redesign are only considered for the design alternatives which are thought profitable.

At each design level, we do the following for each subsystem:

(1) Identify new manipulated variables and new control objectives introduced at the current level.

(2) Develop structural models concerning new manipulated variables and new control objectives.

(3) Decide if structurally functional controllability is required. If yes, structurally functional controllability is considered in the following steps; otherwise, structural state controllability is considered.

(4) Examine if the subsystem is structurally controllable. If yes, we say the subsystem is controllable. If no, we detect excess control objectives. If some excess control objectives cannot be ignored, we introduce new manipulated variables or modify some existing manipulated variables until the subsystem becomes controllable.

(5) If its downstream subsystems use local control, then a controllability test should be made again after the designs of the downstream subsystems are finished if there exist the coupling of variables among the subsystem and its downstream subsystems.

3.7. Integrated Design and Control for the Process of Hydrodealkylation of Toluene to Produce Benzene (HDA)

To illustrate the proposed procedures for the integrated process design and plant-wide control, the well-known process of hydrodealkylation of toluene to produce benzene (HDA) is used as a case study.

3.7.1. Process description

The HDA process produces benzene from hydrogen and toluene, and includes the following two reactions:

$$\text{Toluene} + H_2 \rightarrow \text{Benzene} + CH_4 \tag{3-4}$$

$$2\ \text{Benzene} \quad \Leftrightarrow \text{Diphenyl} + H_2 \tag{3-5}$$

One feed stream is available at a nominal rate of 125 kgmol/h and contains 99.8% toluene and 0.2% heavy, inert impurity. Another feed stream which consists of 95% H_2 and 5% CH_4 is available at 311 K (100 °F) and 3790 kN/m^2 (550 psia).

The pressure of the reactor is maintained at 3450 kN/m^2 (500 psia). In order to obtain a sufficiently large reaction rate, the reactor inlet temperature must exceed 895 K (1150 °F). Also, the hydrogen-to-aromatics ratio at the reactor inlet must exceed 5 to prevent coking. The conversion (x) and selectivity (S) are defined as follows (Douglas, 1988):

$$x = \frac{moles\ Toluene\ converted\ in\ reactor}{moles\ Toluene\ feed\ to\ reactor} \tag{3-6}$$

$$S = \frac{moles\ Benzene\ at\ reactor\ outlet}{moles\ Toluene\ converted} \tag{3-7}$$

For the HDA process, the following equation can be obtained (Douglas, 1988):

$$S = \frac{0.0036}{(1-x)^{1.544}} \tag{3-8}$$

The temperature of the reactor outlet must be kept less than 978 K (1300 °F) to prevent hydrocracking reactions. To prevent coking in the heat exchanger following the reactor, the reactor effluent must be rapidly cooled to 895 K (1150 °F). The process is required to produce benzene of 265 kgmol/h at a purity of 0.9999.

3.7.2. Process design

Douglas (1985) has discussed in much detail the synthesis of the flow sheets for the HDA process at each level of the hierarchical design procedure. Here, the decisions made at each level are briefly reviewed in order to explain the integrated design and control of the HDA process.

Level 2. Input-output structure

At this stage, feed streams are hydrogen and toluene, and product streams are benzene and diphenyl.

Because the primary reaction also produces methane, it is unnecessary to purify the hydrogen feed stream. As hydrogen and methane are lighter than propylene, a purge stream and a gas recycle are needed. This decision is based in part on the fact that "propylene can be condensed with cooling water at high pressure, so that all materials

less volatile than propylene can be recovered by distillation (or some other liquid separation scheme) using only cooling water" (Douglas, 1985).

The overall material balance of the flowsheet at level 2 can be given as follows (Douglas, 1985):

$$F_{FT} = P_B / S \tag{3-9}$$

$$P_D = P_B \frac{1-S}{2S} \tag{3-10}$$

$$F_G = \frac{P_B}{(0.95 - y_{PH})}\left[1 - \frac{2(1-y_{PH})}{(1-S)}\right] \tag{3-11}$$

$$P_G = F_G + P_B \frac{1-S}{2S} \tag{3-12}$$

where, P_B is the production rate of benzene; F_{FT} is the fresh feed rate of toluene;

P_D is the production rate of diphenyl; F_G is the feed hydrogen flow rate;

P_G is purge flow rate;

y_{PH} is mole fraction of hydrogen in the purge stream.

From Equations (3-9) - (3-12), it can be seen that if S (or x), P_B and y_{PH} are given, all other flow rates can then be determined.

Level 3. Recycle structure

A single reactor is only needed because two reactions (3-4) and (3-5) take place at the same temperature and pressure. Since the hydrogen and methane are a gas recycle,

which is separated from the liquid toluene recycle by the benzene product, two recycle streams must be used.

The flow rate of the two recycle streams can be calculated by the following equations (Douglas, 1985):

$$F_T = F_{FT} / x = P_B / xS \tag{3-13}$$

$$R_G = \frac{1}{y_{PH}}\left(5\frac{P_B}{Sx} - 0.95F_G\right) \tag{3-14}$$

where, F_T is the flow rate of toluene entering the reactor;

R_G is the recycle gas flow rate.

Level 4. General structure of separation system

One of design heuristics adopted by Douglas (1985) is that a phase split is always the cheapest separation, and therefore the separation is divided into a vapor recovery system and a liquid separation system. Because the reactor effluent is vapor, the effluent stream should be cooled to cooling water temperature and thus, a phase split can be accomplished. The flash liquid is sent to a liquid recovery system and the flash vapor is sent to a vapor recovery system. The evaluation of the economic potential of the vapor recovery system for the HDA process shows that the recovery system is not necessary.

The liquid separation system only consists of distillation columns. For the sequence of distillation columns, some of the most commonly applied heuristics are (a) recover the lightest component first, (b) recover the most plentiful component first, (c) make the most difficult splits last, and (d) favor equimolar splits (Douglas, 1985). For the HDA process,

all of the above rules favorite the direct sequence. Because hydrogen and methane contained in the liquid of the flash drum will contaminate the benzene product which is expected to be recovered at purity of 0.9999, the first column (stabilizer) is used to recover hydrogen and methane. The second column (benzene column) and the third column (toluene column) are used to recover benzene and toluene, respectively.

Level 5. Heat exchanger networks

A feed-effluent heat exchanger is used to heat the feed stream of the reactor and cool the effluent stream of the reactor, and the liquid effluent of the flash drum is sent back to quench the reactor outlet stream to 895 K.

3.7.3. Integration of process design and control

Level 2. Input-output structure

At this level, we choose the production rate of benzene (P_B), the conversion (x) and the composition of hydrogen in the purge stream (y_{PH}) as control objectives. The manipulated variables available are the feed hydrogen flow rate (F_G) and the fresh feed rate of toluene (F_{FT}). From the mass balance calculations at this level, the following dynamic models can be developed:

$$\dot{P}_B = f(P_B, x, F_G, F_{FT}) \tag{3-15}$$

$$\dot{y}_{PH} = f(y_{PH}, x, F_G, F_{FT}) \tag{3-16}$$

$$\dot{x} = f(x, F_G, F_{FT}) \tag{3-17}$$

Thus, we have:

$$x=\begin{pmatrix} P_B \\ y_{PH} \\ x \end{pmatrix}, \qquad u=\begin{pmatrix} F_G \\ F_{FT} \end{pmatrix}, \qquad y=\begin{pmatrix} P_B \\ y_{PH} \\ x \end{pmatrix}$$

$$A=\begin{pmatrix} x & 0 & x \\ 0 & x & x \\ 0 & 0 & x \end{pmatrix}, \qquad B=\begin{pmatrix} x & x \\ x & x \\ x & x \end{pmatrix}$$

$$C=\begin{pmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{pmatrix}, \qquad D=\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Because $A^* = A$, the criteria of structurally state and functional controllability are the same. We don't differentiate structurally state and functional controllability for the input-output structure. In the subsequent discussions of this work, if $A^* = A$, we also do not differentiate structurally state and functional controllability.

The compound structure of the system can be given as:

$$S_{L2}=\begin{pmatrix} A & B \\ C & D \end{pmatrix}=\begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & u_1 & u_2 \end{matrix} \\ \begin{pmatrix} x & 0 & x & x & x \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ x & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} x_1 \\ x_2 \\ x_3 \\ y_1 \\ y_2 \\ y_3 \end{matrix} \qquad (3\text{-}18)$$

The above structural matrix can be reordered as:

$$
\begin{array}{c}
\begin{array}{ccccc} x_1 & x_2 & x_3 & u_1 & u_2 \end{array} \\
\begin{pmatrix} x & 0 & x & x & x \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ x & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 \end{pmatrix}
\end{array}
\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ y_1 \\ y_2 \\ y_3 \end{array}
\sim
\begin{array}{c}
\begin{array}{ccccc} x_1 & x_2 & x_3 & u_1 & u_2 \end{array} \\
\begin{pmatrix} x & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 \\ 0 & 0 & x & x & x \\ x & 0 & x & x & x \\ 0 & x & x & x & x \end{pmatrix}
\end{array}
\begin{array}{c} y_1 \\ y_2 \\ y_3 \\ x_3 \\ x_1 \\ x_2 \end{array}
$$

It is easy to get that $g.r.(S_{L2}) = 5 \neq 6$. Thus, system S_{L2} is not structurally controllable. By using the method developed in the previous chapter, we can see that any one of y_1, y_2 and y_3 can be an excess control objective. Since F_{FT} has a significant effect on P_B, and F_G has a significant effect on y_{PH} (Fisher *et al.*, 1988), we select x as the excess control objective. Because x must be controlled, we need to redesign the process so that x is controllable. From the previous chapter, we know a new manipulated variable which can affect x should be introduced. It is known that x is sensitive to the inlet temperature of the reactor, we need to introduce a new manipulated variable that can be used to control the inlet temperature of the reactor. Such a new manipulated variable can be the fuel flow of a furnace if we use the furnace to heat the feed streams of the reactor, or the flow rate of the hot stream of a heater if the heater is used. Because the energy integration system will not be considered until level 5, we do not consider introducing a furnace or a heater at this level, instead we only introduce a new manipulated variable ($F_{Heating}$) which is intended to control the inlet temperature of the reactor. Therefore, the flowsheet generated at this level can be as shown in Figure 3-2.

Level 3. Recycle structure

The new control objectives which need to be maintained at this level are:

(a) the reactor outlet temperature ($T_{R,O}$) which should be less than 978 K, and

(b) the ratio of H_2/aromatics at the reactor inlet ($r_{R,I}$) which should be equal to 5.

The new manipulated variable introduced at this level is the recycle gas flow rate (R_G).

At this level, there exist two subsystems, i.e., reactor system and separation system. For these two different subsystems, we assume local control is needed for each subsystem. As stated above, we first consider the integrated design of each subsystem, and then examine if the composite system is still controllable.

(1) Reactor system

The reactor system at this level has two control objectives (i.e., $T_{R,O}$ and $r_{R,I}$). There is only one manipulated variable R_G. The dynamic models describing the reactor system are given below:

$$\dot{T}_{R,O} = f(T_{R,O}, R_G) \tag{3-19}$$

$$\dot{r}_{R,I} = f(r_{R,I}, R_G) \tag{3-20}$$

$$\boldsymbol{x} = \begin{pmatrix} T_{R,O} \\ r_{R,I} \end{pmatrix}, \qquad \boldsymbol{u} = R_G \ , \qquad \boldsymbol{y} = \begin{pmatrix} T_{R,O} \\ r_{R,I} \end{pmatrix}$$

$$\boldsymbol{A} = \begin{pmatrix} \mathrm{x} & 0 \\ 0 & \mathrm{x} \end{pmatrix}, \qquad \boldsymbol{B} = \begin{pmatrix} \mathrm{x} \\ \mathrm{x} \end{pmatrix}$$

$$\boldsymbol{C}=\begin{pmatrix} \mathrm{x} & 0 \\ 0 & \mathrm{x} \end{pmatrix}, \qquad \boldsymbol{D}=\begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{S}_R=\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}=\begin{array}{c} \begin{matrix} x_1 & x_2 & u \end{matrix} \\ \begin{pmatrix} \mathrm{x} & 0 & \mathrm{x} \\ 0 & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & 0 & 0 \\ 0 & \mathrm{x} & 0 \end{pmatrix} \end{array} \begin{matrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{matrix} \tag{3-21}$$

$$\begin{array}{c} \begin{matrix} x_1 & x_2 & u \end{matrix} \\ \begin{pmatrix} \mathrm{x} & 0 & \mathrm{x} \\ 0 & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & 0 & 0 \\ 0 & \mathrm{x} & 0 \end{pmatrix} \end{array} \begin{matrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{matrix} \sim \begin{array}{c} \begin{matrix} x_1 & x_2 & u \end{matrix} \\ \begin{pmatrix} \mathrm{x} & 0 & 0 \\ 0 & \mathrm{x} & 0 \\ \mathrm{x} & 0 & \mathrm{x} \\ 0 & \mathrm{x} & \mathrm{x} \end{pmatrix} \end{array} \begin{matrix} y_1 \\ y_2 \\ x_1 \\ x_2 \end{matrix}$$

It can be obtained that $g.r.(\boldsymbol{S}_R)=2\neq 3$. Thus, subsystem $\boldsymbol{S}_R$ is not structurally controllable. Either y_1 or y_2 can be the excess control objective. Because R_G has greater effect on $r_{R.I}$ than on $T_{R.O}$, we select $T_{R.O}$ as the excess control objective. As stated by Fisher *et al.* (1988), there are three ways to restore the process controllability:

(a) reducing the conversion so that $T_{R.O}$ is always less than 978 K.

(b) overdesigning the reactor so that the reactor can be operated with $T_{R.O}<978$ K at the worst case conditions.

(c) introducing a new manipulated variable to control $T_{R.O}$.

Each of the above choices will impose a certain economic penalty. By evaluating the economic penalty, we can find the cheapest one. In this work, we assume that choice (c) imposes the least economic penalty. Thus, a new manipulated variable which can affect $T_{R.O}$ should be introduced. For simplicity, we assume a cooling jacket around the reactor

can be used. Therefore, the new manipulated variable introduced should be the flow rate of cooling water (denoted as F_{Water}) through the jacket. Now, the reactor system becomes structurally controllable.

(2) Separation system

The separation system has two control objectives (i.e., P_B and y_{PH}) which are introduced at the upper level (i.e., level 2). But, there are no new control objectives introduced at the current level. Thus, the separation system is still controllable if nonlocal control is accepted. If local control must be applied, the controllability test of the separation system is needed because local control introduces new manipulated variables to the separation system, and new control objectives to its upstream system (i.e., reactor system).

When the separation system uses local control, the manipulated variables F_G (i.e., the fresh hydrogen flow rate) and F_{FT} (i.e., the fresh feed rate of toluene) cannot be used to directly control P_B and y_{PH}. We choose the component flows of hydrogen (f_{H_2}) and toluene ($f_{C_7H_8}$) of the feed stream to the separation system as new manipulated variables.

The dynamic modes of the separation system can be described as:

$$\dot{P}_B = f(P_B, f_{H_2}, f_{C_7H_8}) \tag{3-22}$$

$$\dot{y}_{PH} = f(y_{PH}, f_{H_2}, f_{C_7H_8}) \tag{3-23}$$

By checking the generic rank of this system, we can find the separation system is controllable.

(3) Reactor system

The newly introduced manipulated variables of the separation system are also new control objectives of the reactor system. Thus, we need to reexamine the controllability of the reactor system. The dynamic equations of the reactor system can be given as follows:

$$\dot{x} = f(x, F_G, F_{FT}, F_{Heating}) \tag{3-24}$$

$$\dot{T}_{R,O} = f(T_{R,O}, x, F_G, F_{FT}, F_{Water}) \tag{3-25}$$

$$\dot{r}_{R,I} = f(r_{R,I}, F_G, F_{FT}, R_G) \tag{3-26}$$

$$\dot{f}_{H_2} = f(f_{H_2}, x, F_G, F_{FT}) \tag{3-27}$$

$$\dot{f}_{C_7H_8} = f(f_{C_7H_8}, x, F_G, F_{FT}) \tag{3-28}$$

$$\boldsymbol{x} = \begin{pmatrix} x \\ T_{R,O} \\ r_{R,I} \\ f_{H_2} \\ f_{C_7H_8} \end{pmatrix}, \quad \boldsymbol{u} = \begin{pmatrix} F_G \\ F_{FT} \\ R_G \\ F_{Heating} \\ F_{Water} \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} x \\ T_{R,O} \\ r_{R,I} \\ f_{H_2} \\ f_{C_7H_8} \end{pmatrix}$$

$$\boldsymbol{A} = \begin{pmatrix} x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 \\ x & 0 & 0 & x & 0 \\ x & 0 & 0 & 0 & x \end{pmatrix}, \qquad \boldsymbol{B} = \begin{pmatrix} x & x & 0 & x & 0 \\ x & x & 0 & 0 & x \\ x & x & x & 0 & 0 \\ x & x & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{C} = \begin{pmatrix} x & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 \\ 0 & 0 & 0 & x & 0 \\ 0 & 0 & 0 & 0 & x \end{pmatrix}, \qquad \boldsymbol{D} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{S}_R = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix} = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & u_1 & u_2 & u_3 & u_4 & u_5 \end{matrix} \\ \left(\begin{array}{ccccc:ccccc} x & 0 & 0 & 0 & 0 & x & x & 0 & x & 0 \\ x & x & 0 & 0 & 0 & x & x & 0 & 0 & x \\ 0 & 0 & x & 0 & 0 & x & x & x & 0 & 0 \\ x & 0 & 0 & x & 0 & x & x & 0 & 0 & 0 \\ x & 0 & 0 & 0 & x & x & x & 0 & 0 & 0 \\ \hdashline x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 \end{array}\right) \end{array} \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{matrix} \tag{3-29}$$

$\boldsymbol{S}_R$ can be reordered into the following form:

$$\boldsymbol{S}_R \sim \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & u_1 & u_2 & u_4 & u_3 & u_5 \end{matrix} \\ \left(\begin{array}{ccccc:ccccc} x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 \\ \hdashline x & 0 & 0 & x & 0 & x & x & 0 & 0 & 0 \\ x & 0 & 0 & 0 & x & x & x & 0 & 0 & 0 \\ x & 0 & 0 & 0 & 0 & x & x & x & 0 & 0 \\ 0 & 0 & x & 0 & 0 & x & x & 0 & x & 0 \\ x & x & 0 & 0 & 0 & x & x & 0 & 0 & x \end{array}\right) \end{array} \begin{matrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ x_4 \\ x_5 \\ x_1 \\ x_3 \\ x_2 \end{matrix}$$

Because $\boldsymbol{S}_R$ has no dilations, $\boldsymbol{S}_R$ is still structurally controllable after two new control objectives (i.e., f_{H_2} and $f_{C_7H_8}$) are introduced by the local control of the separation system.

