Design of Control Configurations for Complex Process Networks

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Seongmin Heo

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Advised by Prodromos Daoutidis

May 2015

Acknowledgments

First of all, I would like to thank my advisor, Professor Prodromos Daoutidis, for his guidance and support. This thesis would have not been completed without his help. I should also thank Drs. Sujit Jogwar and Srinivas Rangarajan for their help during the early phase of my research. I would also like to thank Professors Xiang Cheng, Satish Kumar and Tryphon Georgiou for agreeing to serve on my thesis defense committee and for their valuable comments during my defense.

The Daoutidis group office has always been a nice, warm place for me. I would like to thank my lab mates Abdulla Malek, Nahla Al Amoodi, Udit Gupta, Nitish Mittal, Michael Zachar, Andrew Allman and Manjiri Moharir for making the office enviroment vivid and lively. I would also like to thank my Korean friends in the department for all the moments that we shared. Also, the support from my old friends back in Korea has been a huge driving force during my PhD research.

Last but not least, I must express my gratitude and love to my family. Their endless love and support have made me who I am now.

To My Family

Abstract

Tight integration is the rule rather than the exception in chemical and energy plants. Despite the significant economic benefits which result from efficient utilization of energy/material resources, effective control of plants with such integration becomes challenging; the network-level dynamics emerging from process interconnections and the model complexity of such plants limit the effectiveness of decentralized control approaches traditionally followed in plant-wide control. The development of effective control methods for complex integrated plants is a challenging, open problem.

This thesis proposes methods to develop effective control strategies for two classes of process networks. In the first part of the thesis, a class of process networks, in which slow network-level dynamics is induced by large rates of energy and/or material recycle, is considered. A graph theoretic algorithm is developed for such complex material integrated process networks to i) identify the material balance variables evolving in each time scale, and ii) design hierarchical control structures by classifying potential manipulated inputs and controlled outputs in each time scale. The application of a similar algorithm developed for energy integrated networks to representative chemical

processes is also presented.

The second part of the thesis focuses on generic process networks where tight integration is not necessarily reflected on a segregation of energy and/or material flows. A method is developed to systematically synthesize control configurations with favorable structural coupling, using relative degree as a measure of such coupling. Hierarchical clustering methods are employed to generate a hierarchy of control configurations ranging from fully decentralized ones to a fully centralized one. An agglomerative hierarchical clustering method is first developed, in which groups of inputs/outputs are merged successively to form fewer and larger groups that are strongly connected topologically. Then, a divisive hierarchical clustering method is developed, in which groups of inputs/outputs are decomposed recursively into smaller groups. The developed methods are applied to typical chemical process networks.

Contents

Acknowledgments **i**

Abstract **iii**

Table of Contents **v**

List of Tables **viii**

List of Figures **xi**

1 Introduction **1**

1.1 Hierarchical control of tightly integrated process networks 2

1.2 Input/output pairing through community detection 4

1.3 Thesis scope and organization 7

2 Hierarchical control of complex energy integrated process networks **9**

2.1 Introduction 9

2.2 Graph reduction for complex energy integrated networks 11
2.3 Energy integrated reactor-heat exchanger networks 18
2.3.1 Toluene hydrodealkylation process 18
2.3.2 Energy integrated solid oxide fuel cell system 50
2.4 Energy integrated distillation columns 53
2.4.1 Vapor recompression distillation 57
2.4.2 Heat integrated network of distillation columns 68

3 Hierarchical control of complex material integrated process networks 74
3.1 Introduction 74
3.2 Analysis of multi-time scale dynamics: singular perturbations based framework 76
3.3 Graph theoretic analysis of multi-time scale material dynamics 82
3.3.1 Process flow graph 82
3.3.2 Graph-theoretic algorithm 86
3.4 Case study 96
3.4.1 HMF production process 96
3.4.2 Vinyl acetate process 102

4 Automated synthesis of control configurations based on structural coupling 113
4.1 Introduction 113
4.2 Relative degree as a measure of structural coupling 115
4.3 Control configuration synthesis based on structural coupling 118
4.3.1 Synthesis of decentralized control configurations 119
4.3.2 Hierarchical clustering of input/output pairs 126
4.3.3 Algorithm for automation of synthesis procedure 137

4.4 Case study – energy integrated solid oxide fuel cell (SOFC) system . . . 139
4.4.1 Process description . . . 139
4.4.2 Automated synthesis of control configurations . . . 141

5 Control-relevant decomposition of process networks based on structural coupling 153

5.1 Relative degree and structural coupling . . . 154
5.2 Control configuration synthesis by network decomposition . . . 155
5.2.1 Divisive hierarchical clustering of process networks based on relative degrees . . . 155
5.2.2 Optimization problem formulation . . . 158
5.3 Case study – Toluene hydrodealkylation process . . . 161
5.3.1 Agglomerative clustering . . . 164
5.3.2 Divisive clustering . . . 168

6 Summary and Future Research 178

6.1 Thesis summary . . . 178
6.2 Future research . . . 180
6.2.1 Complex networks combining energy and material integration . . 180
6.2.2 Analysis of input-to-output paths . . . 182
6.2.3 Optimal clustering . . . 183
6.2.4 Clustering using aggregate topological closeness measures . . . 184

Bibliography 186

List of Tables

2.1 Summary of orders of magnitude of the energy flows of the HDA process reaction part 23

2.2 Node list of the HDA energy flow graph 35

2.3 Summary of time scale analysis for the HDA process 37

2.4 Nominal values of the state variables and the process parameters 42

2.5 Controller parameters for the HDA process 45

2.6 Node list of the energy integrated SOFC system 52

2.7 Summary of orders of magnitude of the energy flows of the energy integrated SOFC system 54

2.8 Summary of time scale analysis for the energy integrated SOFC process 56

2.9 Node list of VRD energy flow graph 59

2.10 Summary of orders of magnitude of the energy flows of the VRD 60

2.11 Summary of time scale analysis for the direct VRD 66

2.12 Node list of five-component separation energy flow graph 71

2.13 Summary of orders of magnitude of the energy flows of five-component separation network 72

3.1 Schematic representation of different types of nodes and edges 84
3.2 Node list of $\mathcal{G}$ of the reactor-separator network 88
3.3 Edge list of $\mathcal{G}$ of the reactor-separator network 89
3.4 Summary of the dynamics of the reactor-separator network 95
3.5 Potential manipulated inputs and controlled outputs in each time scale: reactor-separator network 95
3.6 Node list of $\mathcal{G}$ of the HMF production process 100
3.7 Edge list of $\mathcal{G}$ of the HMF production process 100
3.8 Index list for the chemical species: HMF production process 100
3.9 Summary of the dynamics of the HMF production process in the time scale τ_2 101
3.10 Summary of the dynamics of the HMF production process in the time scale t 101
3.11 Potential manipulated inputs and controlled outputs in each time scale: HMF production process 102
3.12 Node list of $\mathcal{G}$ of the VAc process 105
3.13 Edge list of $\mathcal{G}$ of the VAc process 106
3.14 Index list for the chemical species: VAc process 106
3.15 Original and modified weights of the edges of $\mathcal{O}(1)$ 108
3.16 Summary of dynamics of the VAc process in the time scale τ_3 108
3.17 Summary of dynamics of the VAc process in the time scale τ_2 108
3.18 Summary of dynamics of the VAc process in the time scale t 109

3.19 Potential manipulated inputs and controlled outputs in each time scale: VAc process 110

4.1 Shortest paths between the inputs and the outputs of the simple CSTR example 125

4.2 Node list of the energy integrated SOFC system equation graph in Figure 4.5 145

4.3 Block decentralized control configurations generated from the energy integrated SOFC system, solution 1 150

4.4 Block decentralized control configurations generated from the energy integrated SOFC system, solution 2 150

5.1 Manipulated inputs and controlled ouputs for the HDA process 163

5.2 Cluster list for the HDA process: agglomerative clustering 166

5.3 Block decentralized control configurations for the HDA process: agglomerative clustering 168

List of Figures

2.1 Composite units for prototype networks 13
2.2 Toluene hydrodealkylation process 19
2.3 Energy representation of the reaction part of the HDA process 22
2.4 Energy flow graph of the HDA process 34
2.5 Subgraphs of the HDA reaction part energy flow graph 38
2.6 Evolution of the furnace temperature 43
2.7 Evolution of the separator temperature 44
2.8 Multi-loop control configuration for the HDA process 45
2.8 Closed-loop responses for set-point tracking 47
2.8 Closed-loop responses for disturbance rejection 49
2.9 Energy integrated SOFC system 50
2.10 Energy flow graph of the energy integrated SOFC system 51
2.11 Subgraphs of the energy integrated SOFC energy flow graph 55

2.12 Direct vapor recompression distillation (1: Distillation column, 2: Compressor, 3: Combined reboiler-condenser, 4: Trim condenser, 5: Reflux drum, 6: Auxiliary cooler) 57
2.13 Energy flow graph of direct VRD 58
2.14 Subgraphs of the VRD energy flow graph 61
2.15 Heat integrated distillation column network for five-component separation 69
2.16 Energy flow graph of five-component separation 70
2.17 Subgraphs of the five-component separation energy flow graph 73

3.1 Flowchart of the analysis of multi-time scale dynamics using singular perturbations 80
3.2 A simple reactor-separator network 85
3.3 Process flow graph of the reactor-separator network 87
3.4 Simple network with a pure recycle 91
3.5 Simple network with a partial recycle 92
3.6 Subgraphs of the reactor-separator network 94
3.7 Process flow diagram of the HMF production process 97
3.8 Process flow graph of the HMF production process 98
3.9 Subgraphs of the HMF production process 99
3.10 Process flow diagram of the VAc process 103
3.11 Process flow graph of the VAc process 104
3.12 Subgraphs of the VAc process 107
3.13 Evolution of the total holdup 111
3.14 Evolution of the species holdup in the liquid phase of the separator . . . 112

4.1 A simple CSTR 122
4.2 Equation graph of the simple CSTR example 124

4.3 Hierarchical clustering results for the simple CSTR example 137
4.4 Process flow diagram of the energy integrated SOFC system 139
4.5 Part of the equation graph of the energy integrated SOFC system 143
4.6 Control configuration for solution 1 147
4.7 Hierarchical clustering result for the energy integrated SOFC system, solution 1 148
4.8 Hierarchical clustering result for the energy integrated SOFC system, solution 2 ($C(\{6\}) = \{u_2, y_1\}$, $C(\{7\}) = \{u_1, y_2\}$) 149
4.9 Subnetworks of the energy integrated SOFC system 151
4.10 Subnetworks controlled by different controllers, solution 1-3 152

5.1 Toluene hydrodealkylation process 161
5.2 Hierarchical clustering result for the HDA process: agglomerative clustering 167
5.3 Hierarchical clustering result for the HDA process: divisive clustering, solution 1 174
5.4 Hierarchical clustering result for the HDA process: divisive clustering, solution 2 175
5.5 Subnetworks of the HDA process, solution 1 176
5.6 Subnetworks of the HDA process, solution 2 177

6.1 Flowchart of the control structure design framework proposed in the thesis 181

CHAPTER 1

Introduction

Control structure design, i.e. the selection and pairing of manipulated inputs and controlled outputs, is a classic problem in control that has received a lot of attention in the literature (e.g. [1]). In process control in particular, this problem has been studied extensively in the context of plant-wide control design (e.g. [2]). Examples include:

- heuristic-based approaches whereby the selection and pairing are performed following logical rules [3–5],
- the concept of self-optimizing control for selecting controlled outputs as the basis for control structure design [6–8],
- the concept of relative gain array (RGA) [9–11] and block RGA [12–14], and
- optimization based approaches using for example economic criteria [15–18].