(4) Composite system

We already design both the reactor system and the separation system to be controllable. Now, we need to examine if the composite system that consists of the reactor system and the separation system is still controllable. The dynamic modes of the composite system can be described as follows:

$$\dot{f}_{H_2} = f(f_{H_2}, x, F_G, F_{FT}) \tag{3-30}$$

$$\dot{f}_{C_7H_8} = f(f_{C_7H_8}, x, F_G, F_{FT}) \tag{3-31}$$

$$\dot{x} = f(x, F_G, F_{FT}, F_{Heating}) \tag{3-32}$$

$$\dot{T}_{R,O} = f(T_{R,O}, x, F_G, F_{FT}, f_{Water}) \tag{3-33}$$

$$\dot{r}_{R,I} = f(r_{R,I}, F_G, F_{FT}, R_G) \tag{3-34}$$

$$\dot{P}_B = f(P_B, f_{H_2}, f_{C_7H_8}) \tag{3-35}$$

$$\dot{y}_{PH} = f(y_{PH}, f_{H_2}, f_{C_7H_8}) \tag{3-36}$$

$$\boldsymbol{x} = \begin{pmatrix} f_{H_2} \\ f_{C_7H_8} \\ x \\ T_{R,O} \\ r_{R,I} \\ P_B \\ y_{PH} \end{pmatrix}, \quad \boldsymbol{u} = \begin{pmatrix} F_G \\ F_{FT} \\ R_G \\ F_{Heating} \\ F_{Water} \\ f_{H_2} \\ f_{C_7H_8} \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} f_{H_2} \\ f_{C_7H_8} \\ x \\ T_{R,O} \\ r_{R,I} \\ P_B \\ y_{PH} \end{pmatrix}$$

$$A=\begin{pmatrix} x & 0 & x & 0 & 0 & 0 & 0 \\ 0 & x & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 \\ x & x & 0 & 0 & 0 & x & 0 \\ x & x & 0 & 0 & 0 & 0 & x \end{pmatrix}, \qquad B=\begin{pmatrix} x & x & 0 & 0 & 0 & x & 0 \\ x & x & 0 & 0 & 0 & 0 & x \\ x & x & 0 & x & 0 & 0 & 0 \\ x & x & 0 & 0 & x & 0 & 0 \\ x & x & x & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \end{pmatrix}$$

$$C=\begin{pmatrix} x & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{pmatrix}, \qquad D=\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$S_{L3}=\begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & u_1 & u_2 & u_3 & u_4 & u_5 & u_6 & u_7 \end{matrix} \\ \left(\begin{array}{ccccccc:ccccccc} x & 0 & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & x & 0 \\ 0 & x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 & x \\ 0 & 0 & x & 0 & 0 & 0 & 0 & x & x & 0 & x & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 & 0 & x & x & 0 & 0 & x & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 & x & x & x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & x & x \\ x & x & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & x & x \\ \hdashline x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right) \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{matrix} \end{array} \tag{3-37}$$

S_{L3} is reordered as follows:

$$
S_{L3} \sim
\begin{array}{c}
\begin{array}{ccccccc|ccccccc}
x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & u_6 & u_7 & u_1 & u_2 & u_4 & u_3 & u_5
\end{array} \\
\left(\begin{array}{ccccccc:ccccccc}
x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hdashline
x & x & 0 & 0 & 0 & x & 0 & x & x & 0 & 0 & 0 & 0 & 0 \\
x & x & 0 & 0 & 0 & 0 & x & x & x & 0 & 0 & 0 & 0 & 0 \\
x & 0 & x & 0 & 0 & 0 & 0 & x & 0 & x & x & 0 & 0 & 0 \\
0 & x & x & 0 & 0 & 0 & 0 & 0 & x & x & x & 0 & 0 & 0 \\
0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & x & x & x & 0 & 0 \\
0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & x & x & 0 & x & 0 \\
0 & 0 & x & x & 0 & 0 & 0 & 0 & 0 & x & x & 0 & 0 & x
\end{array}\right)
\end{array}
\begin{array}{c}
y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ \hdashline x_6 \\ x_7 \\ x_1 \\ x_2 \\ x_3 \\ x_5 \\ x_4
\end{array}
$$

Because there exist no dilations in the reordered structural matrix, S_{L3} has a full generic rank, i.e., the composite system at Level 3 is structurally controllable.

Level 4. General structure of separation system

(1) General structure

As stated in the process design of Level 4, a flash drum is used to achieve a phase split. The flash drum has no control objectives to maintain except that the temperature of the feed stream to the flash drum which affects the split between vapor and liquid in the flash drum. Because the energy integration system is not considered until Level 5, the control of the feed stream temperature of the flash drum will not be taken into account until Level 5. The liquid separation system shown in Figure 3-4 also does not bring new control objectives. Therefore, the general structure of the separation system shown in Figure 3-4 is controllable.

(2) Liquid separation system

The liquid separation system consists of three distillation columns in a direct sequence as shown in Figure 3-5. Each column has two new control objectives, i.e., the top and bottom fractional recoveries ($x_{D,i}, x_{B,i}$, i = 1, 2, 3). Two new manipulated variables are also available for each column, i.e., reflux ratio (R_i) and vapor rate (V_i). The dynamic equations of the liquid separation system can be given as follows:

$$\dot{x}_{D,1} = f(x_{D,1}, x_{B,1}, R_1, V_1) \tag{3-38}$$

$$\dot{x}_{B,1} = f(x_{B,1}, x_{D,1}, R_1, V_1) \tag{3-39}$$

$$\dot{x}_{D,2} = f(x_{D,2}, x_{B,2}, R_2, V_2) \tag{3-40}$$

$$\dot{x}_{B,2} = f(x_{B,2}, x_{D,2}, R_2, V_2) \tag{3-41}$$

$$\dot{x}_{D,3} = f(x_{D,3}, x_{B,3}, R_3, V_3) \tag{3-42}$$

$$\dot{x}_{B,3} = f(x_{B,3}, x_{D,3}, R_3, V_3) \tag{3-43}$$

$$\boldsymbol{x} = \begin{pmatrix} x_{D,1} \\ x_{B,1} \\ x_{D,2} \\ x_{B,2} \\ x_{D,3} \\ x_{B,3} \end{pmatrix}; \qquad \boldsymbol{u} = \begin{pmatrix} R_1 \\ V_1 \\ R_2 \\ V_2 \\ R_3 \\ V_3 \end{pmatrix}; \qquad \boldsymbol{y} = \begin{pmatrix} x_{D,1} \\ x_{B,1} \\ x_{D,2} \\ x_{B,2} \\ x_{D,3} \\ x_{B,3} \end{pmatrix}$$

$$\boldsymbol{A} = \begin{pmatrix} x & x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & x & x \end{pmatrix}; \qquad \boldsymbol{B} = \begin{pmatrix} x & x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & x & x \end{pmatrix}$$

$$\boldsymbol{C} = \begin{pmatrix} x & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 \\ 0 & 0 & 0 & 0 & 0 & x \end{pmatrix}; \qquad \boldsymbol{D} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{S}_{LS} = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{matrix} \\ \left(\begin{array}{cccccc:cccccc} x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x \\ \hdashline x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right) \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{matrix} \end{array} \tag{3-44}$$

$\boldsymbol{S}_{LS}$ can be reordered into the following form:

$$\boldsymbol{S}_{LS} \sim \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{matrix} \\ \left(\begin{array}{cccccc:cccccc} x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 \\ \hdashline x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x & 0 & 0 \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & x & x & 0 & 0 & 0 & 0 & x & x \end{array}\right) \begin{matrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{matrix} \end{array}$$

Obviously, $\boldsymbol{S}_{LS}$ has no dilations. Thus, $\boldsymbol{S}_{LS}$ has a full generic rank. Therefore, the liquid separation system is structurally controllable. The flowsheet obtained for Level 4 is shown in Figure 3-5.

Level 5. Heat exchanger networks

After the designs of the reactor and separation systems have been finished, the last design task is to consider the energy integration system. Because a flash drum is used for phase split, we need a cooler to control the temperature of the flash drum feed stream so that the splits between vapor and liquid in the flash drum can be decided. We use a furnace to heat the reactor feed and a cooling jacket to cool the reactor outlet. Based on these requirements, we can design the heat exchanger network of the HDA process by using a systematic method (e.g., pinch design method). The inlet stream of the reactor is first preheated in the exchanger 1 by the cooling flow from the jacket, and then is heated again in the exchanger 2 by the reactor effluent. It is further heated to above 895 K by the furnace. The flowsheet can be shown in Figure 3-6.

We have three control objectives to maintain at this level. First, we need to cool the reactor effluent ($T_{R.E}$) rapidly to 895 K in order to prevent coking in the heat exchangers following the reactor. Secondly, the cooling water return temperature ($T_{WC.O}$) of the water cooler must be kept below 322 K to prevent scaling in the closed cooling water system. Thirdly, the flash drum temperature (T_{Flash}) should be kept at its optimum value, because high flash drum temperature incurs large purge losses of aromatics and low flash drum temperature increases cooling water costs (Fisher *et al.*, 1988). We also have three manipulated variables, i.e., the quench flow rate (F_{Quench}), the cooling water flow rate (F_{WC}) and the by-pass flow rate (F_{Bypass}) of the water cooler.

The dynamic models of the heat exchanger network can be described by the following equations:

$$\dot{T}_{R,E} = f(T_{R,E}, F_{Quench}) \tag{3-45}$$

$$\dot{T}_{WC,O} = f(T_{WC,O}, T_{Flash}, F_{WC}) \tag{3-46}$$

$$\dot{T}_{Flash} = f(T_{WC,O}, T_{Flash}, T_{R,E}, F_{Bypass}) \tag{3-47}$$

$$\boldsymbol{x} = \begin{pmatrix} T_{R,E} \\ T_{WC,O} \\ T_{Flash} \end{pmatrix}; \qquad \boldsymbol{u} = \begin{pmatrix} F_{Quench} \\ F_{WC} \\ F_{Bypass} \end{pmatrix}; \qquad \boldsymbol{y} = \begin{pmatrix} T_{R,E} \\ T_{WC,O} \\ T_{Flash} \end{pmatrix}$$

$$\boldsymbol{A} = \begin{pmatrix} x & 0 & 0 \\ 0 & x & x \\ x & x & x \end{pmatrix}; \qquad \boldsymbol{B} = \begin{pmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{pmatrix}$$

$$\boldsymbol{C} = \begin{pmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{pmatrix}; \qquad \boldsymbol{D} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{S}_{HEN} = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & u_1 & u_2 & u_3 \end{matrix} \\ \left(\begin{array}{ccc:ccc} x & 0 & 0 & x & 0 & 0 \\ 0 & x & x & 0 & x & 0 \\ x & x & x & 0 & 0 & x \\ \hdashline x & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 \end{array}\right) \begin{matrix} x_1 \\ x_2 \\ x_3 \\ y_1 \\ y_2 \\ y_3 \end{matrix} \end{array} \tag{3-48}$$

$$\begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & u_1 & u_2 & u_3 \end{matrix} \\ \left(\begin{array}{ccc:ccc} x & 0 & 0 & x & 0 & 0 \\ 0 & x & x & 0 & x & 0 \\ x & x & x & 0 & 0 & x \\ \hdashline x & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 \end{array}\right) \begin{matrix} x_1 \\ x_2 \\ x_3 \\ y_1 \\ y_2 \\ y_3 \end{matrix} \end{array} \sim \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & u_1 & u_2 & u_3 \end{matrix} \\ \left(\begin{array}{cccccc} x & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 & 0 \\ x & 0 & 0 & x & 0 & 0 \\ 0 & x & x & 0 & x & 0 \\ x & x & x & 0 & 0 & x \end{array}\right) \begin{matrix} y_1 \\ y_2 \\ y_3 \\ x_1 \\ x_2 \\ x_3 \end{matrix} \end{array}$$

Obviously, S_{HEN} has a full generic rank. Thus, the heat exchanger network is also structurally controllable.

3.8. Conclusions

A hierarchical method is presented for the integrated process design and plant-wide control. The design goal at each design level is to generate processes which are both economical and controllable. By incorporating structural controllability test into each design level, we can restore the controllability of uncontrollable processes at early stages. The proposed method requires that control problems be solved at the current level, so that the control system synthesis of a complete plant becomes easier and less complex than considering the whole plant at the same level.

Comparing with the method by Fisher *et al.* (1988), dynamic models instead of steady state models are used in examining the controllability of processes in this work. The controllability test is more rigorous and more reliable because it is in terms of the criteria of structurally state and functional controllability. Secondly, the approach developed in this work guarantees that each subsystem and the composite system at each hierarchical design level are controllable by introducing new manipulated variables or ignore uncritical control objectives if the considered systems are not controllable. Thirdly, local control and the controllability of composite systems are considered so that our method can avoid uncontrollable processes caused by introducing local control.

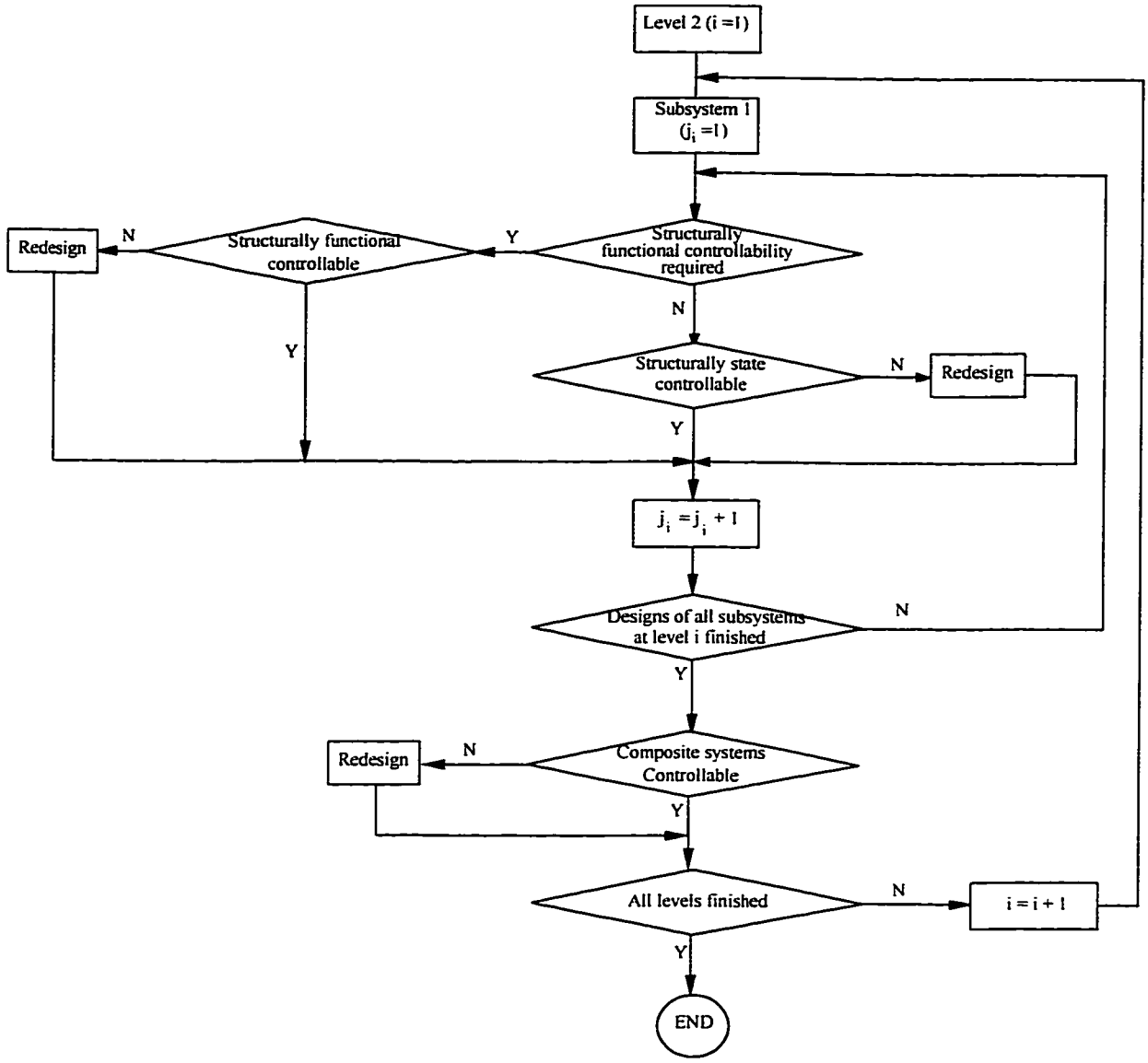


Figure 3-1. Hierarchical procedures for the integrated process design and plant-wide control.

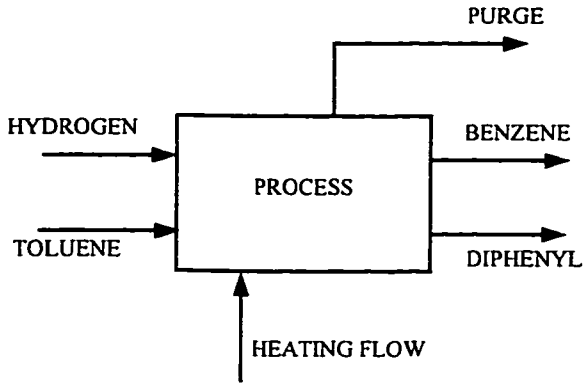


Figure 3-2. Input-output structure.

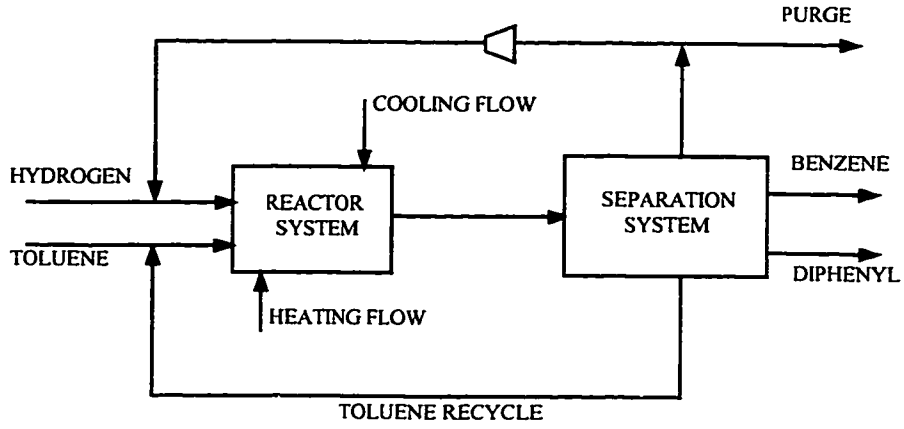


Figure 3-3. Recycle structure.

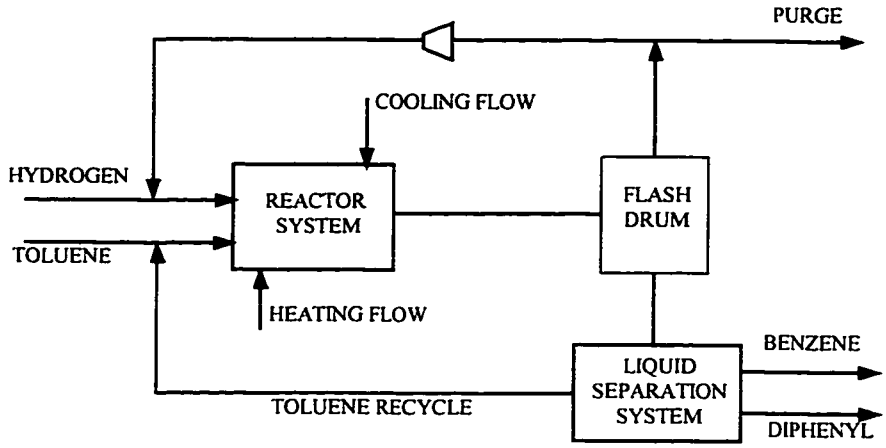


Figure 3-4. General structure of the separation system.

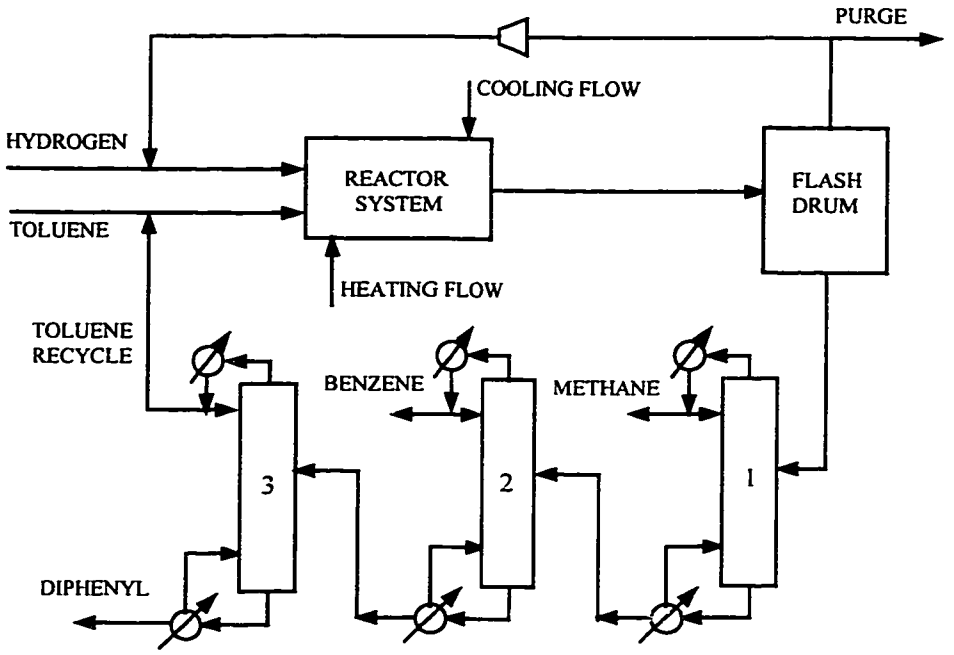


Figure 3-5. Liquid separation system.

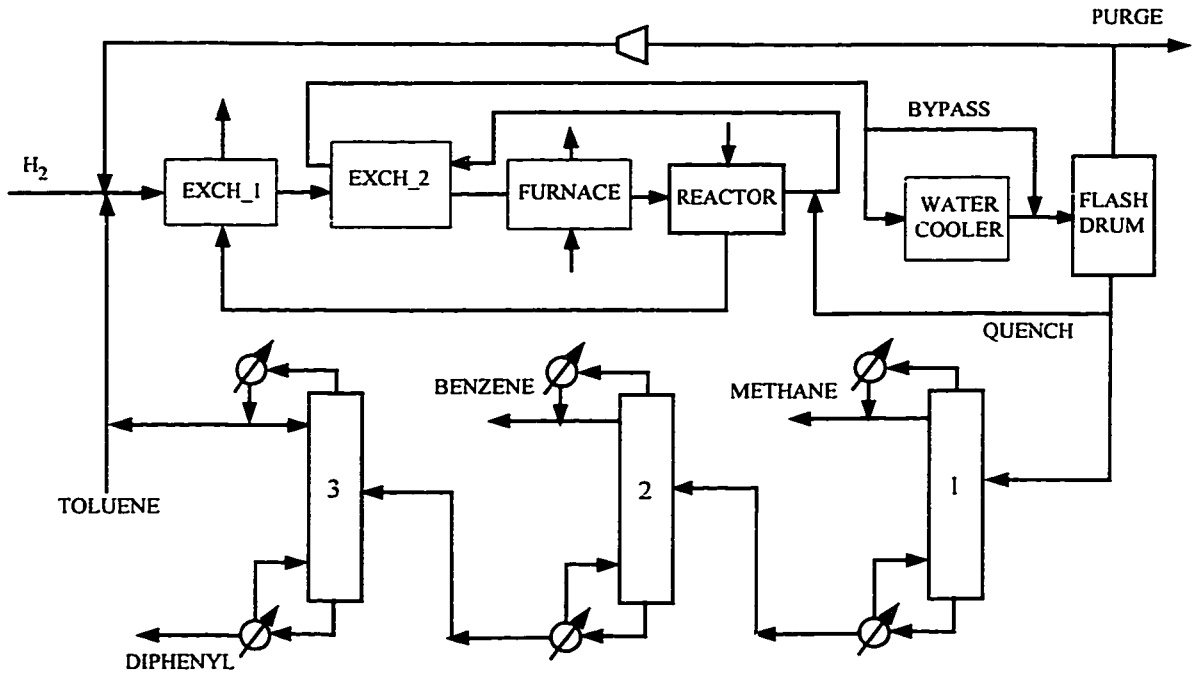


Figure 3-6. Final flowsheet.

3.9. Literature Cited

Daoutidis, P., and C. Kravaris, "Structural Evaluation of Control Configurations for Multivariable Nonlinear Processes," *Chem. Eng. Sci.*, **47**, 1091 (1992).

Douglas, J.M., "A Hierarchical Decision Procedure for Process Synthesis," *AIChE J.*, **31**, 353 (1985).

Douglas, J.M., *Conceptual Design of Chemical Processes*, McGraw-Hill, New York (1988).

Douglas, J.M., "Process Synthesis for Waste Minimization," *Ind. Eng. Chem. Res.*, **31**, 238 (1992).

Douglas, J.M., "Synthesis of Separation System Flowsheets," *AIChE J.*, **41**, 2522 (1995).

Fisher, W.R., M.F. Doherty, and J.M. Douglas, "The Interface Between Design and Control. 1. Process Controllability," *Ind. Eng. Chem. Res.*, **27**, 597 (1988).

Georgiou, A., and C.A. Floudas, "Structural Analysis and Synthesis of Feasible Control Systems —Theory and Applications," *Chem. Eng. Res. Dev.*, **69**, 600 (1989).

Johnston, R.D., G.W. Barton, and M.L. Brisk, "Determination of the Generic Rank of Structural Matrices," *Int. J. Contr.*, **40**, 257 (1984).

Johnston, R.D., G.W. Barton, and M.L. Brisk, "Single-Input-Single-Output Control System Synthesis. Part I: Structural Analysis and The Development of Feedback Control Schems", *Comput. Chem. Eng.*, **9**, 547 (1985).

McKenna, T.F., and M.F. Malone, "Polymer Process Design - I. Continuous Production of Chain-Growth Homopolymers," *Comput. Chem. Eng.*, **14**, 1127 (1990).

Morari, M., and G. Stephanopoulos, "Studies in the Synthesis of Control Structures for Chemical Processes. Part II: Structural Aspects and the Synthesis of Alternative Feasible Control Schemes," *AIChE J.*, **26**, 232 (1980).

Nishida, N., G. Stephanopoulos, and A.W. Westerberg, "A Review of Process Synthesis," *AIChE J.*, **27**, 321 (1980).

Ponton, J.W., and D.W. Laing, "A Hierarchical Approach to the Design of Process Control Systems," *Chem. Eng. Res. Des.*, **71**, 181 (1993).

Rajagopal, S., K.M. Ng., and J.M. Douglas, "A Hierarchical Procedure for the Conceptual Design of Solids Processes," *Comput. Chem. Eng.*, **16**, 675 (1992).

Rosenbrock, H.H., *State Space and Multivariable Theory*, John Wiley, Inc., N.Y. (1970).

Russell, L.W., and J.D. Perkins, "Towards a Method for Diagnosis of Controllability and Operability Problems in Chemical Plants," *Chem. Eng. Res. Des.*, **65**, 453 (1987).

Sacerdoti, E.D., "Planning in a Hierarchy of Abstraction Spaces," *Artificial Intelligence*, **5**, 115 (1974).

Stephanopoulos, G., C. Han, A. Linninger, S. Ali, and E. Stephanopoulos, "Concept of ZAP (Zero Avoidable Pollution) in the Synthesis and Evaluation of Batch Pharmaceutical Process," Annual AIChE Meeting, San Fransico, CA (1994).

CHAPTER 4. DETERMINATION OF INDEPENDENT LOOPS IN HEAT EXCHANGER NETWORKS[1]

4.1. Introduction

Various design methods for heat exchanger network (HEN) synthesis have been developed since Masso and Rudd (1969) first formalized the problem of HEN synthesis. HEN design methods can be broadly classified into two categories: algorithmic and systematic evolutionary methods. Despite considerable progress, the algorithmic methods seem as yet to be restricted for direct application to complicated industrial problems (Suaysompol and Wood, 1991). Current industrial practice appears to be dominated by evolutionary methods (Trivedi *et al.*, 1990).

Systematic evolutionary methods apply a series of heuristic rules to generate HENs that feature more units than the targeted minimum number of units. Such networks are often required to be simplified by decreasing the number of units. To reasonably decrease the size of the search space of units for removal, the concept of loop-breaking is widely applied. Hence, the identification of heat load loops is required. It is very difficult to visually locate loops in a complex network, and a systematic method is useful to effectively find the loops.

Loops in HENs can be classified into independent and dependent loops. Dependent loops can be obtained by addition operations on independent loops. Identification of independent loops can be accomplished by two stages: (a) finding a maximum tree for a

[1] A version of this chapter has been published:
Z. Han, J. Zhu, M. Rao and K.T. Chuang, "Determination of Independent Loops in Heat Exchanger Networks," *Chem. Eng. Comm.*, **164**, 191-204 (1998).

given graph, and (b) obtaining a maximum set of independent loops by adding edges to the tree. Welch (1966) used a similar method to find independent loops for linear graphs. In his work, graphs were represented by incidence matrices and the main operation involved is the addition of vectors standing for edges.

Locating independent loops in HENs has been investigated by Pethe *et al.* (1989). Basically, their method is solving a set of linear equations. They also presented an equation to calculate the number of independent loops. In their work, split streams are merged and their effects on the determination of independent loops are not discussed.

In this study, the nonlinear property of graphs derived from HENs is considered. The idea of finding independent loops by generating a maximum tree for a given graph and adding edges to the tree is extended to HENs. The proposed procedure can produce a set of independent loops quite straightforward, and no addition operations are needed. It is proven that the set of independent loops produced by the proposed procedure is a maximum set of the independent loops. A new equation to determine the number of independent loops for HENs is also developed. Discussions about if split streams should be ignored are made, and it is shown that independent loops obtained in this study could be useful to achieve smaller energy penalties when loop-breaking is applied.

4.2. An Alternative Representation of Heat Exchanger Networks

There are several representations for HEN problems. The stream grid representation (Linnhoff and Flower, 1978a,b) has been widely used in the design of HENs. For example, for the HEN shown in Figure 4-1(a), its stream grid representation is given in Figure 4-1(b). The stream grid representation cannot be easily employed to locate heat

load loops. Incidence matrix (Pethe *et al.*, 1989) is a common mathematical representation, but it does not describe the linkage of two parts of a unit on hot and cold streams. Hence, an alternative mathematical representation of HENs is presented in this work. A matrix, called node adjacency matrix, is defined and applied to describe the structure of a HEN. It is used to represent the connection of units in the flow directions of streams in HENs. Units as well as split and mixing points in the network are all considered as nodes. In the node adjacency matrix, both heading row and heading column represent nodes. Thus, the matrix is square. Each entry has a value of +1, -1 or a blank. If node j can directly reach node i in the flow direction of a hot (or cold) stream, then entry (i, j) is filled with +1 (or -1). Otherwise, it is left a blank. For heaters and coolers, their heat loads can be shifted among these heaters (coolers). If two heaters (or coolers) i and j share a cooling (or heating) source, then entry (i, j) [or entry (j, i), in this work, we choose the entry with $i > j$] is filled with -1 (or +1). Otherwise, entries (i, j) and (j, i) are left blanks. In this work, we assume that a single type of hot (cold) utility is used. But, all developed techniques are also valid for cases where utilities have different energy sources. As an illustration, the node adjacency matrix for the network shown in Figure 4-1(a) is given in Table 4-1.

4.3. Preliminaries

In a HEN, a node consists of two parts, i.e., one on a hot stream (hot part) and another on a cold stream (cold part). A node in a network could stand for the hot part, the cold part or both, so could an edge of a node. A node or an edge holds different meanings

in different loops, which can be considered a kind of nonlinear property of graphs. The node adjacency matrix can be used to account for this nonlinear property.

Although a node adjacency matrix represents a directed graph, we always treat it as an undirected graph because we do not need to care for directions in the loop identification. In an undirected graph, both entry (i, j) and entry (j, i) denote the edge connecting node i and node j.

Let m be the dimension of a node adjacency matrix (i.e., the number of nodes) and p be the number of non-zero non-diagonal entries of the matrix, respectively. Then, the graph obtained from the node adjacency matrix possesses m nodes and p edges. If the network has no split streams, then the inequality $m-1 \le p \le 2m$ is valid since the graph is connected and there exist at most two non-zero non-diagonal entries in each row or each column.

It has been mentioned that a unit consists of two parts in its grid representation. Let σ be a loop and $a_i \in \sigma$. Then we use a_i^+ and a_i^- to denote its hot part and cold part. Node a_i is defined complete in σ if the loop passes from a_i^+ to a_i^- or from a_i^- to a_i^+ in the grid representation. Otherwise, it is incomplete. It is obvious that a_i is complete if and only if $A_{a_i,a_{i-1}} \cdot A_{a_{i+1},a_i} = -1$, where $A_{a_i,a_{i-1}}$ and A_{a_{i+1},a_i} are the values of entries of the adjacency matrix. These entries stand for the edges linking node a_{i-1} and node a_i, as well as node a_i and node a_{i+1} in loop σ, respectively. A loop $(a_1 a_2 \cdots a_{p-1})$ is said to be simple if $a_l \neq a_q$ for all $l \neq q$, where $l, q = 1, 2, \cdots, p-1$, except a_i is an incomplete node of this loop and a_i^+ and a_i^- are in this loop.

Loops have the following two properties which are easily verified.

Property 1. If $(a_1 a_2 \cdots a_n)$ is a loop, then $(a_n \cdots a_2 a_1)$ expresses the same loop as $(a_1 a_2 \cdots a_n)$.

Property 2. If $(a_1 a_2 \cdots a_n)$ is a loop and $\{i, j, \cdots, l\}$ is a cyclic permutation of $\{1, 2, \cdots, n\}$, then $(a_i a_j \cdots a_l)$ is the same loop as $(a_1 a_2 \cdots a_n)$.

Definition 1. (independent loop)

Let $\sum = (\sigma_1, \sigma_2, \cdots, \sigma_l)$ be a set of l loops $\sigma_1, \sigma_2, \cdots, \sigma_l$, where, σ_i $(i = 1, 2, \cdots, l)$ is a loop of a connected graph. This set is said to be dependent if there are loops $\sigma_i \in \Sigma$, for all $i \in I$ (I is a set of indices) such that $\sum_{i \in I} \sigma_i = 0$. Otherwise, $\sigma_1, \sigma_2, \cdots, \sigma_l$ are independent, and Σ is a set of independent loops.

4.4. Identification of Independent Loops

Let G be a connected graph. A tree T of G is a connected subset of G if T contains all nodes of G but does not contain any loops. Sometimes, T is called a maximum tree of G since $G \backslash T$ has no nodes. There often exist more than one maximum tree in a graph. However, these maximum trees contain the same number of edges. The following lemma is fundamental in the graph theory (Deo, 1974). Hence, its proof is omitted.

Lemma 1. Let G be a connected graph with m nodes, then its maximum trees all have $(m-1)$ edges.

Definition 2. (maximum set of independent loops)

Let $\Sigma=(\sigma_1, \sigma_2, \cdots, \sigma_l)$ be an independent set of loops, where, $\sigma_i \in G$ $(i=1, 2, \cdots, l)$. Σ is defined to be a maximum set of independent loops of G if adding any loop σ_k of G to Σ, the set $(\sigma_1, \sigma_2, \cdots, \sigma_l, \sigma_k)$ is dependent.

For a connected graph G obtained from the node adjacency matrix of a network, the following procedure can be used to find independent loops involved in the graph.

Procedure A (locating independent loops)

Step 1. Let node 1 be the root of the tree to be constructed.

Step 2. Find a maximum tree T of G with root $\{1\}$.

Step 3. For $l=2, 3, \cdots, m$, find the unique path from root $\{1\}$ to node l in tree T.

Step 4. For every edge $a_{i,j}$ that is not in the tree T, add the edge to the tree and subsequently obtain a loop along paths from root $\{1\}$ to node i and node j. All these loops form a maximum set of independent loops.

It can be proved that Procedure A exactly produces a maximum set of independent loops of G.

Theorem 1. In Step 3 of Procedure A, the path from the root $\{1\}$ to node l is unique in tree T.

Proof. At first, for every node l $(l=2, 3, \cdots, m)$, there must exist a path from root $\{1\}$ to node l since tree T is connected.

If in tree T there exist two different paths $la_m a_{m-1} \cdots a_2 a_1 1$ and $lb_q b_{q-1} \cdots b_2 b_1$ from node 1 to node l, there must exist different nodes in $la_m a_{m-1} \cdots a_2 a_1 1$ and $lb_q b_{q-1} \cdots b_2 b_1 1$.

Without loss of the generality, we can assume $a_m \neq b_p$ (if $a_m = b_p$, we consider paths $a_m a_{m-1} \cdots a_2 a_1 1$ and $b_q b_{q-1} \cdots b_2 b_1$, instead). Then, by comparing these two paths from left to right, we can find an identical node since their roots are the same. Without loss of the generality, we assume that $a_1 = b_2$ is the first pair. Thus, $b_2 b_3 \cdots b_q l a_m a_{m-1} \cdots a_2 a_1$ is a loop in tree T. However, a tree should not contain any loops. The contradiction shows that the path from root $\{1\}$ to node l is unique. ■

Corollary 1. In Step 4 of Procedure A, adding an edge $a_{i,j}$ to tree T leads to a unique loop.

Proof. From Lemma 1, a maximum tree only holds $(m-1)$ edges. Hence, adding an edge that is not in the tree will result in a loop. By Theorem 1, the path from the root to every other node is uniquely determined. By a proof similar to that of Theorem 1, we can verify that the loop is uniquely determined. The details are omitted. ■

Let $\sigma(i,j)$ be the loop obtained by adding $a_{i,j}$ to the tree T. Then $\Sigma = \{\sigma(i,j), a_{i,j} \notin T\}$, i.e., the set of $\sigma(i,j)$'s, has $p-(m-1)$ loops.

Theorem 2. Σ is independent.

Proof. This theorem is verified by contradiction.

For the sake of convenience, we denote $\Sigma = \{\sigma_i ; i = 1, 2, \cdots, p-m+1\}$. Without loss of the generality, if loops σ_1, σ_2 and σ_3 are dependent, i.e., $\sigma_1 + \sigma_2 = \sigma_3$, then σ_1 and σ_2 have common edge(s), and σ_3 consists of all edges of σ_1 and σ_2 except those common edge(s). Suppose σ_i is obtained by adding edge a_i to tree T. Then $a_i \notin T$ for

$i = 1,2,3$. Because $a_3 \notin \sigma_1$ and $a_3 \notin \sigma_2$, $a_3 \notin \sigma_1 \cup \sigma_2$. However $a_3 \in \sigma_3$, the equation $\sigma_1 + \sigma_2 = \sigma_3$ cannot be valid. The contradiction verifies the validness of Theorem 2. ■

Theorem 3. Σ is a maximum set of independent loops.

Proof. It suffices to verify that each loop is a sum of loops in the set $\Sigma = \{\sigma_i; i = 1, 2, \cdots, p - m + 1\}$. Let χ be a loop. Suppose $b_1, b_2, \cdots, b_n$ are all the edges of χ with the property that they are not in the tree T. These b_i's must exist since tree T does not contain any loop. Considering now b_1 which does not belong to T, there exists a loop in $(\sigma_1, \sigma_2, \cdots, \sigma_{p-m+1})$, without loss of the generality, say σ_1 such that $b_1 \in \sigma_1$. The addition operation $\chi+\sigma_1$ is then valid. Note that $\chi+\sigma_1$ is a loop and it only contains $(n-1)$ edges $b_2, \cdots, b_n$ that do not belong to T since σ_1 only has one edge that is not in T. Repeat now this procedure for b_2. Since b_2 is not in T, there must exists a loop, say σ_2 such that $b_2 \in \sigma_2$. $\chi+\sigma_1+\sigma_2$ is still a loop but it only holds $(n-2)$ edges $b_3, \cdots, b_n$ that are not in tree T. Repeating this procedure n times until there is no loop in the sum $\chi + \sigma_1 + \sigma_2 + \cdots + \sigma_n$. But the sum of loops must be a loop. It means $\chi + \sigma_1 + \sigma_2 + \cdots + \sigma_n = 0$, i.e., $\chi = \sigma_1 + \sigma_2 + \cdots + \sigma_n$. The conclusion is implied. ■

4.5. Determination of Number of Independent Loops

4.5.1. Equation derivation

Let κ be the number of independent loops. Then, Theorem 2 and Theorem 3 imply the following equation:

$$\kappa = p - m + 1 \tag{4-1}$$

Equation (4-1) is quite different from that given by Pethe *et al.* (1989) where the number of independent loops depends on the rank of the incidence matrix they defined, whereas Equation (4-1) depends only on the number of nodes and edges.