Process integration has been a key enabler of efficient operation of chemical plants, as it provides significant economic benefits by minimizing consumption of energy/material resources. However, such benefits are offered at the expense of several operational and control challenges: energy/material recycles result in positive feedback, which is responsible for complex dynamic behavior (e.g. non-minimum phase behavior, chaotic behavior) [19–27] and slower network-level dynamics [28–32]; disturbances occurring at one process unit may propagate in the network [33–36]; also there are typically reduced degrees of freedom available for control due to the tight coupling between units.

Such challenges cannot be effectively overcome through the implementation of fully decentralized control approaches traditionally followed in plant-wide control [37, 38]. The design of control systems for tightly integrated plants thus has become a major problem in process control.

1.1 Hierarchical control of tightly integrated process networks

Process integration often leads to configurations with energy/material flows of different orders of magnitude due to the recovery and recycle of large amounts of mass/energy. The presence of such segregation introduces stiffness and multiple time scale dynamics in the tightly integrated process networks, and the controllers designed on the basis of such models are generally ill-conditioned [39]. It is thus necessary to derive non-stiff, reduced-order models, which describe the dynamics in each time scale, to be used to design well-conditioned controllers.

To this end, in the previous work in our group [30, 31, 40–42], the mathematical structure of the dynamic models of several prototype networks with a *single* recycle loop was analyzed. It was shown that whenever integration results in large rates of

recovery and recycle of energy and/or material (compared to the input/output flows), a natural time-scale hierarchy develops: individual units evolve in a fast time scale (and are affected by the large internal and recycle flows) and slower network-level dynamics emerge (which are affected by the small external flows). The underlying dynamic models, which were shown to exhibit a singularly perturbed form, were used as the basis for the development of:

- a systematic singular perturbations analysis framework for the derivation of non-stiff, reduced-order models in each time scale.
- a framework for the selection of possible manipulated inputs for addressing process unit level control objectives and network level control objectives in the different layers of the hierarchy.

However, industrial plants rarely involve a single recycle loop. Complex process networks typically comprise of combinations of the above prototype networks, resulting in *multi-loop* integrated configurations (e.g. [43]). Such configurations have the potential to demonstrate multi-time scale dynamics (rather than two-time scale dynamics as in the case of the above prototype networks), which may not be apparent by simple observation. The development of an effective model reduction framework to analyze such dynamics is an open problem, whose solution is key to developing effective control strategies.

Although, in principle, a sequential application of singular perturbations can be adopted to this end (e.g. [41, 44]), this approach becomes cumbersome and time consuming with increasing size of process networks. A promising approach to develop a *scalable* framework is to utilize the extensive work in graph theory (e.g. [45]). A graph is a collection of vertices and edges connecting them, which is a natural tool to represent a wide range of networks such as citation networks [46–49], metabolic networks [50, 51],

and ecological networks [52, 53].

In previous work in our group [54], a graph theoretic framework for complex energy integrated process networks has been developed. A graph formalism, namely *energy flow graph*, has been proposed, which captures the energy flow structure (including the orders of magnitude of the different energy flows) of such networks. Then, such a formalism has been used as the basis to develop a graph theoretic algorithm to generate information on:

- the time scales where each unit evolves,
- the form of the reduced order models in each time scale,
- controlled outputs and potential manipulated inputs available in each time scale.

Representative examples of chemical processes will be considered in this thesis, to illustrate the application of the above framework and develop insights on effective control strategies for these processes. An extension of this framework will be also considered to analyze complex material integrated process networks.

1.2 Input/output pairing through community detection

There are numerous process networks where tight integration is not reflected on a segregation of energy/material flows. In such cases, tight integration results in intricate networks of interconnections via significant (but not necessarily large) energy/material flows. Such networks can also exhibit emerging network behavior due to interactions between the constituent units (which is also observed in other fields as documented in e.g. [55–59]).

Extensive research activity in process control has been devoted to develop control

approaches that account for such a networked nature in tightly integrated process networks [2, 16, 17, 60–66]. Key to the development of such approaches is the identification of constituent sub-systems within the network that can be effectively controlled and coordinated. Central to this problem is the pairing of manipulated inputs and controlled outputs, either in the form of single-input single-output controllers (fully decentralized control configurations) or multivariable ones (block decentralized ones).

This problem is closely related to the problem of identifying communities whose members interact strongly among them, yet are weakly coupled to the rest of the network members. Such a problem has been being studied extensively in network theory (e.g. [67]). Graph partition, i.e. community detection based on graph theory, has also become an active area of research (e.g. [68]). A typical community detection problem consists of three main steps:

- to define the community
- to define a quantitative criterion to assess the goodness of a partition
- to develop an algorithmic procedure to identify good partitions

For the problem of the pairing of inputs and outputs, a community can be defined as a group of inputs and outputs that are strongly connected topologically. One measure of topological closeness is the concept of relative degree which captures the directness of the effect of an input on an output, or the physical closeness between the two variables [69]. Then, a good partition can be defined as a partition with small intra-community relative degrees (i.e. relative degrees between the inputs and the outputs in the same community) and large inter-community relative degrees (i.e. relative degrees between the inputs and the outputs of different communities).

Hierarchical clustering has a potential to be adopted in the last step mentioned

above, as it has been a traditional technique to address the community detection problem [70, 71], especially for the cases in which a priori knowledge on the number of communities is not available, which is typically the case for the control structure design. Hierarchical clustering can be classified into two categories:

- Agglomerative hierarchical clustering is a bottom-up procedure, in which the hierarchy is built by merging two clusters successively, starting with the individual objects. At each level of clustering, two closest clusters are selected based on a *similarity* measure [72] (for the problem of the pairing of inputs and outputs, the notion of physical closeness, captured by relative degree, can be viewed as a measure of similarity), and merged. There exist several linkage criteria (e.g. single linkage, complete linkage) to evaluate the similarity between the new cluster and the old clusters [71].

- Divisive hierarchical clustering is a top-down procedure, in which the hierarchy is built by recursively decomposing clusters, starting with a single cluster containing all the objects. A *compactness* measure needs to be defined as a function of a similarity measure. Then, at each level of clustering, the loosest cluster is selected, and decomposed into more compact clusters.

The above procedures, which are complementary in nature, can be implemented together to obtain a fully resolved hierarchy of well-separated compact clusters. Thus, hierarchical clustering has a potential to provide a significant flexibility to the problem of control structure design. This novel approach to this classic problem will be pursued in this thesis.

1.3 Thesis scope and organization

The goal of this thesis is to develop systematic control structure design methods for complex process networks, which are generic and scalable. Specifically, the work presented in this thesis aims:

- to develop a graph theory based framework to analyze multi-time scale dynamics of complex integrated process networks, and to design hierarchical control structures which account for the underlying time scale multiplicity.

- to develop optimization based hierarchical clustering methods to identify groups of manipulated inputs and controlled outputs that are strongly connected topologically, and are thus block decentralized control configuration candidates.

The rest of the thesis is organized as follows. Chapter 2 illustrates the application of the graph reduction framework, which was developed previously in our group, to analyze complex energy integrated process networks. Two different classes of such networks, which are common in process systems, are considered: reactor-heat exchanger networks and energy integrated distillation column networks. For representative cases of such networks, the results from the graph reduction framework are compared with the analytical, singular perturbations-based reduction, and some insights on the dynamic structure of such networks, that affect their control, are provided.

In Chapter 3, a graph reduction framework for complex material integrated process networks is developed, which identifies the time scales where each process variable evolves, and the possible manipulated inputs in each time scale. First, a novel graph formalism, which captures the material flow structure in complex integrated process networks, is proposed. A graph theoretic algorithm is then developed, which essentially mimics the steps of the singular perturbations-based analysis and reduction.

Chapter 4 addresses the control structure design for more general complex integrated process networks. An integer optimization problem is formulated to identify optimal fully decentralized control configurations that minimize the overall structural coupling captured by relative degree. Then, an agglomerative hierarchical clustering procedure is proposed, which allows identifying groups of inputs and outputs that are strongly connected topologically.

A divisive hierarchical clustering procedure is developed in Chapter 5. A compactness measure for input/output clusters is defined as a function of relative degree. Then, an integer nonlinear optimization problem is formulated to decompose a cluster into smaller, more compact clusters, and the hierarchy of input/output clusters is identified through its successive application.

Finally, Chapter 6 summarizes the main results of this thesis, and proposes future research directions.

CHAPTER 2

Hierarchical control of complex energy integrated process networks*

2.1 Introduction

High energy efficiency has become a critical need in modern chemical plants. Energy integration has been an active topic of research [74,75], as it enables reduction in external utility consumption and, thus, provides cost benefits. It involves coupling of heating and cooling requirements of a plant to transfer energy from an energy source to an energy sink. However, energy integrated chemical plants are difficult to operate and control, especially in the context of transitions between different operating points.

In our previous work [31, 76], we have identified two prototype energy integrated networks, those with large energy recycle compared to the external energy flows, and those with large energy throughput. The disparate magnitudes of energy flows in such

*Graph Reduction of Complex Energy-Integrated Networks: Process Systems Applications, Seongmin Heo, Srinivas Rangarajan, Prodromos Daoutidis, and Sujit S. Jogwar, *AIChE Journal*, 60(3):995–1012, 2014 [73].

networks were shown to be at the origin of a two-time scale behavior and a segregation of flows that can be used as manipulated inputs in the different time scales. Exploiting these features, we developed model reduction methods using singular perturbations and a hierarchical control strategy wherein different control objectives are addressed in different time scales.