Let E, C and H be the numbers of heat exchangers, coolers and heaters, respectively. Obviously, $m = E + C + H$ if the network has no split streams. Let n be the number of all streams including cooling and heating streams. Then we have the following equation:

$$p = 2(E + C + H) - n \tag{4-2}$$

Equation (4-2) is easily verified from the fact that each heat exchanger consists of two parts, each part holds one input stream, and all coolers (or heaters) share an identical source. Thus, Equation (4-1) can be rewritten as:

$$\kappa = (E + C + H) - n + 1 \tag{4-3}$$

Denote $U = E + C + H$, then Equation (4-3) leads to:

$$U = n + \kappa - 1 \tag{4-4}$$

Equation (4-4) is the same as the equation given by Linnhoff *et al.* (1979), but is obtained in a different way.

4.5.2. Discussions of κ

We mentioned that Equation (4-3) holds if the networks do not contain split streams. What will happen if the split streams are concerned?

Equation (4-1) is always valid from the knowledge of the graph theory. But, the number of independent loops obtained by Equation (4-1) is larger than that given by

Pethe *et al.* (1989) if the split streams are taken into account. By merging split streams, Pethe *et al.* (1989) treated the networks with split streams the same as those without split stream. Let L stand for the number of independent loops defined by Pethe *et al* (1989). If one split stream is considered, the number of nodes will be increased from m to m+2 and the number of the edges from p to p+3. Thus, the following equation can be easily obtained.

$$\kappa = L + s \tag{4-5}$$

where s is the number of the split streams. Correspondingly, Equation (4-3) is changed into the following equation:

$$\kappa = (E + C + H) + s - n + 1 \tag{4-6}$$

Equation (4-5) or (4-6) shows that if we have s split streams, then the network increases s independent loops. What we are concerned about is whether or not these additional loops are necessary to be considered.

Adopting κ means that we consider the original network straightforward. By traditional approaches, a network with split streams is converted to that without split streams in order to get the nominal loops and then to return to the original network to find loops. Why do we go a round trip and give up the shortcut? It is not only a twist but also may result in confusions as explained below.

Considering the network shown in Figure 4-2(a). This network can be considered as a part of a large network. It can be found that this subnetwork contains 3 independent loops, i.e., (M, 2, 3, 4, 1), (5, S, 6) and (3, 4, 6, 5).

If we ignore the split stream, then the network shown in Figure 4-2(a) becomes what shown in Figure 4-2(b). This network holds two independent loops, i.e., (3, 4) and (5, 6). Loop (3, 4) could represent either loop (3, 2, M, 1, 4) or loop (3, 5, S, 6, 4) in the network shown in Figure 4-2(a). Similarly, loop (5, 6) could represent either loop (5, 3, 2, M, 1, 4, 6) or loop (5, S, 6). We now sum up these two independent loops to locate other dependent loops involved in Figure 4-2(a), then confusions arises. First, because these two independent loops could stand for two different loops, we do not know which loop should be used in the addition operations. Secondly, the summation of these different loops will lead to different results or even without a result. Which result should be the summation of these two loops? Several results are given in Table 4-2 for explanation. The uncertainty makes it difficult to locate loops either manually or by computer. Thus, some special loops for loop-breaking could be unavailable. For example, the removal of node 6 to break loop (3, 4, 6, 5) can result in no energy penalty as shown in Figure 4-3. Loop (3, 4, 6, 5) should be obtained by loop addition operations. However, it is shown in Table 4-2 that this loop could not be found in some cases.

For networks with split streams, Procedure A may find some loops that do not make sense to HENs because these preliminary loops do not contain any complete units. However, these loops are necessary to generate other loops by loop additions. Consider the network shown in Figure 4-2(a) again. A maximum set of independent loops can be described as follows:

(M, 2, 3, 5, S, 6, 4, 1), (M, 2, 3, 4, 1) and (5, S, 6)

where, loop (M, 2, 3, 5, S, 6, 4, 1) is a preliminary loop. If we ignore it, then we cannot obtain loop (3, 4, 6, 5) which should be broken to achieve the possible lower energy penalty. It is better to delete preliminary loops after all loops are located.

The above analyses show that using κ defined in this work, instead of L, can avoid confusions in loop identification and make it easier to find all loops.

When κ is used, we have from Equation (4-6) that

$$\kappa = U + s - n + 1 \tag{4-7}$$

Equation (4-7) can be rewritten as:

$$U = n + \kappa - s - 1 \tag{4-8}$$

Therefore, Equation (4-8), instead of Equation (4-4), is used to calculate the number of units involved in HENs when the number of independent loops determined in this work is applied.

4.6. Conclusions

The concept of heat load loops is important to simplifying heat exchanger networks. It is very difficult to identify visually all heat load loops from complex networks. The method proposed for finding independent loops is to find a maximum tree for a given network and add edges to the tree to get a maximum set of independent loops. To implement the method for heat exchanger networks, an alternative representation of heat exchanger networks, i.e., node adjacency matrix, is presented, which provides a basis of the method.

It is verified that Procedure A exactly produces a maximum set of independent loops. The developed method does not need any complicated operations and is easy to implement by hand or by computers.

An equation is also given to determine the number of independent loops. For a network with split streams, the number of independent loops we obtained is different from that given by Pethe *et al.* (1989). From our discussions, we concluded that in order to completely take into account possible loops in loop-breaking of HENs, split streams should not be merged with other streams but should be treated as separated ones.

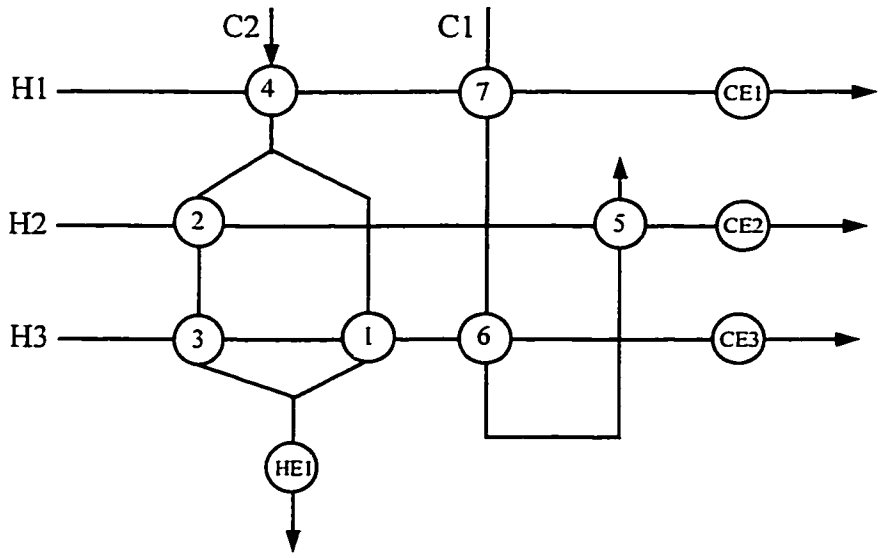


Figure 4-1(a). Flow diagram.

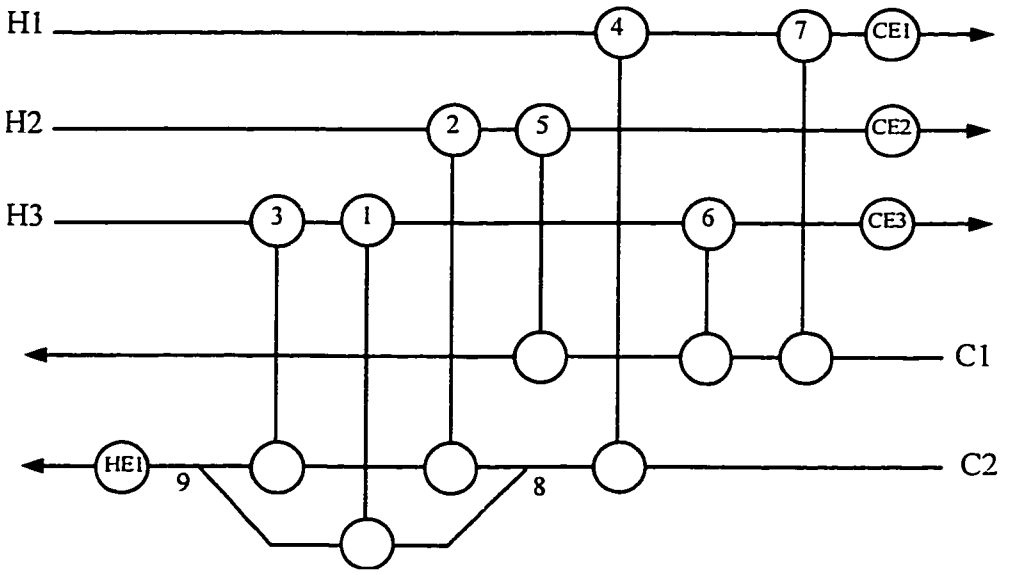


Figure 4-1(b). Stream grid representation (Pethe *et al.*, 1989).

Table 4-1. Node adjacency matrix of the network in Figure 4-1(a).

	1	2	3	4	5	6	7	8	9	HE1	CE1	CE2	CE3
1			1					-1					
2								-1					
3		-1											
4													
5		1				-1							
6	1						-1						
7				1									
8				-1									
9	-1		-1										
HE1									-1				
CE1							1						
CE2					1						1		
CE3						1						1	

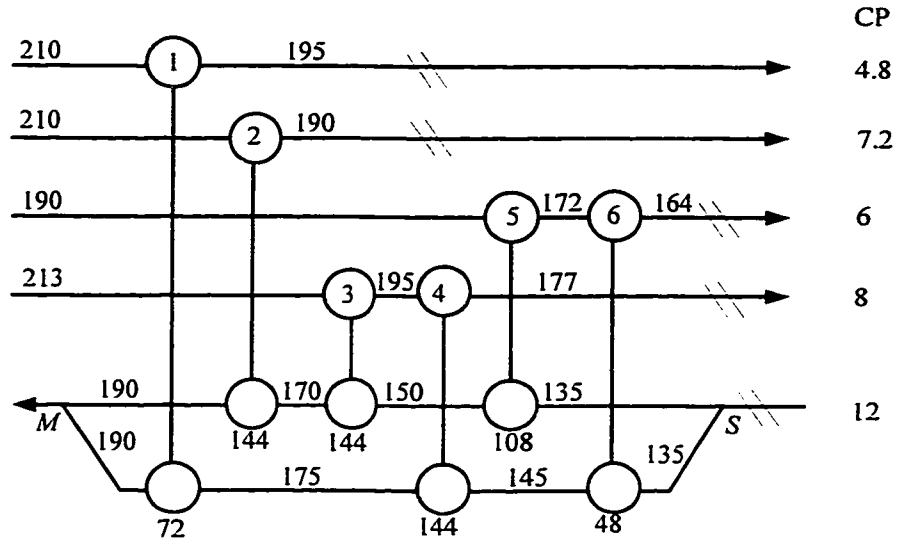


Figure 4-2(a). Original network.

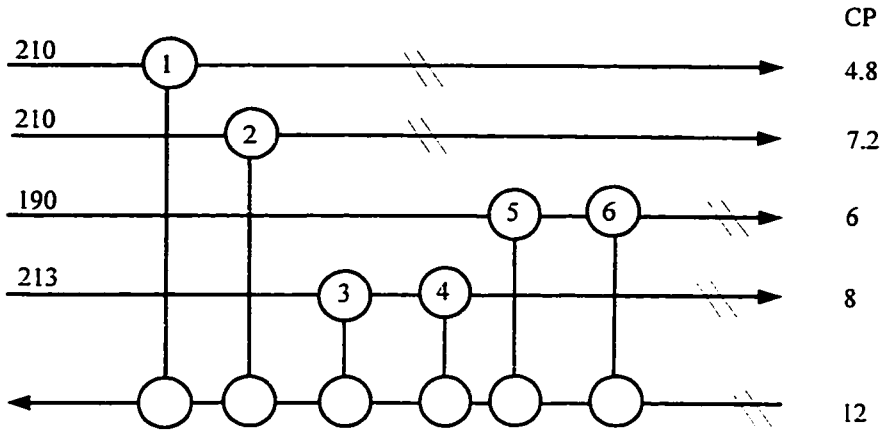


Figure 4-2(b). Simplified network with split stream merged.

Figure 4-2. Explanation of the necessity of considering split streams.

Table 4-2. The addition results of two independent loops.

Loop 1	Loop 2	Loop 1 + Loop 2
(3, 2, M, 1, 4)	(5, S, 6)	no result
(3, 5, S, 6, 4)	(5, 3, 2, M, 1, 4, 6)	(3, 2, M, 1, 4) (i.e., Loop 1)
(3, 5, S, 6, 4)	(5, S, 6)	(3, 4, 6, 5)

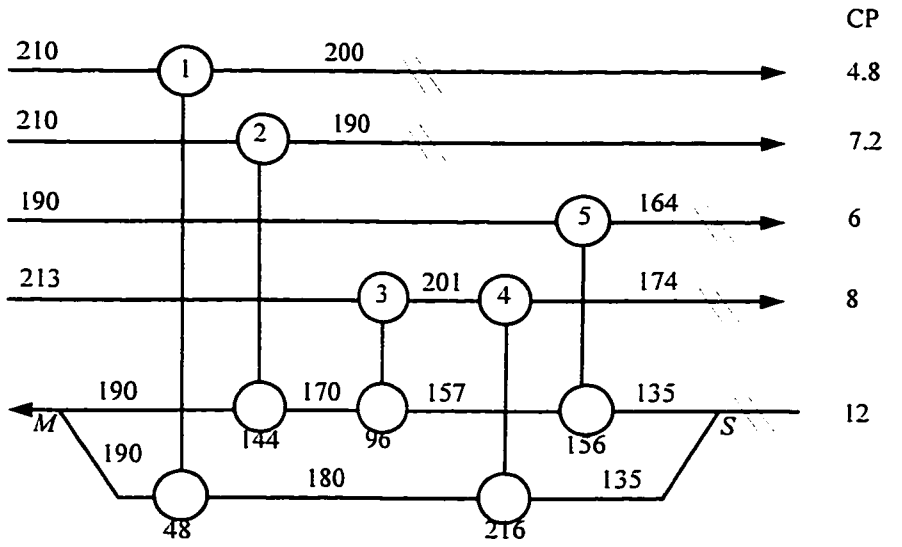


Figure 4-3. Removal of unit 6 from Figure 4-2(a) results in no energy penalty.

4.7. Literature Cited

Deo, N., *Graph Theory with Applications to engineering and computer Science,* Prentice-Hall, Englewood Cliffs, NJ (1974).

Linnhoff, B., and J.R. Flower, "Synthesis of Heat Exchanger Networks — I. Systematic Generation of Energy Optimal Networks," *AIChE J.*, **24**, 633(1978a).

Linnhoff, B., and J.R. Flower, "Synthesis of Heat Exchanger Networks — II. Evolutionary Generation of Networks with Various Criteria of Optimality," *AIChE J.*, **24**, 642 (1978b).

Linnhoff, B., D. R. Mason, and I. Wardle, "Understanding Heat Exchanger Networks," *Comput. Chem. Eng.*, **3**, 295 (1979).

Masso, A.H., and D.F. Rudd, "The Synthesis of Systems Design — II: Heuristic Structuring," *AIChE J.*, **15**, 10 (1969).

Pethe, S., R. Singh, and F.C. Knopf, "A Simple Technique for Locating Loops in Heat Exchanger Networks," *Comput. Chem. Eng.*, **13**, 859 (1989).

Suaysompol, K., and R.M. Wood, "The Flexible Pinch Design Method for Heat Exchanger Networks — Part I. Heuristic Guidelines for Free Hand Designs," *Trans. Inst. Chem. Eng.*, **69**, 458 (1991).

Trivedi, K.K., B.K. O'Neill, J.R. Roach, and R.M. Wood, "Systematic Energy Relaxation in MER Heat Exchanger Networks," *Comput. Chem. Eng.*, **14**, 601 (1990).

Welch, J. T. Jr., "A mechanical Analysis of the Cyclic Structure of Undirected Linear Graphs," *J. ACM*, **13**, 205 (1966).

CHAPTER 5. IDENTIFICATION OF HEAT LOAD LOOPS AND DOWNSTREAM PATHS IN HEAT EXCHANGER NETWORKS[1]

5.1. Introduction

Breaking different loops could produce different results with respect to the energy penalty or the trade-off between capital and energy costs. To achieve the optimal results, it is necessary to take all loops into account. Although it is difficult to consider all loops at the same time because there often exist many loops in networks, it is possible to consider all loops of the same level in simplifying networks. Therefore, a simple method to find all loops is useful.

It is known from linear graph theory that if two loops have edges in common, then one or more new loops can be obtained by deleting the common edges. This operation can be called loop addition. A promising approach to locating all loops is to find independent loops first, and then to obtain all loops by loop additions rather than by search. Let κ and m be the numbers of independent loops and nodes in a heat exchanger network (HEN), respectively. Searching loops using node-by-node techniques may require $O(2^m)$ operations. On the other hand, generating loops by loop additions needs $O(2^\kappa)$ operations. Because $m > \kappa$, locating loops by loop additions is much more efficient. Independent loops of a network can be identified by two stages: (a) finding a maximum tree for the network, and (b) adding edges to the tree to get a maximum set of independent loops. To obtain all loops by loop additions, Welch (1966) and Gibbs

[1] A version of this chapter has been published:
J. Zhu, Z. Han, M. Rao and K.T. Chuang, "Identification of Heat Load Loops and Downstream Paths in Heat Exchanger Networks," *Can. J. Chem. Eng.*, 74, 876-882 (1996).

(1969) proposed methods for linear graphs. Their method used incidence matrices to represent linear graphs, and applied the vector addition of modulo 2 to delete common edges.

Most of existing design techniques generate networks which are assumed to be operated under constant stream flowrates and supply/target temperatures. This assumption is almost impossible in an industrial environment where parameter variation is inevitable because of the changes in the operating and economic environments of a process. Thus, new synthesis techniques should, in addition, cope with the problem of operability, i.e., the networks are still feasible and efficient when these parameters vary within pre-specified ranges (Calandranis and Stephanopoulos, 1986; Huang and Fan, 1992). To address the trade-off between flexibility and economics at the design stage using evolutionary methods, the concept of downstream paths (Linnhoff and Kotjabasakis, 1986) is useful. It is necessary to identify downstream paths in HENs. Techniques for finding downstream paths are required, especially for complicated networks.

In this study, detailed procedures to determine maximum trees and maximum sets of independent loops are presented. To get all dependent loops by loop additions, loop additions for common complete nodes and common edges are formalized. Two-level binary trees are applied to represent loops in order to check whether or not additions of two loops can produce one or more new loops and to remove common parts of two loops to generate new loops. Procedures are proposed for loop additions and for generating all loops from a maximum set of independent loops. By slightly modifying the procedure of

determining maximum trees, the procedure can be used to find downstream paths with the help of stream table which is introduced to represent the matching between hot and cold streams as well as the locations of splitting and mixing points in networks.

5.2. Identification of Heat Load Loops

In the previous chapter, a general procedure for finding independent loops is given. In this section, detailed information how to implement the procedure is presented.