In practice, complex energy integrated chemical plants consist of combinations of energy recycle and energy throughput networks, often with large energy flows, thus showing the potential of multi-time scale energy dynamics [43]. A detailed mathematical analysis through successive application of singular perturbations can, in principle, be used to analyze such networks. However, such an analysis becomes cumbersome as the size of the network increases. To this end, we have proposed a generic and scalable graph-theoretic framework which can be used to analyze such networks [54, 77]. The developed framework relies on knowledge of the energy flow structure in the network and the orders of magnitude of the different energy flows to generate information on i) the time scales where each unit evolves, ii) the form of the reduced order models in each time scale, and iii) controlled outputs and potential manipulated inputs available in each time scale. Such information allows us to develop a hierarchical control strategy for such networks.

In this chapter, we consider two different classes of such networks which are common in process systems: reactor-heat exchanger networks and energy integrated distillation column networks. For representative cases of such networks, we illustrate the application of the graph reduction method, compare it with the analytical, singular perturbations based reduction, and identify and discuss insights on the dynamic structure of such networks that affect their control.

2.2 Graph reduction for complex energy integrated networks

In this section, we describe briefly the main features of the developed graph reduction framework (for details see [54, 77]).

The energy dynamics of a generic complex energy integrated network, which consists of N units and energy flows, h_i, spanning m orders of magnitude, can be described by the following equations:

$$\frac{d\mathbf{H}}{dt} = \sum_{i=0}^{m-1} \frac{1}{\varepsilon_i} \mathbf{F}_i \mathbf{g}_i(\mathbf{H}, \mathbf{u}_i) \tag{2.1}$$

where the energy flows of different orders of magnitude are indexed by i. ε_i are small parameters, which are defined as ratios of nominal steady state flows ($\varepsilon_i = h_{0,s}/h_{i,s}$), such that $\varepsilon_i \ll 1$ and $\varepsilon_{i+1}/\varepsilon_i \ll 1$. In the subsequent analyses, we assume ε_0 to be equal to 1. $\mathbf{H}$ is the vector of enthalpy of the N process units, $\mathbf{g}_i$ are vectors with contributions from the energy flows of $\mathcal{O}(1/\varepsilon_i)$, and $\mathbf{F}_i$ are the corresponding selector matrices such that $\mathbf{F}_i(j,k) = 1$ if the energy balance equation of the unit k includes the j-th element of $\mathbf{g}_i$ and 0 otherwise. $\mathbf{u}_i$ represent scaled energy flows that can be used as manipulated variables.

Note that Eq. (2.1) is a singularly perturbed system with multiple small parameters. Model reduction can be performed sequentially by considering each time scale $\tau_i = t/\varepsilon_i$ (starting from the fastest) and setting the small parameter ε_i to 0, to obtain reduced models of the slow and fast dynamics in this time scale. For each time scale τ_i, the description of the dynamics can be obtained in the limit $\varepsilon_i \to 0$ as:

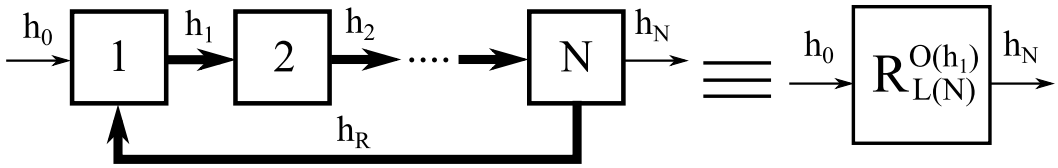
$$\frac{d\mathbf{H}_i}{d\tau_i} = (\mathbf{F}_i^T \mathbf{F}_i) \hat{\mathbf{g}}_i(\mathbf{H}_i \mathbf{u}_i)$$

where $\mathbf{H}_i = \mathbf{F}_i^T \mathbf{H}$ represent the unit enthalpies evolving in the time scale τ_i, and $\hat{\mathbf{g}}_i$ represents the correspondingly adjusted vector. Quasi-steady state constraints for the time scale τ_i can be identified as $\hat{\mathbf{g}}_i(\mathbf{H}_i, \mathbf{u}_i) = 0$. Differentiating these constraints with respect to time, we obtain:

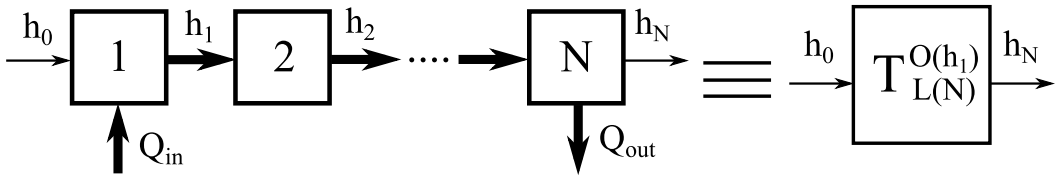
$$0 = \frac{\partial \hat{\mathbf{g}}_i}{\partial t} = \frac{\partial \hat{\mathbf{g}}_i}{\partial \mathbf{H}_i} \frac{d\mathbf{H}_i}{dt}$$

If the constraints are linearly independent, the matrix $\partial \hat{\mathbf{g}}_i / \partial \mathbf{H}_i$ is nonsingular, and we get $d\mathbf{H}_i/dt = 0$. This implies that the enthalpies $\mathbf{H}_i$ evolve only in the time scale τ_i, and not in slower ones. If the constraints are linearly dependent, the matrix $\partial \hat{\mathbf{g}}_i / \partial \mathbf{H}_i$ becomes singular, and the i-th time scale dynamics of $\mathbf{H}_i$ are accompanied by a slower evolution in subsequent time scales. The model of the slower dynamics in this case is an index two differential algebraic equation (DAE) system with the dynamic equations being the differential equations, the quasi-steady state constraints being the algebraic equations, and the limiting terms $\lim_{\varepsilon_i \to 0} \tilde{\mathbf{g}}_i / \varepsilon_i$ being the algebraic variables, where $\tilde{\mathbf{g}}_i$ represents the linearly independent subset of the quasi-steady state constraints. Large throughputs, splitting, and mixing structures give rise to linearly independent constraints [76], whereas large recycles give rise to linearly dependent constraints [31].

The above reduction can be performed in a graph-theoretic framework. The energy flow structures of complex energy integrated process networks can be represented as *energy flow graphs* with nodes corresponding to the individual units of the networks, and directed and weighted edges corresponding to the energy flows. In this setting, energy recycles and throughputs are energy flow cycles and energy flow paths, respectively. Energy flows with different orders of magnitude can be distinguished by using lines of different thickness (corresponding to different edge weights) in such graphs. Also, we use nodes with solid borders to represent normal nodes (i.e. nodes for the process units),



(a) Recycle



(b) Throughput

Figure 2.1: Composite units for prototype networks

and nodes with dashed borders to represent auxiliary nodes (i.e. nodes for external energy sources/sinks).

Figure 2.1a shows an energy flow graph for a large recycle network. Such networks can be equivalently represented as a single composite node R. Figure 2.1b shows an energy flow graph of a network with large energy throughput. The units in such networks can be clustered to form a single composite node T. Such clustering reduces the complexity in the representation of the networks and facilitates graph reduction. In the following sections, networks with large energy recycle and networks with large energy throughput will be referred to as *prototype recycle* and *prototype throughput*, respectively.

The algorithm for the graph-theoretic analysis is provided below.

ComplexNetworkAnalysis($\mathcal{G}, W$)

1: Sort W in descending order

2: **for** $i = 1$ to Size(W)=m **do**

3: μ = W[i]

4: $\mathbf{g}_{\mu}(\mathbf{u}_{\mu}) = h_{1,s} \times Ebalance(\mathcal{G}, \mu)$;

5: $\mathcal{H}_{\mu} = InducedSubgraph(\mathcal{G}, \mu)$

6: $\mathcal{T}(\tau_{\mu})$ = nodes $\in \mathcal{H}_{\mu}$

7: **for** each node N $\in \mathcal{H}_{\mu}$ **do**

8: **if** N is a composite node **then**

9: add $\sum N[k]$ to $\mathcal{Y}(\tau_{\mu})$

10: **else**

11: add N to $\mathcal{Y}(\tau_{\mu})$

12: **end if**

13: **end for**

14: $\mathcal{U}(\tau_{\mu})$ = Edges in $\mathcal{H}_{\mu}$

15: C = $SmallestElementaryCycle(\mathcal{H}_{\mu})$

16: **while** C $\neq \phi$ **do**

17: $GraphReduce(\mathcal{G}$,C,$\mu)$

18: $GraphReduce$ $(\mathcal{H}_{\mu}$,C,$\mu)$

19: C = $SmallestElementaryCycle(\mathcal{H}_{\mu})$

20: **end while**

21: **if** size(RecycleTimes) $\neq 0$ **then**

22: $DAE_{\mu} = \mathbf{C}_{\mu^-}\mathbf{B}_{\mu^-}\mathbf{z}_{\mu^-}$

23: $Constraint_{\mu} = \tilde{\mathbf{g}}_{\mu^-}(\mathbf{u}_{\mu^-})$

24: **if** size(RecycleTimes) > 1 **then**

25: $AddConstraints_{\mu} = \sum_{j=1}^{size(RecycleTimes)} DAE_j$

26: **end if**

27: **end if**

if degree(N) = 0 for any node N $\in \mathcal{H}_\mu$ **then**

Add τ_μ to RecycleTimes

Add N_j to PureRecycles

All but 1 out of N_j should be controlled in this time scale

end if

if degree(N) $\neq$ 0 for all nodes N $\in \mathcal{H}_\mu$ **then**

Clear RecycleTimes, PureRecycles

end if

for all node N $\in \mathcal{H}_\mu$ such that degree(N) $\neq$ 0 **do**

if N is a composite node **then**

Remove $N[k]$ from PureRecycles

end if

Remove N from $\mathcal{G}$

end for

end for

Energy balance equations are

$$\frac{d\mathbf{H}}{dt} = \sum_{\mu=W[1]}^{W[m]} \frac{1}{\varepsilon_\mu} \mathbf{g}_\mu(\mathbf{H}, \mathbf{u}_\mu)$$

for all $\mu \in W$ **do**

Reduced order model in τ_μ is

$$\begin{aligned} \frac{d\mathbf{H}_\mu}{d\tau_\mu} &= \bar{\mathbf{g}}_\mu(\mathbf{H}, \mathbf{u}_\mu) + DAE_\mu \\ 0 &= Constraint_\mu + AddConstraints_\mu \end{aligned}$$

48: **end for**

49: **return** $\mathcal{T}, \mathcal{Y}, \mathcal{U}$

The algorithm has been implemented as a computational tool written in C++ to automate this graph reduction analysis and has been used in all examples studied herein. In what follows, we briefly describe the graph reduction and analysis method and its implementation through the algorithm.