5.2.1. Determining maximum trees

The procedure of identifying a maximum tree is carried out recursively. Let S be a set of nodes of a connected graph G, and n be the number of nodes in G. At first, S only contains the root of the tree. In the ith recursive step, nodes which can be reached from one node of S are added to S. The procedure stops when S contains all nodes of G. This procedure has at most $(n\text{-}1)$ recurrences.

Procedure A (finding a maximum tree)

Step 1. Define any unit in a network as a root. We choose unit 1 in this work as the root, i.e. $S = \{1\}$.

Step 2. For each node denoted as i in S, find node j, if it exists, that can be reached directly from node i, i.e., entry a_{ji} or a_{ij} of the node adjacency matrix of the network is not blank. Add node j to S and record the pair (j, i).

Step 3. If S contains all nodes of G, then go to Step 4. Otherwise, go to Step 2.

Step 4. For every node j (j = 2, 3,..., n) in G, if the last element i of the pair (j, i) recorded in Step 2 is not equal to 1, then find the pair (i,k) and extend the pair (j, i) to a chain (j, i, k). If k is not 1, then find pair (k,l) and extend the chain (j, i, k) to (j, i, k, l). The procedure continues until the last element is 1.

The procedure will be completed by at most $O(n^2)$ steps. The following two remarks give further explanations of Procedure A:

(1) The first elements of the pairs obtained in Step 2 are different, so are the first elements of the chains obtained in Step 4.

(2) Each chain obtained in Step 4 expresses the path from the root {1} to the first node of the chain. The chains are uniquely determined, so are the paths.

For the network shown in Figure 5-1, the pairs found in Step 2 of Procedure A are as follows:

(3, 1), (8, 1), (6, 1), (9, 1), (2, 3), (4, 8), (7, 6), (5, 6), (CE3, 6), (HE1, 9),

(CE1, 7), (CE2, 5)

The chains obtained in Step 4 are:

(3, 1), (8, 1), (6, 1), (9, 1), (2, 3, 1), (4, 8, 1), (7, 6, 1), (5, 6, 1), (CE3, 6, 1),

(HE1, 9, 1), (CE1, 7, 6, 1), (CE2, 5, 6, 1)

By using the pairs found in Step 2 of Procedure A, we can draw the tree as shown in Figure 5-2.

5.2.2. Identifying a maximum set of independent loops

If there exist edges that are not included in the tree, then adding one of these edges to the tree must result in a loop. Adding all edges to the tree will result in a maximum set of independent loops. The procedure for locating independent loops is described as follows:

Procedure B (finding a maximum set of independent loops)

Step 1. Let (j_l, i_l), l = 1, 2, ... , L, be the edges that are not included in the tree. Denote $E_l = (j_l, i_l)$.

Step 2. For E_l, find two chains starting with j_l and i_l that are obtained in Step 4 of Procedure A. Discover the first identical element from left to right in these two chains. If g_l is the first identical element, then $(j_l, \cdots, g_l, \cdots, i_l)$ is an independent loop.

Step 3. If $l = L$, then all independent loops are found; otherwise, let l=l+1 and go to *Step 2.*

For the network shown in Figure 5-1, the edges that do not appear in the tree obtained in Procedure A are (2, 8), (5, 2), (7, 4), (9, 3), (CE2, CE1) and (CE3, CE2). By adding these edges to the tree shown in Figure 5-2, a maximum set of independent loops can be found by Procedure B as follows:

(2,3,1,8), (5,6,1,3,2), (7,6,1,8,4), (9,1,3), (CE2,5,6,7,CE1), (CE3,6,5,CE2).

A FORTRAN program to implement Procedures A and B was developed, which can be used to efficiently find independent loops of complicated networks.

5.3. Loop Additions

For graphs derived from HENs, there exists a nonlinear property of graphs as discussed in the previous chapter. Thus, even when two loops have no common edges but have common complete nodes, the combination of these two loops can still produce one or more new loops, which can be considered another type of loop addition.

For example, loops (3, 1, 9) and (1, 6, 5, 2, 8) in the network shown in Figure 5-1 have no edges in common, but share a common complete node (i.e., node 1), deleting the common part produces a new loop (3, 9, 1, 8, 2, 5, 6, 1).

5.3.1. Types of loop additions

From the above analyses, two types of loop additions are involved in generating loops from a set of independent loops.

Type 1 Addition of common edges

If two loops have edges in common, the addition of these two loops is to remove the common edges to generate one or more new loops. It is called the addition of common edges.

Type 2 Addition of common complete nodes

If two loops hold a common complete node, the addition of these two loops is to remove the linkage between the hot part and the cold part of the common complete node. Thus, the original complete node becomes an incomplete one in the new loop to be produced by loop additions. This operation is called the addition of common complete nodes.

It should be pointed out that the addition of common edges is superior to the addition of common nodes, i.e., if one vertex of common edges is complete, the addition of common edges should be performed first. For example, loop (CE3, 6, 5, CE2) and loop (1, 6, 7, CE1, CE2, 5, 2, 3) of Figure 5-1 have common nodes 5, CE2 and 6. There exists a common edge between nodes 5 and CE2. The addition of common edges removes the common edge (5, CE2). Node 6 is a common complete node. The addition of common complete nodes changes node 6 from a complete node in these two loops to an incomplete node in the new loop. After conducting two types of additions, a new loop (3, 1, 6, CE3, CE2, CE1, 7, 6, 5, 2) can be produced.

5.3.2. Binary trees for loop additions

To produce loops by loop additions, an effective technique is needed to implement the above two types of additions. In this study, a set of two-level binary trees is adopted to represent a loop (all binary trees mentioned later in this work are two-level binary trees). With binary trees, it is easy to conduct loop additions and to check whether or not loop additions can produce one or more new loops. To use a set of binary trees to represent a loop, all nodes of the loop are used as roots of the set of binary trees. Leaf nodes of a root (node a_i) are the two nodes preceding and succeeding to node a_i in the loop. For example, loop (9, 1, 3) can be represented by a set of binary trees as shown in Figure 5-3.

After two sets of binary trees are established for two loops, common leaf nodes of identical roots in two sets of binary trees representing two loops should be removed. It means that if two loops have edges in common, then some roots of binary trees must

contain zero leaf nodes. For example, when this step is finished, the two sets of binary trees representing loop (9, 1, 3) and loop (3, 2, 5, 6, 1) of Figure 5-1 are shown in Figure 5-4.

For convenience, a non-common complete node is used as the starting node and its one non-zero leaf node as the second node of the new loop to be generated. Assuming that nodes a_{i-1} and a_i are the last two nodes in the set of nodes already found for the new loop, and a_{i-1} and a_i are located in a set of binary trees representing one of two loops (e.g., loop 1), then the above two types of loop additions for loop 1 and loop 2 can be implemented as follows:

(i) node a_i is not a common complete node of loop 1 and loop 2.

If in the set of binary trees representing loop 1, root a_i has two non-zero leaf nodes, and one leaf node is a_{i-1}, then the other leaf node is an additional new node a_{i+1} of the new loop. Otherwise, one non-zero leaf node of root a_i in the set of binary trees representing loop 2 is located as node a_{i+1}.

(ii) node a_i is a common complete node of loop 1 and loop 2.

In this case, new node a_{i+1} to be located and node a_{i-1} already found should belong to different loops and should be on the same stream. Such a node a_{i+1} can be identified using binary trees and the node adjacency matrix of the HEN.

After the new node a_{i+1} is found for the new loop, then leaf node a_{i-1} of root a_i and leaf node a_i of root a_{i-1} in two sets of binary trees representing two loops should be removed before continuing to find other new nodes.

If node a_{i+1} is the same as node a_1, and leaf nodes of roots a_{i+1} in the two sets of binary trees are zero or node a_i, then a new loop $(a_1, a_2, \cdots, a_i)$ is obtained.

5.3.3. Procedure for loop additions

Based on the loop representation by binary trees, the following procedure is proposed for loop additions.

Procedure C (loop additions)

Step 1. Establish two sets of binary trees for two loops, and find common complete nodes existing in two loops. Remove common leaf nodes of identical roots in the two sets of binary trees.

Step 2. Choose an uncommon complete node as the starting node (a_1) of the new loop to be produced, and one non-zero leaf node of root a_1 is selected as the second node (a_2) of the new loop. Let $k = 2$.

Step 3. If node a_k of the new loop is a common complete node, then find the next node (a_{k+1}) of the new loop by the addition of common complete nodes. Otherwise, find the next node (a_{k+1}) of the new loop by the addition of common edges. Remove leaf node a_k of root a_{k-1} and leaf node a_{k-1} of root a_k in two sets of binary trees.

Step 4. If $a_{k+1} = a_1$ and leaf nodes of roots a_{k+1} in the two sets of binary trees are either zeros or node a_k, then a new loop $(a_1, a_2, \cdots, a_k)$ is obtained and go to Step 5. Otherwise, let $k = k + 1$ and go to Step 3.

Step 5. Remove leaf node a_{k+1} of root a_k and leaf node a_k of root a_{k+1} in two sets of binary trees. After that, if there still exist non-zero leaf nodes in the two sets of binary trees, then go to Step 2 to find another new loop. Otherwise, no additional new loops can be found.

It should be noted that additions of any two loops may or may not produce one or more new loops. To save computation time, it is suggested to check whether or not additions of two loops have results before conducting Procedure C. In order to do so, we only need to check whether there exist common edges and/or common complete units in the two loops. If there exist zero leaf nodes in the two sets of binary trees representing two loops after Step 2 of Procedure C is done, then these two loops have edges in common. The following equation can be used to check if two loops have one or more common complete nodes.

$$A_{a_{i-1},a_i} + A_{a_i,a_{i-1}} + A_{a_i,a_{i+1}} + A_{a_{i+1},a_i} = 0$$

where,

$A_{j,k}$ is the value of entry (j, k) of the node adjacency matrix.

5.4. Loop Generation from Independent Loops

Before we present the proposed procedure, some notations are given below:

S is the set of loops already found in the network. R is the set of arrays, each of which records indices of loops whose additions produce a corresponding loop in S. R^* is the set of arrays, each of which records indices of loops whose additions have no results.

Considering the fact that additions of loops cannot always produce one or more new loops, and additions of a loop with a set of loops whose additions have no results could generate one or more new loops, the following procedure is proposed to locate all other loops given a maximum set of independent loops $B_1, B_2, \cdots B_n$.

Procedure D (finding all loops from a maximum set of independent loops)

Step 1. Let $i = 1$, $N = n$, and $N^* = 0$, where i is the index of independent loop B_i, N is the number of loops in S or the number of arrays in R, and N^* is the number of arrays in R^*. Put indices of the maximum set of independent loops $B_1, B_2, \cdots, B_n$ in N arrays of R.

Step 2. Let $j = i+1$, $k = 0$ and $k^* = 0$.

Step 3. Check whether or not additions of loop i and loop j in R have been performed before. If yes, go to Step 5; otherwise, go to Step 4.

Step 4. Check whether or not additions of loop i and loop j in R can produce a new loop. If not, then add array (i, j) to the end of the series of arrays in R^* and let $k^* = k^* + 1$. If yes, then add array (i, j) to the end of the series of arrays in R and use Procedure C to get one or more new loop. Add the new loop(s) to the end of the loop list in S and let $k = k + l$ (where, l is the number of new loops obtained by additions of loop i and loop j).

Step 5. Let $j = j+1$. If $j \leq N$, go to Step 3; otherwise, go to Step 6.

Step 6. If $N^* = 0$, go to Step 10. Otherwise, let $j^* = 1$.

Step 7. Check whether or not loop additions of loop i and every loop in array j^* of R^* have been performed before. If yes, go to Step 9; otherwise, go to Step 8.

Step 8. Check whether or not loop additions of loop i and loops involved in array j^* of R^* can produce one or more new loops. If not, then add array $(i, array\ j^*)$ to the end of the series of arrays in R^* and let $k^* = k^* + 1$. If yes, then use Procedure C to get one or more new loops, and add the new loop(s) to the end of the loop list in S and let $k = k + l$ (where, l is the number of new loops obtained by additions of loop i and loops involved in array j^* of R^*).

Step 9. Let $j^* = j^* + 1$. If $j^* \leq N^*$, go to Step 7; otherwise, let $N = N + k$ and $N^* = N^* + k^*$, and go to Step 10.

Step 10. Let $i = i + 1$. If $i \leq n - 1$ go to Step 2. Otherwise, stop. S consists of all loops of the HEN.

By induction, it can be proven that the above algorithm can produce all loops by completing additions of the first $(n-1)$ independent loops with loops in R and R^*.

In the above procedure, it is needed to check whether or not additions of loops have been accomplished before. In order to do so, the following property of loops should be used:

$$\sigma + \sigma = 0$$

where, σ is a loop.

A FORTRAN program to accomplish Procedures C and D was developed and is very helpful to carry out loop additions. For the network given in Figure 5-1, a total of 50 loops (including 6 independent loops) can be found. To get the dependent loops from the set of independent loops, 44 effective loop addition operations are performed.

5.5. Identification of Downstream Paths

To describe the method for finding downstream paths, we assume that every stream experiences a disturbance at its inlet and has a controlled variable at its outlet. A path is defined as the unbroken connection between any two points in the grid representation (Linnhoff and Kotjabasakis, 1986). For example, for the network shown in Figure 5-1, there exist two paths between the disturbance of stream H3 and the controlled variable of stream C1 as shown in Figure 5-5. Propagation of the disturbance of stream H3 along path 1 can affect the controlled variable of stream C1. But, propagation of the disturbance along path 2 does not have any influence on the controlled variable of stream C1 because part of path 2 is against the flow of stream C2. It can be defined that a downstream path is the directed path which is always in the same direction of streams through which it passes. By the definition, it can be concluded that a disturbance can only affect a controlled variable along with the downstream paths between the disturbance and the controlled variable (Linnhoff and Kotjabasakis, 1986).

To find downstream paths, the node adjacency matrix is not enough. We also need to know the matching among hot and cold streams as well as the locations of splitting and mixing points in the network. Thus, a table, called stream table, is used. In the stream table, the heading row stands for hot streams, heating stream and split stream, while the heading column stands for cold streams, cooling stream and split stream (if different types of utilities are used, then several heating or cooling streams will appear as rows or columns in the stream table). The (i, j)th block is filled with the indices of the units or the indices of splitting and mixing points between streams i and j. Some blocks can have more than one index because there maybe exist more than one unit or splitting and mixing points between the two streams. For example, the stream table for the network shown in Figure 5-1 is given in Table 5-1.

We now use both the stream table and the node adjacency matrix to locate downstream paths between a disturbance and a controlled variable. To do so, the (i, j) th and (j, i) th entries of the node adjacency matrix for two heaters (coolers) i and j should become blanks now.

To find downstream paths between two streams, it only needs to find downstream paths among units on these two streams. A downstream path is exactly the same as a directed path in graph theory. Therefore, the identification of a downstream path from unit i to unit j can be easily implemented with the use of the node adjacency matrix. It only needs to find a directed path from unit i to unit j. This can be achieved by using unit i as the root and finding its directed tree. A directed tree can be found by modifying Procedure A. If unit j is included in the tree, then there is a downstream path between

unit i and unit j, and also the path is determined by the sequence of nodes from unit i to unit j along the path in the directed tree.

From the above analyses, the following steps are presented to determine the existence of downstream paths from the disturbance of stream S1 to the controller variable of stream S2 in a network.

Procedure E (identification of downstream paths):

Step 1. From the stream table, find the units located on streams S1 and S2. Let I and J be the numbers of the units on S1 and S2, respectively. Denote n_i $(i = 1, 2, \cdots, I)$ for units on S1, and m_j $(j = 1, 2, \cdots, J)$ for units on S2.

Step 2. For $i = 1$, define the set S as in Procedure A. At first, let $S = \{n_i\}$.

Step 3. For each node q in S, find node p if it exists, that can be reached directly from node q along the flow direction of a stream, i.e., the entry a_{pq} in the node adjacency matrix is not blank, then add p to the set S. This step stops if S contains all nodes of the network, or $a_{pq} = 0$ for every p.

Step 4. If unit m_j is included in the tree, then there exists a downstream path from S1 to S2. Otherwise, let $i = i + 1$ and go to Step 2 until $i > I$.

Obviously, if let S2 represent all other streams except stream S1 in the network, then we can find all downstream paths from S1 to other streams using the above procedure.

Procedure E is obtained by modifying Procedure A. It is also simple and can be easily implemented. For the network shown in Figure 5-1, results of identifying downstream paths are given in Table 5-2.

5.6. Conclusions

Identifications of all heat load loops are divided into three steps. Firstly, a maximum tree is constructed. Secondly, a maximum set of independent loops is found. Finally, dependent loops are obtained based on the addition operations of independent loops. Detailed procedures are proposed to implement the above steps. All procedures are well-defined and easy to accomplish.

Downstream paths are identified by finding directed trees. The proposed procedure not only can decide whether there exist downstream paths between any two different streams, but also generate sequences of nodes along downstream paths. This is important for breaking downstream paths to prevent the controlled variables from influences of disturbances.

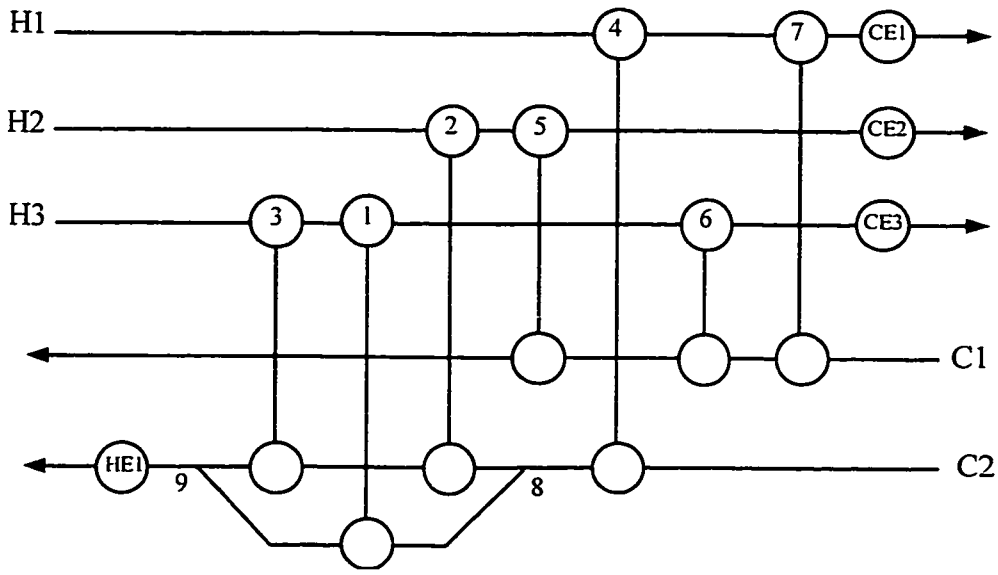


Figure 5-1. An example network (Pethe *et al.*, 1989).