Complex networks comprise of combinations of energy recycles and throughputs, some of which are prototype. The developed graph reduction method begins with a user specified classification of the energy flows in classes of distinct orders of magnitude, W. Starting with the largest flows which correspond to the fastest dynamics, and proceeding to all subsequent classes of flows, it first extracts a *subgraph*, using the subroutine *InducedSubgraph*, corresponding to each time scale. It thus identifies the units evolving in each time scale (those with nodes in that subgraph) and proceeds to identify prototype patterns, using the subroutine *SmallestElementaryCycle*, and simplify them through clustering using the subroutine *GraphReduce*. Throughput blocks are removed from the original energy flow graph, before proceeding to the subsequent time scales, since the energy dynamics of all the units of prototype throughputs evolve in a single time scale.

Next, it constructs the canonical forms of the original full order dynamic model as well as the reduced order dynamic models for the energy balance variables (enthalpies) in each time scale. The energy balance equations for the full order model (or the reduced models) are derived based on the connectivity information of the energy flow graph (or subgraph) using the subroutine *Ebalance*. For the reduced model, the method also predicts the linear independency of the quasi-steady state constraints from the existence of auxiliary nodes connected to individual composite nodes. Specifically, if there exists no auxiliary node connected to a composite node in a subgraph, the quasi-steady state

constraints resulting from the corresponding dynamic model of that composite node are linearly dependent. In this case, the composite node is added to the set *PureRecycles* and the corresponding time scale is added to the set *RecycleTimes*. For a specific time scale, *Constraint* represents the linearly independent constraints obtained from the previous time scale, and *DAE* represents the contribution of the previous time scale dynamics to the current time scale dynamics.

Finally, the method classifies the control objectives and the manipulated inputs available in each time scale. Specifically, the enthalpy of any node in a subgraph can be controlled in the corresponding time scale using any edge in the same subgraph as the manipulated input. Note that if a composite node does not have any auxiliary node connected to it (i.e. a prototype recycle composite node), one cannot simultaneously control the enthalpies of all the units, comprising that composite node, as the dynamic equations in this case are linearly dependent. Thus, all but one of the units for such a composite node can be controlled in the corresponding time scale, and the remaining unit can be controlled in the subsequent time scale.

To present the results obtained from the graph-theoretic analysis, we use the following notations:

- τ_i: The i-th time scale defined by t/ε_i, where ε_i is a small parameter.
- $\mathcal{G}$: The original energy flow graph of the network.
- $\mathcal{H}_i$: The subgraph of $\mathcal{G}$ corresponding to the time scale τ_i. It contains all and only the energy flows of the order of magnitude i.
- $\mathcal{T}(\tau_i)$: The set of nodes whose enthalpy evolves in the time scale τ_i.
- $\mathcal{Y}(\tau_i)$: The set of nodes whose enthalpy needs to be controlled in the time scale τ_i.

- $\mathcal{U}(\tau_i)$: The set of edges which can be used as the manipulated input in the time scale τ_i.

In what follows, we analyze relevant example chemical process networks using this graph-theoretic framework.

2.3 Energy integrated reactor-heat exchanger networks

For processes with high temperature reactions, there are opportunities for energy integration as the outlet stream of the reactor, which is at high temperature, can be used to provide energy to other process streams; such integration can be achieved through heat exchangers, thus forming reactor-heat exchanger networks. In what follows, we analyze two typical examples of such networks.

2.3.1 Toluene hydrodealkylation process

Process description

Let us consider a design alternative (alternative 1) of the toluene hydrodealkylation (HDA) process proposed by Terrill and Douglas [78] as shown in Figure 2.2. In this process, the reactant mixture is preheated in a feed-effluent heat exchanger (FEHE), where the mixture is completely vaporized, and is further heated using a furnace. It is then fed to an adiabatic plug-flow reactor where the following reactions occur:

$$C_6H_5CH_3 + H_2 \rightarrow C_6H_6 + CH_4 \qquad \text{Reaction 1}$$

$$2C_6H_6 \rightleftharpoons (C_6H_5)_2 \qquad \text{Reaction 2}$$

where benzene, which is the main product of the process, is produced through an irreversible exothermic reaction ($\Delta H_1 = -41.826\text{kJ/mol}$). The reactor effluent is cooled

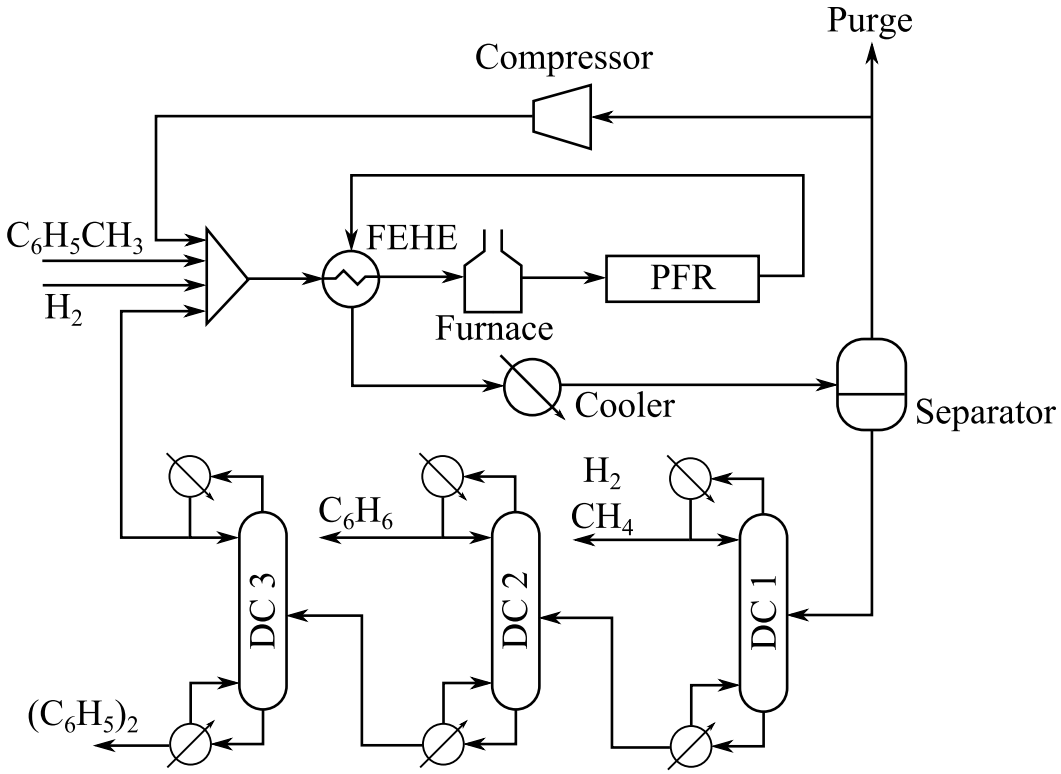


Figure 2.2: Toluene hydrodealkylation process

using the FEHE and the cooler, and is fed to the separator. A majority of the vapor outlet of the separator is compressed, and fed back to the reactor, while the residual vapor stream is purged. The liquid outlet of the separator goes through a series of distillation columns. In the stabilizer column, hydrogen and methane gas are removed from the liquid stream. In the product column, benzene product is produced as the distillate. Diphenyl is the bottoms of the recycle column, and the distillate of the recycle column, which is the unreacted toluene, is recycled.

We first perform the detailed mathematical analysis using singular perturbations as the model reduction tool. Then, the graph-theoretic analysis is applied, and the results are compared. Finally, via numerical simulations, the energy dynamics of the process is investigated to confirm the analysis results.

Detailed analysis

In this analysis, we focus on the reaction part of the process, where we observe the following features, which give rise to a potential of multi-time scale energy dynamics:

- The reactant mixture is completely vaporized in the FEHE, indicating that a large amount of energy is recovered through the FEHE.
- There is a large material recycle fed back to the reactor.

Remark 2.1. *Note that the separation part of the network, which consists of three distillation columns, is not included in the analysis. The reason is that the energy integration in that part is dominated by large energy throughputs, which do not lead to time scale multiplicity in the energy dynamics. More detailed justification and examples of energy integrated distillation column networks that lead to time scale multiplicity in the energy dynamics can be found in the subsequent section.*

The governing equations of the energy dynamics of the network can be written as follows:

$$\begin{aligned}
\frac{dT_m}{dt} &= \frac{R(T_{cp}-T_m)}{V_m} + \frac{F_{tf}(T_{tf}-T_m)}{V_m} + \frac{F_{hf}(T_{hf}-T_m)}{V_m} + \frac{F_{tr}(T_{tr}-T_m)}{V_m} \\
\frac{dT_c}{dt} &= \frac{F(T_m-T_c)}{V_c} + \frac{Q_{rec}}{\rho C_p V_c} \\
\frac{dT_f}{dt} &= \frac{F(T_c-T_f)}{V_f} + \frac{Q_f}{\rho C_p V_f} \\
\frac{dT_r}{dt} &= \frac{F(T_f-T_r)}{V_r} - \frac{\mathbf{\Delta H}\cdot\mathbf{r}}{\rho C_p} \\
\frac{dT_h}{dt} &= \frac{F(T_r-T_h)}{V_h} - \frac{Q_{rec}}{\rho C_p V_h} \\
\frac{dT_{cl}}{dt} &= \frac{F(T_h-T_{cl})}{V_{cl}} - \frac{Q_{cl}}{\rho C_p V_{cl}} \\
\frac{dT_{sp}}{dt} &= \frac{R(T_{cl}-T_{sp})}{V_{sp}} + \frac{F_P(T_{cl}-T_{sp})}{V_{sp}} + \frac{F_l(T_{cl}-T_{sp})}{V_{sp}} \\
\frac{dT_{cp}}{dt} &= \frac{R(T_{sp}-T_{cp})}{V_{cp}} + \frac{W}{\rho C_p V_{cp}}
\end{aligned} \tag{2.2}$$

with

$$F = R + F_{tf} + F_{hf} + F_{tr}$$

$$Q_{rec} = UA\frac{(T_r-T_c)-(T_h-T_m)}{ln\left[(T_r-T_c)/(T_h-T_m)\right]}$$

where the subscripts m, c, f, r, h, cl, sp and cp are used to represent the mixer, the cold channel of the FEHE, the furnace, the reactor, the hot channel of the FEHE, the cooler, the separator and the compressor, respectively. Also, subscripts tf, hf and tr are used for the toluene feed stream, the hydrogen feed stream and the toluene recycle stream, respectively. V is the volume of the unit, and Q is the duty. F is the volumetric flow rate, and R is the volumetric flow rate of the gas recycle. W is the power input to the compressor. U represents the overall heat transfer coefficient and A is the heat transfer

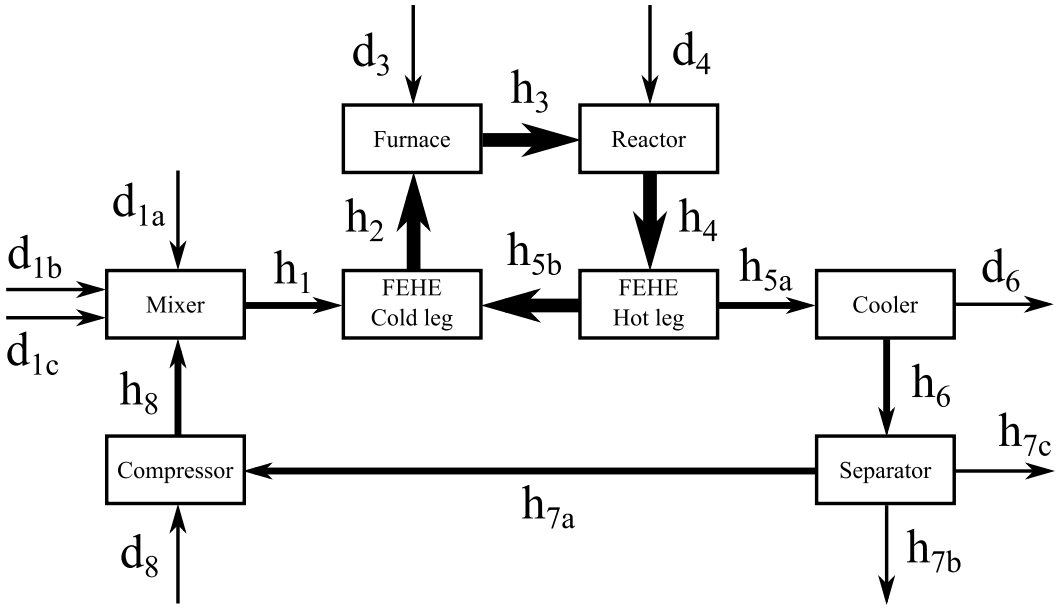


Figure 2.3: Energy representation of the reaction part of the HDA process

area. $\Delta\mathbf{H} = [\Delta H_1, \Delta H_2]^T$ is the vector of the heats of reaction and $\mathbf{r} = [r_1, r_2]^T$ is the vector of the reaction rates, where the subscripts 1 and 2 correspond to the reaction 1 and reaction 2, respectively.

The energy representation of the reaction part of the network is shown in Figure 2.3. The orders of magnitude of the energy flows are determined based on the sensible heat content of the flows as shown in Table 2.1. ε_1 and ε_2 are small parameters ($\varepsilon_2 \ll \varepsilon_1 \ll 1$), which are the ratios of the energy input through the toluene feed stream (d_{1a}) to the rate of energy recycle through the gas recycle (h_8) and the rate of energy transferred inside the FEHE (h_{5b}), respectively, i.e.:

$$\varepsilon_1 = \frac{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s}{[R\rho C_p(T_{cp} - T_{\text{ref}})]_s}$$
$$\varepsilon_2 = \frac{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s}{Q_{rec,s}}$$

where T_{ref} is a reference temperature, and subscript s represents a steady state value.

Energy flow	mC_pT(kW)	Order of magnitude
h_1	6676	$\mathcal{O}(1/\varepsilon_1)$
h_2	33183	$\mathcal{O}(1/\varepsilon_2)$
h_3	33746	$\mathcal{O}(1/\varepsilon_2)$
h_4	36036	$\mathcal{O}(1/\varepsilon_2)$
h_5	10102	$\mathcal{O}(1/\varepsilon_1)$
h_6	8249	$\mathcal{O}(1/\varepsilon_1)$
h_{7a}	4993	$\mathcal{O}(1/\varepsilon_1)$
h_{7b}	1996	$\mathcal{O}(1)$
h_{7c}	688	$\mathcal{O}(1)$
h_8	5642	$\mathcal{O}(1/\varepsilon_1)$
d_{1a}	523	$\mathcal{O}(1)$
d_{1b}	179	$\mathcal{O}(1)$
d_{1c}	331	$\mathcal{O}(1)$

Table 2.1: Summary of orders of magnitude of the energy flows of the HDA process reaction part

We define the $\mathcal{O}(1)$ steady state ratios:

$$k_{hf} = \frac{[F_{hf}\rho C_p(T_{hf} - T_{\text{ref}})]_s}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s} \qquad k_{tr} = \frac{[F_{tr}\rho C_p(T_{tr} - T_{\text{ref}})]_s}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s}$$

$$k_{qf} = \frac{Q_{f,s}}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s} \qquad k_{hr} = \frac{[\Delta\mathbf{H}\cdot\mathbf{r}V_r]_s}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s}$$

$$k_{qcl} = \frac{Q_{cl,s}}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s} \qquad k_P = \frac{[F_P\rho C_p(T_s - T_{\text{ref}})]_s}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s}$$

$$k_l = \frac{[F_l\rho C_p(T_s - T_{\text{ref}})]_s}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s} \qquad k_w = \frac{W_s}{[F_{tf}\rho C_p(T_{tf} - T_{\text{ref}})]_s}$$

$$k_m = \frac{[F\rho C_p(T_m - T_{\text{ref}})]_s}{[R\rho C_p(T_{cp} - T_{\text{ref}})]_s} \qquad k_h = \frac{[F\rho C_p(T_h - T_{\text{ref}})]_s}{[R\rho C_p(T_{cp} - T_{\text{ref}})]_s}$$

$$k_{cl} = \frac{[F\rho C_p(T_{cl} - T_{\text{ref}})]_s}{[R\rho C_p(T_{cp} - T_{\text{ref}})]_s} \qquad k_{sp} = \frac{[R\rho C_p(T_{sp} - T_{\text{ref}})]_s}{[R\rho C_p(T_{cp} - T_{\text{ref}})]_s}$$

$$k_c = \frac{[F\rho C_p(T_c - T_{\text{ref}})]_s}{Q_{rec,s}} \qquad k_f = \frac{[F\rho C_p(T_f - T_{\text{ref}})]_s}{Q_{rec,s}}$$

$$k_r = \frac{[F\rho C_p(T_r - T_{\text{ref}})]_s}{Q_{rec,s}}$$

and $\mathcal{O}(1)$ scaled energy flows:

$$
\begin{aligned}
u_{tf} &= \frac{F_{tf}\rho C_p(T_{tf}-T_{\text{ref}})}{[F_{tf}\rho C_p(T_{tf}-T_{\text{ref}})]_s} & u_{hf} &= \frac{F_{hf}\rho C_p(T_{hf}-T_{\text{ref}})}{[F_{hf}\rho C_p(T_{hf}-T_{\text{ref}})]_s}\\
u_{tr} &= \frac{F_{tr}\rho C_p(T_{tr}-T_{\text{ref}})}{[F_{tr}\rho C_p(T_{tr}-T_{\text{ref}})]_s} & u_{qf} &= \frac{Q_f}{Q_{f,s}}\\
u_{hr} &= \frac{\Delta\mathbf{H}\cdot\mathbf{r}V_r}{[\Delta\mathbf{H}\cdot\mathbf{r}V_r]_s} & u_{qcl} &= \frac{Q_{cl}}{Q_{cl,s}}\\
u_P &= \frac{F_P\rho C_p(T_s-T_{\text{ref}})}{[F_P\rho C_p(T_s-T_{\text{ref}})]_s} & u_l &= \frac{F_l\rho C_p(T_s-T_{\text{ref}})}{[F_l\rho C_p(T_s-T_{\text{ref}})]_s}\\
u_w &= \frac{W}{W_s} & u_{cp} &= \frac{R\rho C_p(T_{cp}-T_{\text{ref}})}{[R\rho C_p(T_{cp}-T_{\text{ref}})]_s}\\
u_m &= \frac{F\rho C_p(T_m-T_{\text{ref}})}{[F\rho C_p(T_m-T_{\text{ref}})]_s} & u_h &= \frac{F\rho C_p(T_h-T_{\text{ref}})}{[F\rho C_p(T_h-T_{\text{ref}})]_s}\\
u_{cl} &= \frac{F\rho C_p(T_{cl}-T_{\text{ref}})}{[F\rho C_p(T_{cl}-T_{\text{ref}})]_s} & u_{sp} &= \frac{R\rho C_p(T_{sp}-T_{\text{ref}})}{[R\rho C_p(T_{sp}-T_{\text{ref}})]_s}\\
u_{qrec} &= \frac{Q_{rec}}{Q_{rec,s}} & u_c &= \frac{F\rho C_p(T_c-T_{\text{ref}})}{[F\rho C_p(T_c-T_{\text{ref}})]_s}\\
u_f &= \frac{F\rho C_p(T_f-T_{\text{ref}})}{[F\rho C_p(T_f-T_{\text{ref}})]_s} & u_r &= \frac{F\rho C_p(T_r-T_{\text{ref}})}{[F\rho C_p(T_r-T_{\text{ref}})]_s}
\end{aligned}
$$