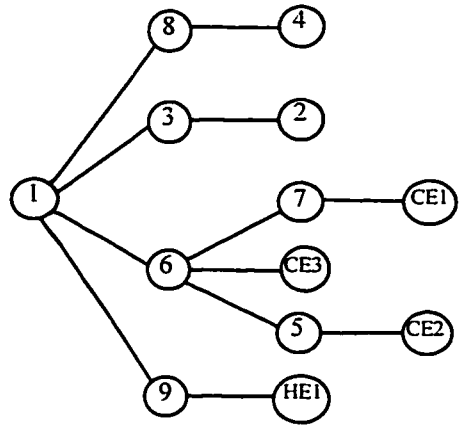


Figure 5-2. A maximum tree for the network in Figure 5-1.

loop (9, 1, 3)

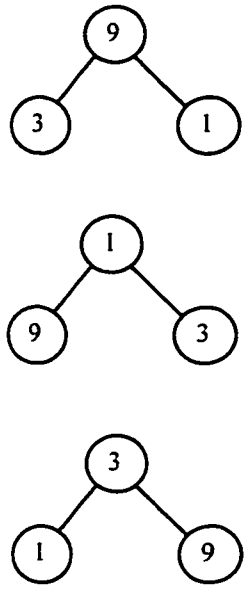


Figure 5-3. Loop representation by a set of binary trees

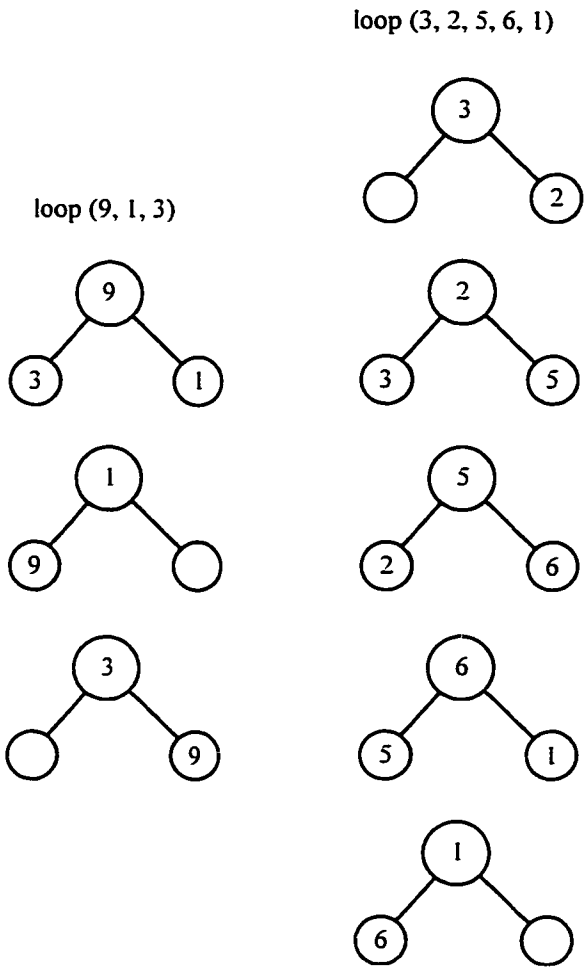


Figure 5-4. Two sets of binary trees after removing common leaf nodes of identical roots

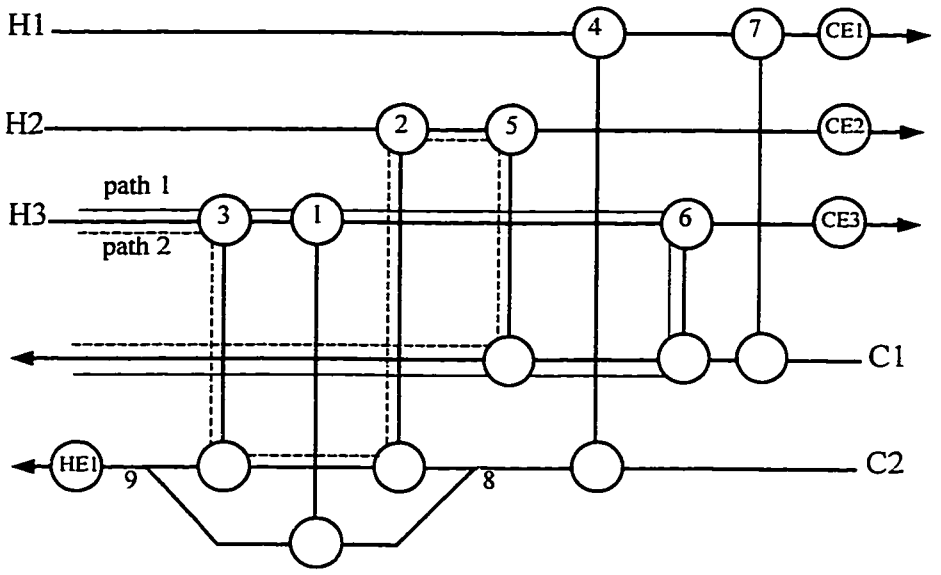


Figure 5-5. Downstream Paths.

Table 5-1. Stream table of the network in Figure 5-1.

	H1	H2	H3	Heating	Split
C1	7	5	6		
C2	4	2	1, 3	HE1	8, 9
Cooling	CE1	CE2	CE3		
Split					

Table 5-2. Downstream paths of the network in Figure 5-1.

From \ To	H1	H2	H3	C1	C2
H1		(4,2), (7,5) (4,3,6,5) (4,1,6,5)	(4,3) (4,1) (7,6)	(7), (4,2,5) (4,3,6) (4,1,6)	(4)
H2			(2,3)	(5) (2,3,6)	(2)
H3		(6,5)		(6)	(3) (1)
C1	(7)	(5)	(6)		
C2	(4)	(2), (3,6,5) (1,6,5) (4,7,5)	(1) (3) (4,7,6)	(4,7), (2,5) (1,6) (3,6)	

5.7. Literature Cited

Calandranis, J., and G. Stephanopoulos, "Structural Operability Analysis of Heat Exchanger Networks," *Chem. Eng. Res. Des.*, **64**, 347 (1986).

Gibbs, N.E., "A Cycle Generation Algorithm for Finite Undirected Linear Graphs," *J. ACM*, **16**, 564 (1969).

Huang, Y.L., and L.T. Fan, "Distributed Strategy for Integration of Process Design and Control — A Knowledge Engineering Approach to the Incorporation of the Controllability Into Exchanger Network Synthesis," *Comput. Chem. Eng.*, **16**, 497 (1992).

Linnhoff, B., and E. Kotjabasakis, "Downstream Paths for Operable Process Design," *Chem. Eng. Prog.*, 23 (May, 1986).

Pethe, S., R. Singh, and F.C. Knopf, "A Simple Technique for Locating Loops in Heat Exchanger Networks," *Comput. Chem. Eng.*, **13**, 859 (1989).

Welch, J.T., JR., "A Mechanical Analysis of the Cyclic Structure of Undirected Linear Graphs," *J. ACM*, **13**, 205 (1966).

CHAPTER 6. SIMPLIFYING HEAT EXCHANGER NETWORKS TO ACHIEVE THE MINIMUM ENERGY PENALTY[1]

6.1. Introduction

As one kind of major methods for synthesis of heat exchanger networks (HENs), evolutionary design methods (e.g., pinch design method (Linnhoff and Hindmarsh, 1983)), based on heuristics and thermodynamics, have been widely used in industrial practices (Gundersen and Naess, 1988; Linnhoff, 1993). The HENs generated by evolutionary methods often feature the maximum energy recovery for chosen approach temperature Δt_{min}s but involve more units than the targeted minimum number of units. It is often required to simplify the generated networks by removing extra units. Such simplifications are accomplished to achieve certain objectives such as the minimum energy penalties or the optimal trade-off between energy and capital costs. This work only considers simplifying the networks to achieve the minimum energy penalty. It is very difficult to determine the best units for removal because the energy penalty incurred by deleting a unit depends on both heat loads and interactions of units in the networks. An often-adopted heuristic rule is removing the unit with the smallest heat load (Linnhoff *et al.*, 1982). This rule assumed that the energy penalty incurred by deleting a unit is proportional to the quantity of the heat load of the unit, and neglected the effect of unit interactions on the energy relaxation. Obviously, it solves the problem in an over-simplified way and thus is not reliable.

---

[1] A version of this chapter has been submitted for publication:
J.Y. Zhu, M. Rao and K.T. Chuang, "A New Method for Simplifying Heat Exchanger Networks," *Ind. Eng. Chem. Res.*, (1998).

Several methods have been proposed to deal with the complexities of HENs (Trivedi *et al.*, 1990; Zhu *et al.*, 1993). Trivedi *et al.* (1990) classified units in a network into three types and concluded that units which are in the set where unit loads can only be decreased are possible candidates for deletion. This method cannot always find the best units because it takes into account the restriction of their Type A units but ignores the effects of heat loads and capabilities of absorbing transferred energy of other units. Zhu *et al.* (1993) applied the Kirchhoff's law for electrical networks to simplify HENs. The Kirchhoff's law is used to calculate the energy penalties incurred by deleting units from HENs. Their method determines which units should be removed by comparing the amounts of energy penalties of different units. For complex networks, it could not be easy to do such calculations.

In this work, a new method is proposed for determining the best units to simplify HENs with respect to the minimum energy penalty. The method takes into account effects of both the loads and interactions of units on the energy relaxation by identifying constrained units and examining the redistribution of Δt_{min}s caused by deleting units from networks. The proposed method does not need to calculate accurate energy penalties of all units but only needs to estimate their lower bounds. A systematic procedure is presented to find the best units for breaking given loops.

6.2. Loop-breaking and Energy Penalty

The number of units involved in a HEN has been deduced from graph theory (Boland and Linnhoff, 1979) as follows:

$$U = N + L - s$$

where, U is the number of units, N is the total number of streams (including heating and cooling utility streams), L is the number of independent heat load loops and s is the number of separated components ($s = 1$ for a connected network). When no loop (i.e., $L = 0$) is involved, the network has the minimum number of units. Thus, to decrease the number of units, loops should be broken.

If one unit is deleted from a loop, then its load should be redistributed to other units in the network. Such load redistribution often causes some Δts to violate the Δt_{min} restriction. The restoration of such Δts to Δt_{min} can be achieved by increasing the utility consumption and thus an energy penalty is incurred. The removal of different units could result in very different energy penalties. To minimize the energy penalty, units which are the best for deletion with respect to the energy consumption should be identified in a systematical way.

6.3. Constrained Units for Loop-breaking

Due to the energy balance and Δt_{min} constrictions, some units in HENs could have some features that are helpful to determine the best units for loop-breaking. Trivedi *et al.* (1990) classified units in a network into the following three types:

Type A — "A unit that has one temperature difference equal to Δt_{min} and the temperature of one of the streams supplied to the unit is fixed. If the other temperature difference is on the side of the exchanger where the other stream enters this unit, then the entry temperature of this stream must be unaffected by loop-breaking. Heat loads of these units cannot be increased and they are often found adjacent to the pinch."

Type B — "Such units are required by the network to enable one of the streams to achieve its target temperature. Clearly they cannot be eliminated but their load may be increased or decreased."

Type C — "During the loop-breaking process, units are removed and their loads transferred to some of the remaining units. Type C units can absorb a fraction of this load and the remainder must be removed along a path. The magnitude of this remainder represents a lower bound for the energy penalty incurred following such loop-breaking."

In this work, we don't treat Type B units of Trivedi *et al.* (1990) as constrained units because if new utility units (heaters and/or coolers) are allowed to be introduced, Type B units then can be deleted from HENs. Like the method by Zhu *et al.* (1993), this work will also consider adding new utility units to restore the violation of thermodynamic constrictions. The reasons are that firstly, the objective of loop-breaking in this work is to achieve the minimum energy penalty. If we can get less energy penalties by introducing new utility units, then we choose introducing new utility units. Secondly, utility units can be regarded as sinks for disturbance (Huang and Fan, 1992). Introduction of new utility units will improve the controllability of HENs with respect to disturbance rejection.

If introducing new utility units during loop-breaking is not allowed, then identification of units which cannot be removed without violation of thermodynamic constraints is helpful to decrease the size of search space of units for removal. But, besides restrictions of some target temperatures as explained by Trivedi *et al.* (1990), the constriction of energy balance will also make some units not be removable. For example,

in the network shown in Figure 6-3(b), if additional utility units are not introduced, unit 5 cannot be removed because of the constriction of the energy balance of stream C2 (i.e., the load of unit 5 is greater than the load of unit 4 plus the load of unit 10).

Type C units of Trivedi *et al.* (1990) are not considered constrained units either in this work because the amount of energy such units can absorb depends on interactions of units in HENs and normally cannot be evaluated simply by considering these units independent of other units.

When a unit is removed from a network, part of its heat load will be transferred to other units in the network and the remaining part of its heat load will be transferred to heaters and coolers as the energy penalty incurred by its removal. Obviously, the more the heat load of the removed unit is absorbed by other units, the less the energy penalty is incurred.

In HENs, some units cannot absorb any energy transferred from a removed unit, and also cannot allow the energy of the removed unit to pass through themselves to other units. Therefore, the identification of these units is useful to evaluate energy penalties incurred by removing some units.

Typical cases of such constrained units are that a unit whose supplied (or target) temperatures of the hot and cold streams are fixed and the Δt at one side of the unit is equal to Δt_{min}. The schematic illustrations of the cases are shown in Figure 6-1.

In the grid representations of HENs, a unit is represented by two parts, one located on a hot stream (defined as hot part), another one located on a cold stream (cold part).

Another important feature of the constrained units defined in this work is that its hot part does not allow energy to be transferred from low to high temperatures, and its cold part does not allow energy to be transferred from high to low temperatures. Due to this feature of the constrained units, the removal of some units will result in the same amounts of energy penalties as their loads. Obviously, the identification of the constrained units will be helpful to evaluate energy penalties.

Constrained units defined in this work are similar to Type A units presented by Trivedi *et al.* (1990). But, the criterion of detecting such constrained units is more rigorous than that by Trivedi *et al.* (1990), and also the constrained units play different roles in finding the best units for loop-breaking.

6.4. Estimation of Lower Bound of Energy Penalty

Although there are no simple ways to quickly evaluate the exact energy penalty incurred by deleting a unit except that its removal will result in the same amount of energy penalty as its load, we could easily estimate the lower bound of the energy penalty by examining temperature redistribution. By doing such estimations, we could exclude some units from removal and decrease the number of units which need the accurate calculations of energy penalties.

6.4.1. Redistribution of Δt_{min}

When a unit is deleted from the network, the distribution of Δt_{min}s will be changed. Three kinds of such changes could happen: (a) some Δts could be tighten to Δt_{min}s, (b)

some Δt_{min}s still exist but at different temperatures, and (c) some Δt_{min}s are relaxed to Δts.

For any unit in HENs, by only comparing the heat capacities of the hot stream (denoted as *CPH*) and the cold stream (denoted as *CPC*) through the unit, it is easy to find which side of the unit has smaller Δt. If $CPH = CPC$, Δts at both sides of the unit are the same. If $CPH < CPC$, then Δt at the left-hand side of the unit is larger than that at the right-hand side of the unit. If $CPH > CPC$, then Δt at the left-hand side of the unit is smaller than that at the right-hand side of the unit. Thus, we can easily determine the redistribution of Δt_{min}s caused by deleting a unit.

6.4.2. Estimation of lower bound of energy penalty

When a unit is removed from a network, sometimes by simply examining the redistribution of Δt_{min}s, we can easily calculate the lower bound of the energy penalty. For example, considering the network shown in Figure 6-2, when unit 2 is removed, we can evaluate the lower bound of its energy penalty as follows.

From Figure 6-2, we know that unit 3 has a smaller Δt at its left-hand side than at its right-hand side, and unit 4 has a smaller Δt at its right-hand side than at its left-hand side. When unit 2 is removed, we can easily obtain that (a) unit 3 has a Δt_{min} at 180/160 °C at its left-hand side, and (b) unit 4 also has a Δt_{min} at 180/160 °C at its right-hand side in the stream grid representation. From (a), we can conclude that the temperature of the hot outlet of unit 3 cannot be less than its original temperature (122 °C), that is, the load of cooler C2 cannot be decreased after unit 2 is removed. From (b), we know that the load

of cooler C1 will increase 1200 kW (i.e., 30*(180-140)). Therefore, the energy penalty incurred by removing unit 2 should not be less than 1200 kW.

Alternatively, we can also calculate the lower bound of the above energy penalty in the following way.

When unit 2 is removed, the temperature of the hot outlet of unit 1 cannot be less than 220 °C because the temperature of the cold outlet of unit 4 is 200 °C. The maximum energy which can be transferred to unit 1 is 810 kW (i.e., 30*(247-220)). Therefore, the energy penalty incurred by removing unit 2 should not be less than 990 kW (i.e., (1800-810)).

The above example has shown that by examining temperature distributions, it could be possible to easily evaluate the lower bound of the energy penalty incurred by removing a unit. As we will see in the following sections, estimating lower bounds of energy penalties plays an important role in preliminarily excluding some units from removal. Thus, unnecessary calculations of the accurate energy penalties can be avoided.

6.5. Proposed Method for Loop-breaking

So far, we have discussed the effects of both loads and locations of units on the energy relaxation. Based on the above analyses, a systematic method for loop-breaking is proposed to determine units which could incur lower energy penalties.

6.5.1. Sequence of units considered for removal from the same loop

Given a loop to break, the sequence we adopt to consider units in the given loop for removal is in terms of heat loads of units. A unit with a larger load is considered to be

removed before those with smaller loads. The reason is that when we consider a unit for removal, we first estimate its lower bound of the energy penalty. If its lower bound of the energy penalty is larger than the load of at least one unit in the same loop, then as a preliminary decision, we do not consider deleting the unit with the larger load. By using this sequence, we could quickly exclude some units as candidates for removal in the preliminary screening.

It could be possible that the energy penalty resulting from deleting a unit is larger than the load of the unit. Therefore, as stated above, by only comparing lower bounds of energy penalties and loads of units, we cannot confirm which unit is the best one for removal. However, by doing so, we can guess which unit is considered for removal. Thus, we can decrease the number of units which need accurate calculations of their energy penalties. The following proposed procedure for loop-breaking can be used to find the best units for removal based on the above idea.

6.5.2. Sequence for breaking multi-loops

It is obvious that different sequences of loops to break will result in different energy penalties. It is also impractical to take all loops into account to determine the best units with respect to energy penalties, because the number of all loops could be very big even for a simple HEN. To avoid examining all loops, heuristics are applied to decide sequences of loops to break (Trivedi *et al.*, 1990; Zhu *et al.*, 1993). In this work, the heuristic suggested by Zhu *et al.* (1993) is adopted, i.e., "first consider process-process loops with the smallest number of units (the lower level loops) and those with the smallest heat loads."

6.5.3. Procedures for loop-breaking

Based on the above discussions, the following steps are proposed to simplify networks by loop-breaking:

Step 1. For a given network, we find all independent loops by using one of existing methods (e.g. Zhu *et al.*, 1996).

Step 2. We first break process-process loops. When no process-process loops are left or breaking process-process loops could cause higher energy penalty than some process-utility loops, we break process-utility loops.

Step 3. For a loop selected for breaking, we identify constrained units which cannot absorb any transferred energy.

Step 4. We identify units whose removal will result in the same amounts of energy penalties as their loads if the given loop does include such units.

Step 5. In the remaining units, we first estimate the lower bound of the energy penalty of the unit with the largest load.

(a) We compare the lower bound with the energy penalties determined in Step 4. If the lower bound is larger than one of the energy penalties, then the unit should not be deleted and go to Step 6. Otherwise, go to (b).

(b) We compare the lower bound with loads of remaining units. If the lower bound is greater than the load of at least one remaining unit, as a preliminary decision, we do not consider the unit with the larger load as a

candidate for removal. Otherwise, we need to calculate its accurate energy penalty at Step 7.

Step 6. Continue to another unit with the next largest load and repeat Step 5 until all units have been checked.

Step 7. For the units whose accurate energy penalties are required for selecting the best unit for removal as discussed in Step 5, we calculate their accurate energy penalties. Among these units and the units identified in Step 4 whose energy penalties are equal to their loads, we find the unit with the lowest energy penalty.

Step 8. Because the energy penalty incurred by removing a unit could be larger than its load, we need to check if the lowest energy penalty determined in Step 7 is larger than one or more lower bounds of energy penalties evaluated in Step 5. If not, the unit with the lowest energy penalty is the best unit to be deleted from the given loop. If yes, we need to calculate the accurate energy penalties of the units whose estimated lower bounds of energy penalties are less than the lowest energy penalty determined in Step 7. Then, the unit with the lowest energy penalty is selected as the best one to break the given loop.