The energy dynamics equations in (2.2) thus become:

$$\begin{aligned}
\frac{dT_m}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_m}\left(\frac{u_{cp}-k_m u_m}{\varepsilon_1}+u_{tf}+k_{hf}u_{hf}+k_{tr}u_{tr}\right)\\
\frac{dT_c}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_c}\left(\frac{u_{qrec}-k_c u_c}{\varepsilon_2}+\frac{k_m u_m}{\varepsilon_1}\right)\\
\frac{dT_f}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_f}\left(\frac{k_c u_c-k_f u_f}{\varepsilon_2}+k_{qf}u_{qf}\right)\\
\frac{dT_r}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_r}\left(\frac{k_f u_f-k_r u_r}{\varepsilon_2}-k_{hr}u_{hr}\right)\\
\frac{dT_h}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_h}\left(\frac{k_r u_r-u_{qrec}}{\varepsilon_2}-\frac{k_h u_h}{\varepsilon_1}\right)\\
\frac{dT_{cl}}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_{cl}}\left(\frac{k_h u_h-k_{cl}u_{cl}}{\varepsilon_1}-k_{cl}u_{cl}\right)\\
\frac{dT_{sp}}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_{sp}}\left(\frac{k_{cl}u_{cl}-k_{sp}u_{sp}}{\varepsilon_1}-k_P u_P-k_l u_l\right)\\
\frac{dT_{cp}}{dt} &= \frac{[F_{tf}(T_{tf}-T_{\text{ref}})]_s}{V_{cp}}\left(\frac{k_{sp}u_{sp}-u_{cp}}{\varepsilon_1}+k_w u_w\right)
\end{aligned} \tag{2.3}$$

or, equivalently, in a vector form:

$$\frac{d\mathbf{T}}{dt}=\sum_{i=0}^{2}\frac{1}{\varepsilon_i}\mathbf{F}_i\mathbf{g}_i \tag{2.4}$$

with

$$\mathbf{T}=[T_m,T_c,T_f,T_r,T_h,T_{cl},T_s,T_{cp}]^T$$

$$\mathbf{F}_0=\begin{bmatrix} 1 & & \mathbf{0}_{1\times5}\\ & \mathbf{0}_{1\times6} & \\ \mathbf{0}_{2\times1} & \mathbf{I}_{2\times2} & \mathbf{0}_{2\times3}\\ & \mathbf{0}_{1\times6} & \\ \mathbf{0}_{3\times3} & & \mathbf{I}_{3\times3}\end{bmatrix}$$

$$\mathbf{g}_0 = [F_{tf}(T_{tf} - T_{ref})]_s \begin{bmatrix} \frac{u_{tf} + k_{hf}u_{hf} + k_{tr}u_{tr}}{V_m} \\ \frac{k_{qf}u_{qf}}{V_f} \\ -\frac{k_{hr}u_{hr}}{V_r} \\ -\frac{k_{qcl}u_{qcl}}{V_{cl}} \\ \frac{-k_P u_P - k_l u_l}{V_{sp}} \\ \frac{k_w u_w}{V_{cp}} \end{bmatrix}$$

$$\mathbf{F}_1 = \begin{bmatrix} \mathbf{I}_{2\times2} & \mathbf{0}_{2\times4} \\ \mathbf{0}_{2\times6} & \\ \mathbf{0}_{4\times2} & \mathbf{I}_{4\times4} \end{bmatrix}$$

$$\mathbf{g}_1 = [F_{tf}(T_{tf} - T_{ref})]_s \begin{bmatrix} \frac{u_{cp} - k_m u_m}{Vm} \\ \frac{k_m u_m}{V_c} \\ -\frac{k_h u_h}{V_h} \\ \frac{k_h u_h - k_{cl}u_{cl}}{V_{cl}} \\ \frac{k_{cl}u_{cl} - k_{sp}u_{sp}}{V_{sp}} \\ \frac{k_{sp}u_{sp} - u_{cp}}{V_{cp}} \end{bmatrix}$$

$$\mathbf{F}_2 = \begin{bmatrix} \mathbf{0}_{1\times 4} \\ \mathbf{I}_{4\times 4} \\ \mathbf{0}_{3\times 4} \end{bmatrix}$$

$$\mathbf{g}_2 = [F_{tf}(T_{tf} - T_{ref})]_s \begin{bmatrix} \dfrac{u_{qrec} - k_c u_c}{V_c} \\ \dfrac{k_c u_c - k_f u_f}{V_f} \\ \dfrac{k_f u_f - k_r u_r}{V_r} \\ \dfrac{k_r u_r - u_{qrec}}{V_h} \end{bmatrix}$$

where $\mathbf{I}$ is the identity matrix, and $\mathbf{0}$ is a matrix with all entries equal to 0.

Note that Eq. (2.4) suggests possibly three-time scale energy dynamics. We follow a sequential application of singular perturbations for deriving approximate models in each time scale.

Let us define the fast time scale $\tau_2 = t/\varepsilon_2$. Substituting this into Eq. (2.3) and taking the limit $\varepsilon_2 \to 0$, we obtain the description of the energy dynamics in the fast time scale:

$$\frac{d\mathbf{T}}{d\tau_2} = \mathbf{F}_2 \mathbf{g}_2 \tag{2.5}$$

We note that only the temperatures of the FEHE, the furnace and the reactor evolve in the fast time scale (i.e. $\mathbf{T}_2 = \{T_c, T_f, T_r, T_h\}$). From Eq. (2.5), we identify the quasi-steady state constraints, $\mathbf{g}_2 = 0$, and it can be verified that these equations are not linearly independent. Thus, we take a linearly independent subset $\tilde{\mathbf{g}}_2$, which is defined

by $\mathbf{g}_2 = \mathbf{B}_2\tilde{\mathbf{g}}_2$, with:

$$\mathbf{B}_2 = [F_{tf}(T_{tf} - T_{\text{ref}})]_s \begin{bmatrix} 1/V_c & 0 & 0 \\ 0 & 1/V_f & 0 \\ 0 & 0 & 1/V_r \\ -1/V_h & -1/V_h & -1/V_h \end{bmatrix}$$

$$\tilde{\mathbf{g}}_2 = \begin{bmatrix} u_{qrec} - k_c u_c \\ k_c u_c - k_f u_f \\ k_f u_f - k_r u_r \end{bmatrix}$$

Now, we take the same limit $\varepsilon_2 \to 0$ in the slow time scale t, and obtain:

$$\begin{aligned} \frac{d\mathbf{T}}{dt} &= \sum_{i=0}^{1} \frac{1}{\varepsilon_i}\mathbf{F}_i\mathbf{g}_i + \mathbf{F}_2\mathbf{B}_2\left(\lim_{\varepsilon_2 \to 0} \frac{\tilde{\mathbf{g}}_2}{\varepsilon_2}\right) \\ 0 &= \tilde{\mathbf{g}}_2 \end{aligned}$$

which describes the energy dynamics after the fast boundary layer. The limiting terms represent the differences between the large energy flows, which are indeterminate yet finite. These terms can be represented by a set of algebraic variables $\mathbf{z}_2$ to obtain the following DAE system:

$$\begin{aligned} \frac{d\mathbf{T}}{dt} &= \sum_{i=0}^{1} \frac{1}{\varepsilon_i}\mathbf{F}_i\mathbf{g}_i + \mathbf{F}_2\mathbf{B}_2\mathbf{z}_2 \\ 0 &= \tilde{\mathbf{g}}_2 \end{aligned} \tag{2.6}$$

The algebraic variables $\mathbf{z}_2$ can be evaluated by differentiating the constraints in Eq. (2.6):

$$\mathbf{z}_2 = -\left(\mathcal{L}_{\mathbf{F}_2\mathbf{B}_2}\tilde{\mathbf{g}}_2\right)^{-1}\left[\sum_{i=0}^{1}\frac{1}{\varepsilon_i}\mathcal{L}_{\mathbf{F}_i\mathbf{g}_i}\tilde{\mathbf{g}}_2\right] \tag{2.7}$$

where $\mathcal{L}$ represents the Lie derivative, which is defined as:

$$\mathcal{L}_{\mathbf{f(x)}}\mathbf{h}(\mathbf{x}) = \mathbf{J_h}\mathbf{f}$$

where $\mathbf{h}(\mathbf{x})$ is a vector, $\mathbf{f}(\mathbf{x})$ is a vector (or a matrix), and $\mathbf{J_h} = \{\partial h_i/\partial x_j\}$ is the jacobian matrix of $\mathbf{h}$.

Using the solution for $\mathbf{z}_2$ in Eq. (2.7), we can rewrite Eq. (2.6) as follows:

$$\begin{aligned}\frac{d\mathbf{T}}{dt} &= \sum_{i=0}^{1}\frac{1}{\varepsilon_i}\mathbf{F}_i\mathbf{g}_i - \mathbf{F}_2\mathbf{B}_2\left(\mathcal{L}_{\mathbf{F}_2\mathbf{B}_2}\tilde{\mathbf{g}}_2\right)^{-1}\left[\sum_{i=0}^{1}\frac{1}{\varepsilon_i}\mathcal{L}_{\mathbf{F}_i\mathbf{g}_i}\tilde{\mathbf{g}}_2\right]\\ 0 &= \tilde{\mathbf{g}}_2\end{aligned} \tag{2.8}$$

or, in a slightly rearranged form:

$$\begin{aligned}\frac{d\mathbf{T}}{dt} &= \sum_{i=0}^{1}\frac{1}{\varepsilon_i}\hat{\mathbf{F}}_i\hat{\mathbf{g}}_i\\ 0 &= \tilde{\mathbf{g}}_2\end{aligned} \tag{2.9}$$

where $\hat{\mathbf{F}}_i$ and $\hat{\mathbf{g}}_i$ represent the correspondingly adjusted selector matrices and vectors, respectively, given as:

$$\hat{\mathbf{F}}_0 = \begin{bmatrix} 1 & & \mathbf{0}_{1\times5} \\ \mathbf{0}_{3\times1} & \mathbf{1}_{3\times1} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{4\times2} & & \mathbf{I}_{4\times4} \end{bmatrix}$$

$$\hat{\mathbf{g}}_0 = [F_{tf}(T_{tf} - T_{\text{ref}})]_s \begin{bmatrix} \frac{u_{tf} + k_{hf}u_{hf} + k_{tr}u_{tr}}{V_m} \\ \frac{k_{qf}u_{qf} - k_{hr}u_{hr} - \frac{V_h}{V_m}(k_{hf}u_{hf} + k_{tr}u_{tr} + u_{tf})}{V_c + V_f + V_r + \frac{8F\rho C_p}{UA}V_h} \\ \frac{\frac{V_c+V_f+V_r}{V_m}(k_{hf}u_{hf} + k_{tr}u_{tr} + u_{tf}) + \frac{8F\rho C_p}{UA}(k_{qf}u_{qf} - k_{hr}u_{hr})}{V_c + V_f + V_r + \frac{8F\rho C_p}{UA}V_h} \\ -\frac{k_{qcl}u_{qcl}}{V_{cl}} \\ \frac{-k_P u_P - k_l u_l}{V_{sp}} \\ \frac{k_w u_w}{V_{cp}} \end{bmatrix}$$

$$\hat{\mathbf{F}}_1 = \begin{bmatrix} 1 & & \mathbf{0}_{1\times 5} \\ \mathbf{0}_{3\times 1} & \mathbf{1}_{3\times 1} & \mathbf{0}_{3\times 4} \\ \mathbf{0}_{4\times 2} & & \mathbf{I}_{4\times 4} \end{bmatrix}$$

$$\hat{\mathbf{g}}_1 = [F_{tf}(T_{tf} - T_{ref})]_s \begin{bmatrix} \frac{u_{cp} - k_m u_m}{Vm} \\ \frac{k_m u_m - k_h u_h + \frac{V_h}{V_m}(k_m u_m - u_{cp})}{V_c + V_f + V_r + \frac{8F\rho C_p}{UA}V_h} \\ \frac{\frac{V_c+V_f+V_r}{V_m}(u_{cp} - k_m u_m) + \frac{8F\rho C_p}{UA}(k_m u_m - k_h u_h)}{V_c + V_f + V_r + \frac{8F\rho C_p}{UA}V_h} \\ \frac{k_h u_h - k_{cl}u_{cl}}{V_{cl}} \\ \frac{k_{cl}u_{cl} - k_{sp}u_{sp}}{V_{sp}} \\ \frac{k_{sp}u_{sp} - u_{cp}}{V_{cp}} \end{bmatrix}$$

where **1** is a matrix with all entries equal to 1.

Note that the dynamic equations after the fast boundary layer (i.e. Eq. (2.9)) are also in a singularly perturbed form, indicating that further model reduction can be performed.

Let us define an intermediate time scale $\tau_1 = t/\varepsilon_1$. Substituting this into Eq. (2.9) and taking the limit $\varepsilon_1 \to 0$, we obtain the description of the intermediate energy dynamics:

$$\begin{aligned} \frac{d\mathbf{T}}{d\tau_1} &= \hat{\mathbf{F}}_1 \hat{\mathbf{g}}_1 \\ 0 &= \tilde{\mathbf{g}}_2 \end{aligned} \tag{2.10}$$

Note that all the temperatures evolve in this time scale. The quasi-steady state constraints of the intermediate time scale dynamics are identified as $\hat{\mathbf{g}}_1 = 0$, which are linearly dependent. A linearly independent subset $\tilde{\hat{\mathbf{g}}}_1$, defined as $\hat{\mathbf{g}}_1 = \mathbf{B}_1 \tilde{\hat{\mathbf{g}}}_1$, is taken, with:

$$\mathbf{B}_1 = [F_{tf}(T_{tf} - T_{\text{ref}})]_s \times \begin{bmatrix} \frac{1}{V_m} & 0 & 0 & 0 \\ -\frac{\frac{V_h}{V_m}}{V_c + V_f + V_r + \frac{8\rho C_p}{UA} V_h} & \frac{1}{V_c + V_f + V_r + \frac{8\rho C_p}{UA} V_h} & 0 & 0 \\ \frac{\frac{V_c + V_f + V_r}{V_m}}{V_c + V_f + V_r + \frac{8\rho C_p}{UA} V_h} & \frac{\frac{8F\rho C_p}{UA}}{V_c + V_f + V_r + \frac{8\rho C_p}{UA} V_h} & 0 & 0 \\ 0 & 0 & \frac{1}{V_{cl}} & 0 \\ 0 & 0 & 0 & \frac{1}{V_{sp}} \\ -\frac{1}{V_{cp}} & -\frac{1}{V_{cp}} & -\frac{1}{V_{cp}} & -\frac{1}{V_{cp}} \end{bmatrix}$$

$$\tilde{\hat{\mathbf{g}}}_1 = \begin{bmatrix} u_{cp} - k_m u_m \\ k_m u_m - k_h u_h \\ k_h u_h - k_{cl} u_{cl} \\ k_{cl} u_{cl} - k_{sp} u_{sp} \end{bmatrix}$$

We take the same limit $\varepsilon_1 \to 0$ in the original time scale t, and obtain the description of the energy dynamics after the intermediate boundary layer:

$$\begin{aligned} \frac{d\mathbf{T}}{dt} &= \hat{\mathbf{F}}_0 \hat{\mathbf{g}}_0 + \hat{\mathbf{F}}_1 \mathbf{B}_1 \left(\lim_{\varepsilon_1 \to 0} \frac{\tilde{\hat{\mathbf{g}}}_1}{\varepsilon_1} \right) \\ 0 &= \tilde{\mathbf{g}}_2 \\ 0 &= \tilde{\hat{\mathbf{g}}}_1 \end{aligned}$$

We define another set of algebraic variables $\mathbf{z}_1$ to represent the limiting terms, and obtain:

$$\begin{aligned} \frac{d\mathbf{T}}{dt} &= \hat{\mathbf{F}}_0 \hat{\mathbf{g}}_0 + \hat{\mathbf{F}}_1 \mathbf{B}_1 \mathbf{z}_1 \\ 0 &= \tilde{\mathbf{g}}_2 \\ 0 &= \tilde{\hat{\mathbf{g}}}_1 \end{aligned} \tag{2.11}$$

$\mathbf{z}_1$ can be computed by differentiating the constraints in Eq. (2.11):

$$\mathbf{z}_1 = -(\mathcal{L}_{\hat{\mathbf{F}}_1 \mathbf{B}_1} \tilde{\hat{\mathbf{g}}}_1)^{-1} (\mathcal{L}_{\hat{\mathbf{F}}_0 \hat{\mathbf{g}}_0} \tilde{\hat{\mathbf{g}}}_1) \tag{2.12}$$

Using the solution of Eq. (2.12), the description of the slow energy dynamics is obtained:

$$\begin{aligned} \frac{d\mathbf{T}}{dt} &= \hat{\mathbf{F}}_0\hat{\mathbf{g}}_0 - \hat{\mathbf{F}}_1\mathbf{B}_1(\mathcal{L}_{\hat{\mathbf{F}}_1\mathbf{B}_1}\tilde{\hat{\mathbf{g}}}_1)^{-1}(\mathcal{L}_{\hat{\mathbf{F}}_0\hat{\mathbf{g}}_0}\tilde{\hat{\mathbf{g}}}_1) \\ 0 &= \tilde{\mathbf{g}}_2 \\ 0 &= \tilde{\hat{\mathbf{g}}}_1 \end{aligned} \tag{2.13}$$

or in a rearranged form:

$$\begin{aligned} \frac{d\mathbf{T}}{dt} &= \mathbf{F}'_0\mathbf{g}'_0 \\ 0 &= \tilde{\mathbf{g}}_2 \\ 0 &= \tilde{\hat{\mathbf{g}}}_1 \end{aligned} \tag{2.14}$$

where $\mathbf{F}'_0$ and $\mathbf{g}'_0$ are the correspondingly adjusted selector matrix and vector, respectively. The exact forms of $\mathbf{F}'_0$ and $\mathbf{g}'_0$ are not presented here for brevity.

In summary, the energy dynamics of the FEHE, the furnace and the reactor evolve over all three time scales, while those of the mixer, the cooler, the separator and the compressor evolve in the intermediate and slow time scales.

Graph theoretic analysis

Let us now proceed to the graph-theoretic analysis. The HDA process network can be represented as an energy flow graph $\mathcal{G}$, as shown in Figure 2.4. The node list is provided in Table 2.2. The graph reduction framework is applied to the network using the information in Figure 2.4 and Table 2.1 to obtain the following results.

The subgraph corresponding to the fast time scale $\mathcal{H}_2$, with the largest magnitude $\mathcal{O}(1/\varepsilon_2)$ flows, is shown in Figure 2.5a. The units evolving in this time scale are the

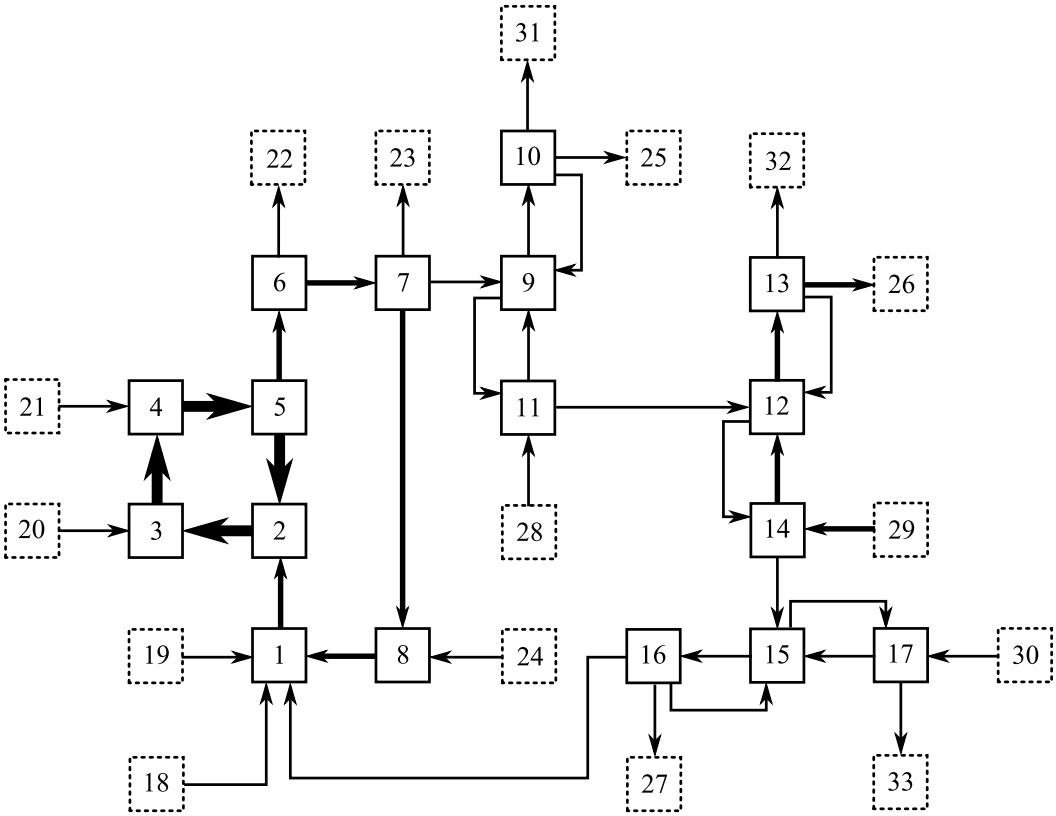


Figure 2.4: Energy flow graph of the HDA process

nodes of $\mathcal{H}_2$, i.e.:

$$\mathcal{T}_2 = \{2, 3, 4, 5\}$$

This subgraph contains a prototype recycle, which can be simplified as a single composite node R_1. Figure 2.5b shows the subgraph for the intermediate time scale $\mathcal{H}_1$. A subset of the units which evolve in this time scale are given by:

$$\mathcal{T}_1 = \{1, R_1, 6, 7, 8, 12, 13, 14\}$$

This subgraph has one prototype recycle which can be clustered to form a composite