Although we may still need to calculate the accurate energy penalties of some units, the number of such units could be much less than the number of all units in the given loop.

6.6. Illustrations

To explain the proposed method and demonstrate its reliability and efficiency in simplifying networks, an example network which has been used by Trivedi *et al.* (1990) and Zhu *et al.* (1993) is also simplified by applying the proposed method for comparison. The pinch design result of the example is shown in Figure 6-3(a).

First, let's consider breaking loops (4, 11). Units 6 and 7 cannot absorb any transferred energy. Because the load of unit 4 is larger than that of unit 11, we first estimate the lower bound of the energy penalty of units 4. Since units 6 and 7 cannot absorb any transferred energy, so is unit 10. Unit 9 cannot increase its heat load, otherwise the Δt at the left-hand side of unit 9 will be less than Δt_{min} because the heat capacity of stream H2 is greater than that of stream C2. The energy which can be absorbed by unit 3 or unit 11 will be constrained by the Δt_{min} of the left-hand side of unit 5 or the Δt_{min} of the right-hand side of unit 9. Thus, it can be estimated that deleting unit 4 will result in the same amount of the energy penalty as its load. As a preliminary decision, unit 11 is selected for removal. The accurate energy balance calculations show that the energy penalty incurred by deleting unit 11 is close to zero. Thus, to break loop (4, 11), unit 11 should be removed.

The next loop to be broken is loop (3, 9). The load of unit 3 is greater than that of unit 9, so we first estimate the lower bound of the energy penalty resulted in by deleting unit 3. If unit 3 is removed, an additional heater should be introduced to the hot end of stream C2, and the load of the new heater will not be less than 492 kW because the cold outlet temperature of unit 4 cannot be less than 150 °C. Thus, even if we assume the load

of unit 1 could be decreased to zero, the energy penalty resulted in by removing unit 3 cannot be less than 261 kW (i.e., 492-231) which is still larger than the load of unit 9. We choose unit 9 for removal, and its energy penalty is equal to 13 kW. The simplified network generated by deleting units 11 and 9 is shown in Figure 6-3(b).

To further simplify the network, let's consider breaking loop (2, 5, 8, 7). Since unit 7 has the largest load, we first estimate the lower bound of its energy penalty. If unit 7 is removed, the maximum transferred energy that unit 2 can absorb is equal to 34 kW (i.e., (231-227)*8.44) and unit 6 cannot absorb any energy. Thus, the lower bound of the energy penalty incurred by removing unit 7 is 876 kW (i.e., (910-34)). Secondly, we estimate the lower bounds of energy penalties of units 5 and 8. Because the maximum transferred energy unit 5 can absorb is equal to 25 kW (i.e., (173-170)*8.44) and unit 7 cannot absorb any energy, the lower bound of the energy penalty resulted in by removing unit 8 is 268 kW (i.e., (293-25)). If unit 5 is removed, the hot outlet temperature of unit 7 will not be less than 170 °C, which means the load of unit 7 will be decreased to 673 kW (i.e., (227-170)*11.816). As mentioned before, the maximum additional energy that can be absorbed by unit 2 is 34 kW and unit 6 cannot absorb any energy. Thus, the lower bound of the energy penalty incurred by deleting unit 5 can be estimated as 447 kW (i.e., (910-673-34+244)). The energy penalties by removing units 5, 7 or 8 are all larger than the load of unit 2. Therefore, unit 2 is selected for removal. The energy penalty incurred by deleting unit 2 is equal to 144 kW. The obtained network is given in Figure 6-3(c).

Now, loop (3, 10, 6, 7, 8, 5) is considered for breaking. Because units 6 and 7 cannot absorb any energy, their removal will incur the same amounts of energy penalties as their

loads. If unit 3 is removed, a new heater should be introduced to stream C2 because of the restriction of the source temperature of stream H1. The load of the new heater cannot be less than 489 kW (i.e., (217-150)*7.296). Thus, the energy penalty incurred by deleting unit 3 cannot be less than 489 kW. If unit 5 is deleted, the hot outlet temperature of unit 7 cannot be less than 170 °C, and the load of unit 7 will be less than 674 kW (i.e., (227-170)*11.816). The increase of the load of unit 1, i.e., the energy penalty of unit 5, will not be less than 236 kW (i.e., (910-674)). If unit 8 is removed, the load of unit 3 cannot be less than 489 kW (i.e., (217-150)*7.296), and thus the maximum energy unit 5 can absorb is equal to 152 kW (i.e., (641-489)), which is less than the load of unit 8. Therefore, if unit 8 is removed, a new heater is required for stream C1 and the load of the new heater will not be less than 141 kW (i.e., (293-152)). Because units 6 and 7 cannot increase their loads, the lower bound of the energy penalty incurred by removing unit 8 is 141 kW. From the above analyses, it can be seen that energy penalties incurred by deleting units 3, 5 or 8 are all larger than the load of unit 10. Thus, unit 10 is selected for deletion. A new cooler needs to be introduced to stream H4 and the energy penalty resulted in by deleting unit 10 is zero. The simplified network so far obtained is shown in Figure 6-3(d).

After all process-process loops are broken, process-utility loops are considered. Now, let's break loop (4, 12, 13, 8, 5, 3). Unit 12 must be maintained in order to achieve the target temperature of stream H1. Unit 13 cannot be deleted; otherwise, the Δt at the left-hand side of unit 8 will violate the restriction of Δt_{min} because the heat capacity of the hot stream of unit 8 is larger than the heat capacity of its cold stream. As analyzed

before, the lower bounds of energy penalties incurred by removing units 3, 5 or 8 are 489 kW, 236 kW or 141 kW, respectively. The energy penalty resulted in by deleting unit 4 is equal to 90 kW which is less than the lower bounds of energy penalties of units 3, 5 or 8. Thus, unit 4 is deleted.

Now, loop (3, 5, 8, 7, 1, 15) needs to be broken. Because unit 7 cannot absorb any transferred energy and its hot part does not allow energy to be transferred from low to high temperatures, removing unit 8 will result in the same amount of the energy penalty as its load. The lower bounds of energy penalties incurred by deleting units 3 or 7 are 489 kW or 910 kW, which are larger than the load of unit 5. Unit 5 is tentatively selected for removal. The energy penalty incurred by deleting unit 5 is equal to 147 kW, and this energy penalty can be further decreased by 22 kW if we replace unit 14 with the original unit 10. Obviously, unit 5 should be deleted. The simplified network is shown in Figure 6-3(e).

In the network shown in Figure 6-3(e), the only loop that needs to be broken is loop (6, 10, 8, 7). Because units 6 and 7 cannot absorb any transferred energy, their removal will result in the same amounts of energy penalties as their loads. Unit 10 cannot be deleted without violating the constriction of Δt_{min}. Thus, unit 8 should be deleted. After unit 8 is deleted, we can restore unit 16 to the original unit 5. Therefore, the simplified network with the minimum number of units is shown in Figure 6-3(f). Compared with the network shown in Figure 6-3(a), the energy penalty of the simplified network with the minimum number of units is equal to 526 kW.

If a small violation of Δt_{min} restriction for unit 10 is allowed as did by Zhu *et al.* (1993), then unit 10 is the best one to break loop (6, 10, 8, 7) in Figure 6-3(e). The final simplified network is shown in Figure 6-3(g) and the total energy penalty is 376 kW.

6.7. Conclusions

Although Trivedi *et al.* (1990) realized that constrained units are helpful to optimal loop-breaking, their method based on their analyses about the effects of constrained units on energy relaxation could produce sub-optimal designs. In this work, only the constrained units which cannot absorb any transferred energy are employed in loop-breaking, while other two constrained units as classified by Trivedi *et al.* (1990) are not applied. It is revealed that due to the existence of such constrained units, deleting some units will result in the same amounts of energy penalties as their loads.

To ensure that the unit located for breaking a given loop produces the minimum energy penalty and to avoid accurate calculations of energy penalties of units in the loop, estimations of lower bounds of the energy penalties are introduced. For a given loop, the lower bounds of the energy penalties of units in the decreasing order of their loads are estimated by examining the redistribution of Δt_{min}s. As a preliminary screening, we can exclude some units from removal by comparing the lower bounds with the loads of other units. Such estimations provide an efficient and reliable way to find the best unit to break the given loop.

Based on the detection of constrained units and the estimations of lower bounds of energy penalties, a systematic method is developed to find the best units for simplifying HENs. Comparing with the method by Trivedi *et al.* (1990), this method could be more

reliable because it fully considers the effects of loads and interactions of units on the determination of energy penalties. Comparing with the method by Zhu *et al.* (1993), our method could be more efficient, especially for complex networks since our method does not require calculating the energy penalties of all units.

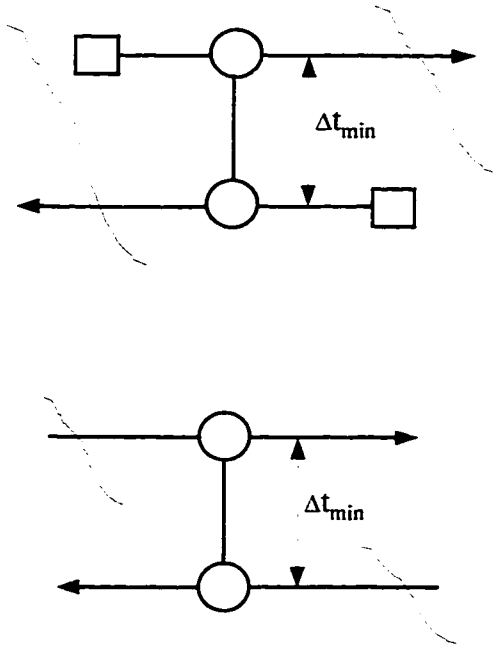


Figure 6-1. Typical cases of units which cannot absorb any energy.

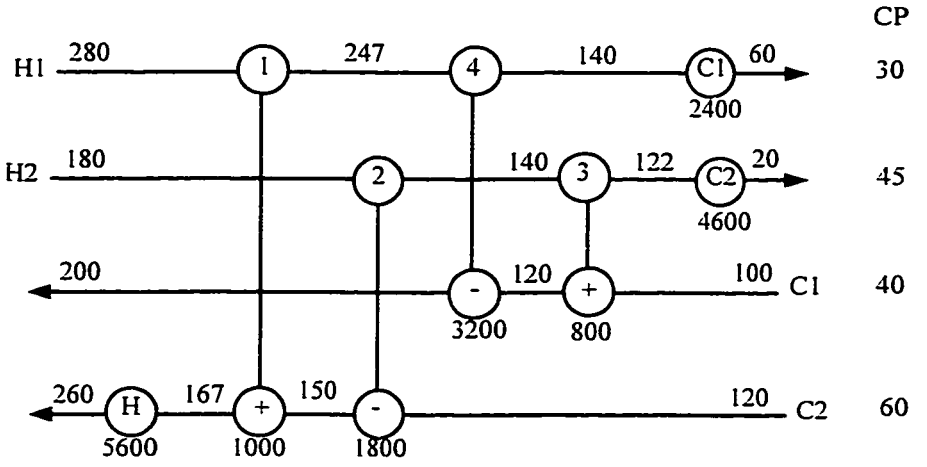


Figure 6-2. An example network for explaining the estimation of lower bounds of energy penalties.

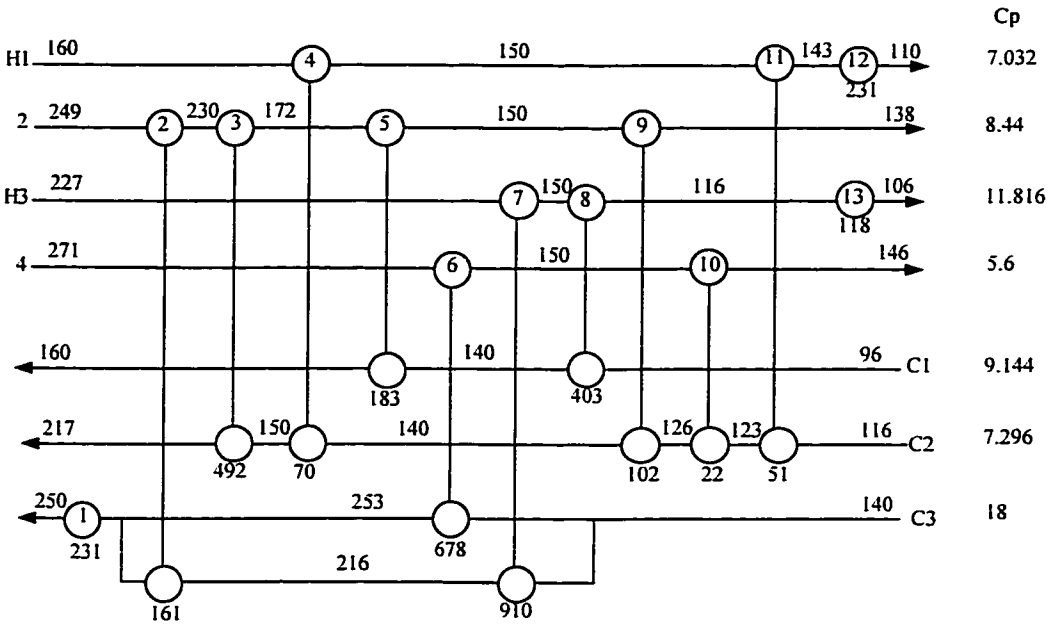


Figure 6-3(a). The initial network (Trivedi *et al.*, 1990).

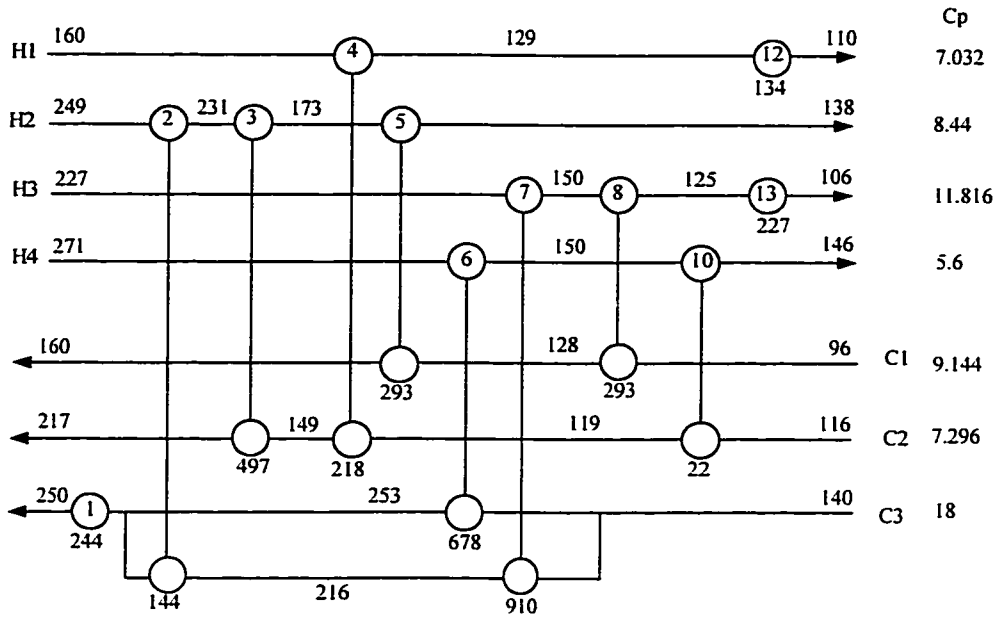


Figure 6-3(b). The simplified network after units 11 and 9 are deleted (energy penalty = 13 kW).

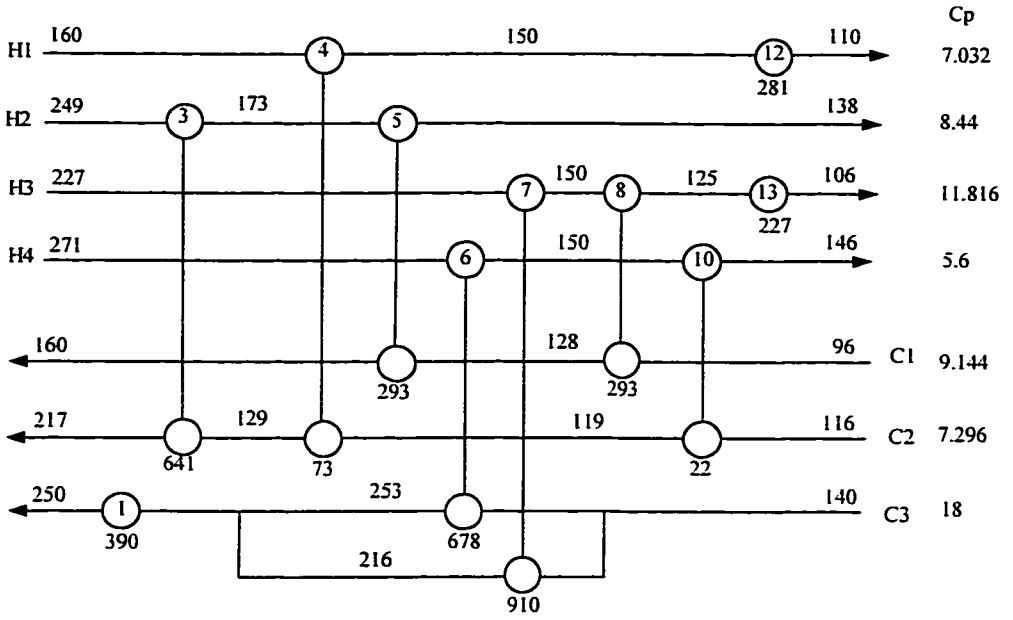


Figure 6-3(c). The resulted network after units 2, 9 and 11 are removed (energy penalty = 159 kW).

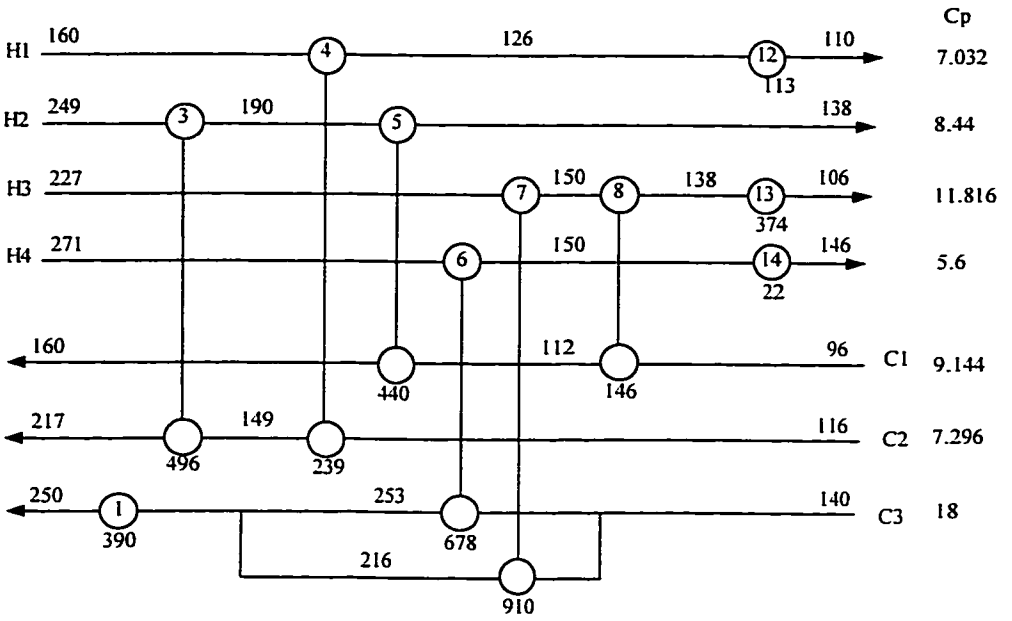


Figure 6-3(d). The simplified network after units 11, 9, 2 and 10 are removed (energy penalty = 159 kW).

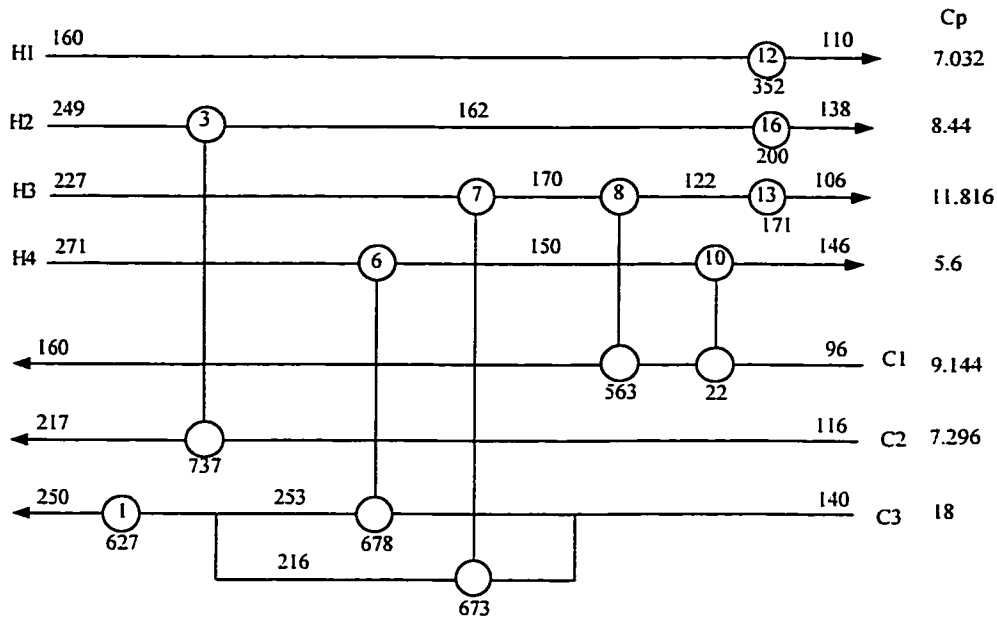


Figure 6-3(e). The simplified network after units 11, 9, 2, 4 and 5 are removed (energy penalty = 396 kW).

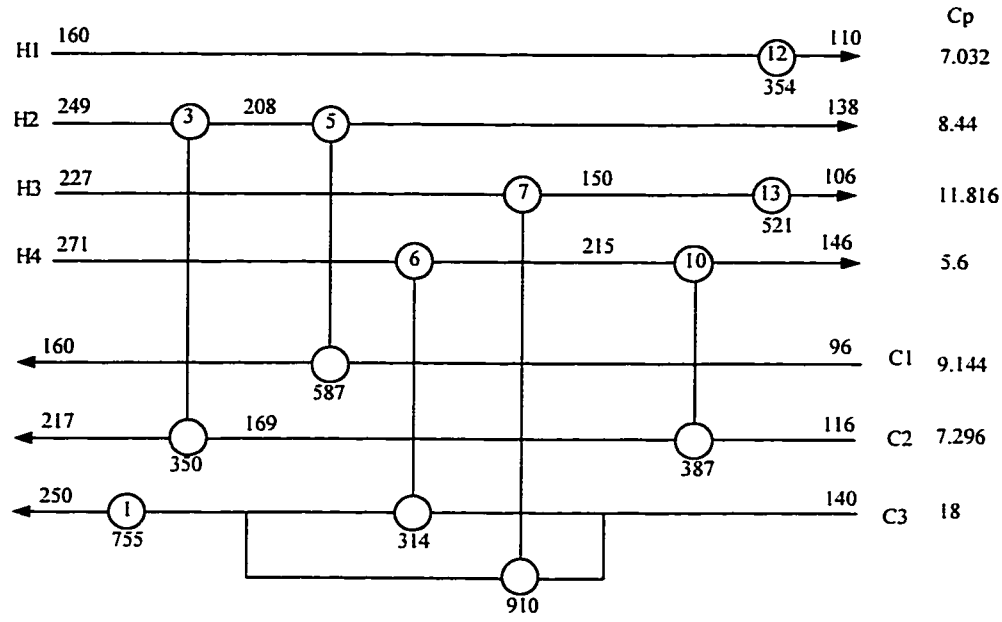


Figure 6-3(f). The final simplified design with the minimum number of units (energy penalty = 526 kW).

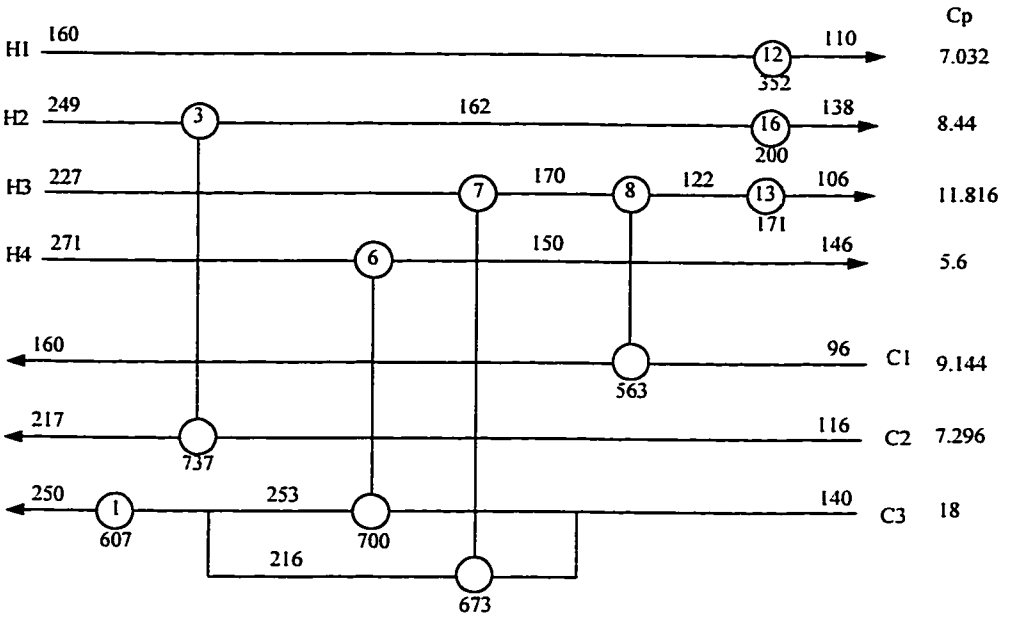


Figure 6-3(g). The final simplified design with the minimum number of units and a small violation of Δt_{min} restriction (energy penalty = 376 kW).

6.8. Literature Cited

Boland, D.; Linnhoff, B. The Preliminary Design of Networks for Heat Exchanger by Systematic Methods. *The Chem. Eng.* **1979**, *April*, 9.

Gundersen, T.; Naess, L. The Synthesis of Cost Optimal Heat Exchanger Networks. *Comput. Chem. Eng.* 1988, *12*, 503.

Huang, Y.L.; Fan, L.T. Distributed Strategy for Integration of Process Design and Control: An Knowledge Engineering Approach to the Incorporation of Controllability into Exchanger Network Synthesis. *Comput. Chem. Eng.* **1992**, *16*, 497.

Linnhoff, B.; Townsend, D.W.; Boland, D.; Hewitt, G.F.; Thomas, B.E.A.; Gray, A.R.; Marsland, R.H. *User's Guide for Process Integration for the Efficient Use of Energy;* IChemE: Rubgy, 1982.

Linnhoff, B.; Hindmarsh, E. The Pinch Design Method for Heat Exchanger Networks. *Chem. Eng. Sci.* **1983**, *38*, 745.

Linnhoff, B. Pinch Analysis — A State-of-the-art Overview. *Chem. Eng. Res. Des.* **1993**, *71*, 503.

Trivedi, K.K.; O'Neill, B.K.; Roach, J.R.; Wood, R.M. Systematic Energy Relaxation in MER Heat Exchanger Networks. *Comput. Chem. Eng.* **1990**, *14*, 601.

Zhu, J.; Han, Z.; Rao, M.; Chuang, K.T.; Identification of Heat Load Loops and Downstream Paths in Heat Exchanger Networks. *Can. J. Chem. Eng.* **1996**, *74*, 876.

Zhu, X.X.; O'Neill, B.K; Roach, J.R.; Wood, R.M. Kirchhoff's Law and Loop-breaking for the Design of Heat Exchanger Networks. *Chem. Eng. Comm.* **1993**, *126*, 141.

CHAPTER 7. CONCLUSIONS AND RECOMMENDATIONS

7.1. Conclusions

This thesis addresses several aspects of control system synthesis, integrated process design and plant-wide control, and heat exchanger network synthesis. Its main contributions can be summarized below in terms of the order of chapters in the thesis.

7.1.1. Excess control objective detection and plant redesign

Control objectives and manipulated variables are represented as rows and columns in the compound structural matrices of control systems, respectively (Johnston *et al.*, 1984). Systematic methods for finding excess rows and adding new columns to the structural matrices are proposed. All excess rows are obtained by finding excess rows from individual maximum dilation submatrices and finding additional excess rows from combinations of different maximum dilation submatrices. By revealing properties of maximum dilation submatrices, various rules are presented to locate all excess rows. In case excess control objectives must be maintained, plant redesign must be carried out to introduce new manipulated variables. Thus, a method is also proposed for determining new manipulated variables which should be introduced by plant redesign.

The main contribution of this work is that it develops a new method to detect all excess control objectives for control system synthesis. The proposed method is simple and also has sound theoretic bases.

7.1.2. Integrated process design and plant-wide control

A framework to incorporate structural state or structurally functional controllability requirements into conceptual process design is proposed. It is based on the hierarchical

design procedure developed by Douglas (1985). The basic idea for the integrated process design and plant-wide control is that each design level solves its control problem at the current level. Before proceeding to the next level, processes of the current level should be all controllable.

At each level, every subsystem is designed separately and only the new control objectives introduced by the current level are considered, which makes the control system synthesis of larger systems simpler. If local control is applied to some subsystems, the controllability test of the composite system at the current level is also required. As a case study, the integrated process design and control of the HDA process is presented to demonstrate the proposed method.

This work presents a systematic method to consider control requirements simultaneously in the design stages of the hierarchical procedure for conceptual design. It can be used to avoid some control problems at early stages of process designs or provide another criterion to screen design alternatives. The method can also be used for control system synthesis of complete plants.

7.1.3. Determination of independent loops of heat exchanger networks

A promising method to find independent loops is constructing a maximum tree and then adding edges that are not in the tree to the tree. To apply the method to locate independent loops of heat exchanger networks (HENs), an alternative representation of HENs is presented. Based on this representation, a procedure for finding independent loops of HENs is proposed. It is proved that a set of loops located by the proposed procedure is a maximum set of independent loops of HENs.

It is shown that split streams should be treated as separated ones in determining independent loops of HENs. By doing so, some confusion in identifying loops caused by ignoring split streams can be avoided, and some loops whose breaking can result in a smaller energy penalty will not be missed. When split streams are considered independently, the existing equation cannot be applied directly to calculate the number of independent loops. Thus, a new equation is derived to determine the number of independent loops of HENs.

This work presents and proves a method to find a maximum set of independent loops of HENs. It also considers effects of split streams on determining independent loops of HENs.

7.1.4. Identification of heat load loops and downstream paths

Based on graph theory, the problem of identifying loops of HENs can be decomposed into three subproblems: (a) determining a maximum tree, (b) identifying a maximum set of independent loops, and (c) generating all dependent loops by loop additions.

With the help of node adjacency matrices of HENs, a maximum tree of a HEN can be easily constructed by adding nodes which have direct connections with the nodes already in the tree. The maximum tree contains all nodes. By identifying edges that are not in the tree and adding the edges to the tree, a maximum set of independent loops can be obtained. To find all loops of a HEN, two types of loop additions are formalized. Binary trees are used to implement the loop additions.

Locating downstream paths (Linnhoff and Kotjabasakis, 1986) is accomplished by constructing directed trees between nodes on streams with disturbances and nodes on streams with controlled variables. Such directed trees can be generated by a procedure with a slight modification of the procedure for determining a maximum tree.

This work provides well-defined procedures for identifying maximum sets of independent loops, finding all loops by loop additions, and determining downstream paths between disturbances and controlled variables.

7.1.5. Simplification of heat exchanger networks

It is realized that the energy penalty incurred by removing a unit from a given heat load loop depends not only on the loads of all units, but also on the structure of the heat exchanger network. To effectively account for these two effects, constraint units and estimations of lower bounds of energy penalties are employed. Different from the method by Trivedi *et al.* (1990), constrained units in this work only include the units which cannot absorb any transferred energy, and are used to find units whose removal will result in the same energy penalties as their loads. The lower bounds of energy penalties of units can be evaluated by examining the redistribution of Δt_{min}s. Estimating lower bounds of energy penalties and determining the best unit by comparing the lower bounds of energy penalties and heat loads of units are the main difference between this and other published work (Trivedi *et al.*, 1990; Zhu *et al.*, 1993).

This work develops a reliable and efficient method to determine the best unit for breaking a given heat load loop with respect to the minimum energy penalty.

7.2. Recommendations for the future work

7.2.1. Integrated process design and plant-wide control based on numerical models

The integrated process design and plant-wide control in this thesis only considers structural models and structural controllability. However, structural controllability does not use the quantitative information of processes, thus the incorporation of structural controllability test into design levels in this thesis may not reveal control potentials and control system design requirements. The design alternatives obtained may still have some control problems in numerical sense. Therefore, when more quantitative information of processes are available, additional control considerations in numerical sense should be incorporated into the design levels, so that not only structural control problems, but also control problems in numerical sense can be avoided at the stages of process designs.

7.2.2. Simplifying heat exchanger networks to achieve the minimum total economical cost

When different units are removed from a HEN, not only the energy consumption but also the capital cost will be different. In this thesis, the best units are determined to achieve the objective of the minimum energy penalty. However, we often need to design a HEN with the optimal trade-off between the capital and energy costs. Thus, systematic methods to determine the best units for simplifying the HEN with respect to the minimum total economical cost need to be developed.

7.2.3. Integrated process design and control of heat exchanger networks

The integrated process design and control of HENs has not been addressed in this thesis yet, although some essential work has been discussed in detail. The objective of this work is to generate HENs that not only feature the optimal total economical cost, but also satisfy certain control requirements (e.g., higher controllability). A systematic method is needed to incorporate the control requirements into certain evolutionary design methods (e.g., pinch design method). To effectively implement the developed method, an intelligence system could be very useful.

7.3. Literature Cited

Douglas, J.M., "A Hierarchical Decision Procedure for Process Synthesis," *AIChE J.*, **31**, 353 (1985).

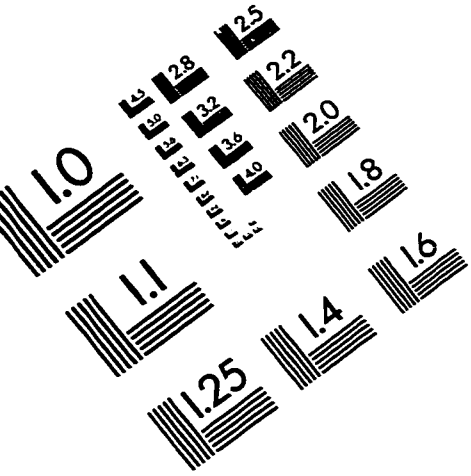
Johnston, R.D., G.W. Barton, and M.L. Brisk, "Control Objective Reduction in Single-Input Single-Output Control Schemes," *Int. J. Control*, **40**, 265 (1984a).

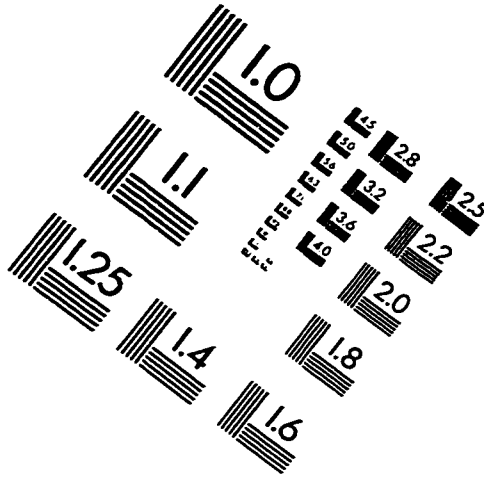
Linnhoff, B., and E. Kotjabasakis, "Downstream Paths for Operable Process Design," *Chem. Eng. Prog.*, 23 (May, 1986).

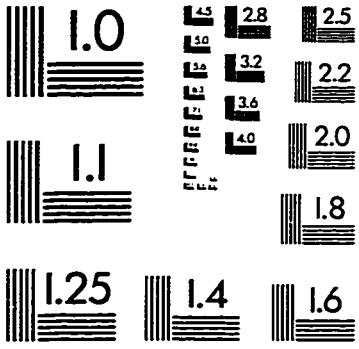
Trivedi, K.K., B.K. O'Neill, J.R. Roach, and R.M. Wood, "Systematic Energy Relaxation in MER Heat Exchanger Networks," *Comput. Chem. Eng.*, **14**, 601 (1990).

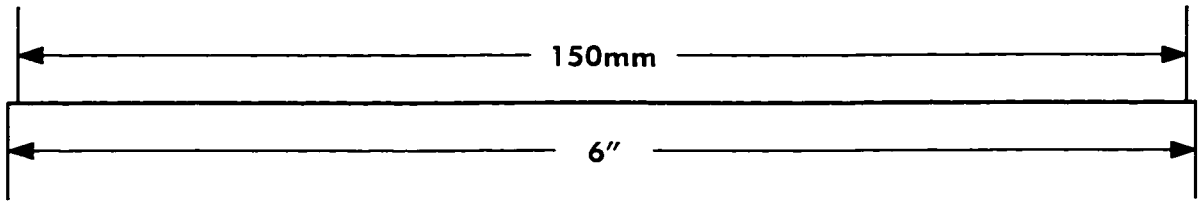
Zhu, X.X., B.K. O'Neill, J.R. Roach, and R.M. Wood, "Kirchhoff's Law and Loop-breaking for the Design of Heat Exchanger Networks," *Chem. Eng. Comm.*, **126**, 141 (1993).

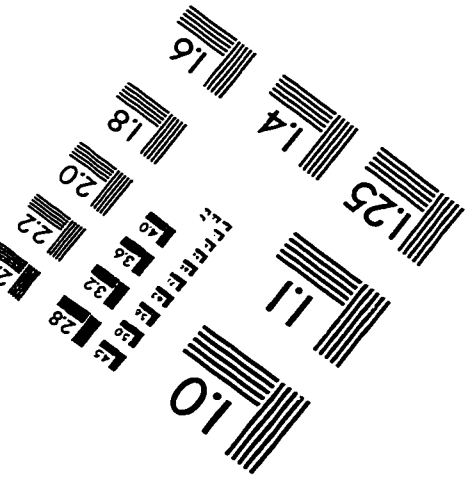
IMAGE EVALUATION TEST TARGET (QA-3)

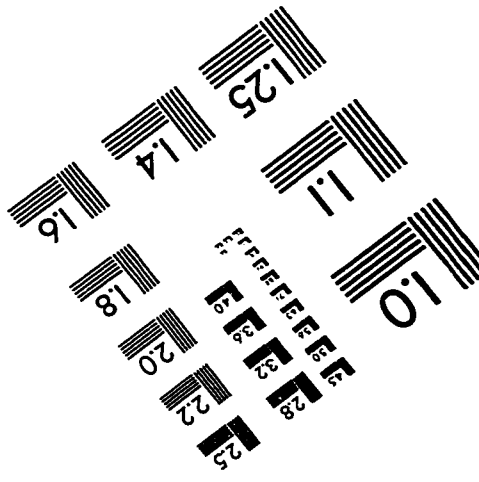












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989