Normal nodes		Auxiliary nodes	
Index	Unit	Index	Unit
1	Mixer	18	Toluene feed stream
2	FEHE-Cold	19	H_2 feed stream
3	Furnace	20	Furnace source
4	Plug-flow reactor	21	Heat of reaction
5	FEHE-Hot	22	Cooler sink
6	Cooler	23	Purge stream
7	Separator	24	Compressor power input
8	Compressor	25	S-Condenser sink
9	Stabilizer (S)	26	P-Condenser sink
10	S-Condenser	27	R-Condenser sink
11	S-Reboiler	28	S-Reboiler source
12	Product column (P)	29	P-Reboiler source
13	P-Condenser	30	R-Reboiler source
14	P-Reboiler	31	Methane product stream
15	Recycle column (R)	32	Benzene product stream
16	R-Condenser	33	Diphenyl product stream
17	R-Reboiler		

Table 2.2: Node list of the HDA energy flow graph

node R_2. Also, a prototype throughput is identified, which is simplified as a single composite node T_1, and it is removed from the original energy flow graph $\mathcal{G}$. Lastly, the slow time scale subgraph $\mathcal{H}_0$ is shown in Figure 2.5c. The units evolving in this time scale are given as:

$$\mathcal{T}_0 = \{R_2, 9, 10, 11, 15, 16, 17\}$$

We conclude that, based on the result above, the network is expected to exhibit three-time scale energy dynamics, and the time scales wherein each unit evolves are summarized in Table 2.3. Note that this result matches the outcome of the detailed mathematical analysis.

The full order dynamic model equations are obtained as:

$$\frac{d\mathbf{H}}{dt} = \frac{1}{\varepsilon_0}\mathbf{g}_0 + \frac{1}{\varepsilon_1}\mathbf{g}_1 + \frac{1}{\varepsilon_2}\mathbf{g}_2 \tag{2.15}$$

with

$$\mathbf{g}_0 = h_{18-1,s}\begin{bmatrix} k_{18-1}u_{18-1} + k_{19-1}u_{19-1} + k_{16-1}u_{16-1} \\ 0 \\ k_{20-3}u_{20-3} \\ k_{21-4}u_{21-4} \\ 0 \\ -k_{6-22}u_{6-22} \\ -k_{7-9}u_{7-9} - k_{7-23}u_{7-23} \\ k_{24-8}u_{24-8} \end{bmatrix}$$

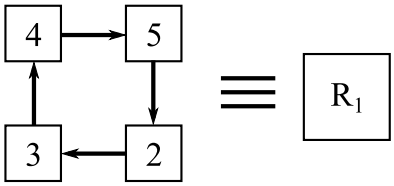
$$\mathbf{g}_1 = h_{18-1,s}\begin{bmatrix} -k_{1-2}u_{1-2} + k_{8-1}u_{8-1} \\ k_{1-2}u_{1-2} \\ 0 \\ 0 \\ -k_{5-6}u_{5-6} \\ k_{5-6}u_{5-6} - k_{6-7}u_{6-7} \\ k_{6-7}u_{6-7} - k_{7-8}u_{7-8} \\ k_{7-8}u_{7-8} - k_{8-1}u_{8-1} \end{bmatrix}$$

Enthalpy	Evolution in each time scale		
	Fast	Intermediate	Slow
H_1	X	O	O
H_2	O	O	O
H_3	O	O	O
H_4	O	O	O
H_5	O	O	O
H_6	X	O	O
H_7	X	O	O
H_8	X	O	O
H_9	X	X	O
H_{10}	X	X	O
H_{11}	X	X	O
H_{12}	X	O	X
H_{13}	X	O	X
H_{14}	X	O	X
H_{15}	X	X	O
H_{16}	X	X	O
H_{17}	X	X	O

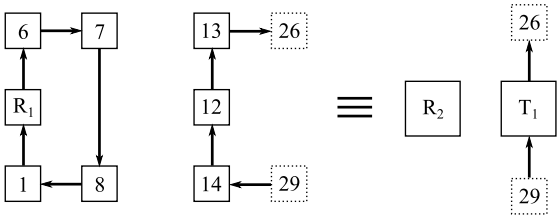
Table 2.3: Summary of time scale analysis for the HDA process

$$\mathbf{g}_2 = h_{18-1,s} \begin{bmatrix} 0 \\ -k_{2-3}u_{2-3} + k_{5-2}u_{5-2} \\ k_{2-3}u_{2-3} - k_{3-4}u_{3-4} \\ k_{3-4}u_{3-4} - k_{4-5}u_{4-5} \\ k_{4-5}u_{4-5} - k_{5-2}u_{5-2} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

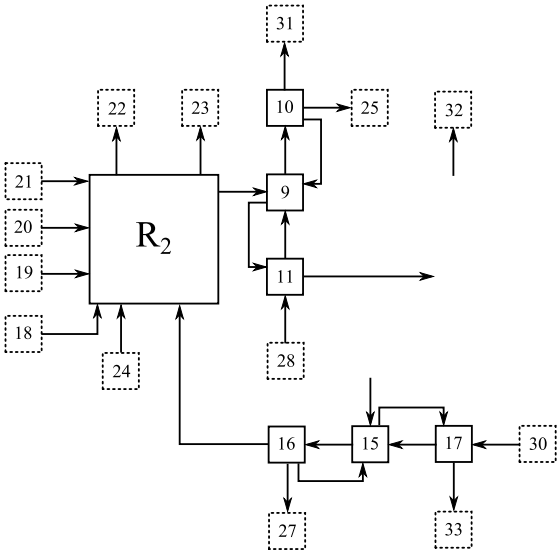
where h_{i-j} represents the energy flow from the i-th node to the j-th node. Note that Eq. (2.15) has the same canonical form as Eq. (2.4).



(a) with the largest energy flows



(b) with the intermediate energy flows



(c) with the smallest energy flows

Figure 2.5: Subgraphs of the HDA reaction part energy flow graph

The dynamics of the reduced order model for the fast time scale is given as:

$$\frac{d\mathbf{H}_2}{d\tau_2} = \mathbf{g}_2' \tag{2.16}$$

with

$$\mathbf{g}_2' = h_{18-1,s} \begin{bmatrix} -k_{2-3}u_{2-3} + k_{5-2}u_{5-2} \\ k_{2-3}u_{2-3} - k_{3-4}u_{3-4} \\ k_{3-4}u_{3-4} - k_{4-5}u_{4-5} \\ k_{4-5}u_{4-5} - k_{5-2}u_{5-2} \end{bmatrix}$$

where $\mathbf{H}_2 = \{H_2, H_3, H_4, H_5\}$. The quasi-steady state constraints, which are not linearly independent, are given as $\mathbf{g}_2' = 0$. Note that, by premultiplying Eq. (2.5) by $\mathbf{F}_2^T$, we can rewrite Eq. (2.5) so that it is in the same canonical form with Eq. (2.16).

The intermediate time scale dynamics is given as:

$$\begin{aligned} \frac{d\mathbf{H}_1}{d\tau_1} &= \mathbf{g}_1 + \mathbf{C}_2\mathbf{B}_2\bar{\mathbf{z}}_2 \\ 0 &= \tilde{\mathbf{g}}_2' \end{aligned} \tag{2.17}$$

with

$$\mathbf{B}_2 = \begin{bmatrix} \mathbf{I}_{3\times 3} \\ -\mathbf{1}_{1\times 3} \end{bmatrix}$$

$$\mathbf{C}_2 = \begin{bmatrix} \mathbf{0}_{1\times 4} \\ \mathbf{I}_{4\times 4} \\ \mathbf{0}_{3\times 3} \end{bmatrix}$$

where $\mathbf{H}_1 = \{H_1, H_2, \ldots, H_8\}$. $\tilde{\mathbf{g}}_2'$, which represents a linearly independent subset of

the quasi-steady state constraints, is given as $\mathbf{g}_2' = \mathbf{B}_2\tilde{\mathbf{g}}_2'$. $\bar{\mathbf{z}}_2$ is a vector of algebraic variables, given as $\lim_{\varepsilon_2 \to 0} \tilde{\mathbf{g}}_2'/\varepsilon_2$. The linearly dependent quasi-steady state constraints of the intermediate energy dynamics, $\mathbf{g}_1' = \mathbf{g}_1 + \mathbf{C}_2\mathbf{B}_2\bar{\mathbf{z}}_2 = 0$, are also given by the algorithm. It can be shown that Eq. (2.17) has the same canonical form with Eq. (2.6).

Finally, the slow time scale dynamic model is given as:

$$\begin{aligned} \frac{d\mathbf{H}}{dt} &= \mathbf{g}_0 + \mathbf{C}_1\mathbf{B}_1\bar{\mathbf{z}}_1 \\ 0 &= \tilde{\mathbf{g}}_2' \\ 0 &= \tilde{\mathbf{g}}_1' \end{aligned} \tag{2.18}$$

with

$$\mathbf{B}_1 = \begin{bmatrix} \mathbf{I}_{7\times 7} \\ -\mathbf{1}_{1\times 7} \end{bmatrix}$$

$$\mathbf{C}_1 = \mathbf{I}_{8\times 8}$$

The linearly independent subset of the quasi-steady state constraints $\tilde{\mathbf{g}}_1'$ can be defined similarly as is done in the intermediate time scale dynamics, and $\bar{\mathbf{z}}_1$ are algebraic variables, defined by $\lim_{\varepsilon_1 \to 0} \tilde{\mathbf{g}}_1'/\varepsilon_1$. Note that Eq. (2.18) has the same canonical form with Eq. (2.11).

Finally, the controlled outputs and the potential manipulated inputs available in each time scale are obtained using the algorithm. In the fast time scale (τ_2), the enthalpies of the FEHE, the furnace, and the reactor need to be controlled, i.e.

$$\mathcal{Y}(\tau_2) = \{2, 3, 4, 5\}$$

and all but one out of four units can be controlled simultaneously, noting that $\mathcal{H}_2$

contains no auxiliary node. Potential manipulated inputs in this time scale include

$$\mathcal{U}(\tau_2) = \{2-3, 3-4, 4-5, 5-2\}$$

The enthalpies of the mixer, the cooler, the separator and the compressor as well as the total enthalpy of the FEHE-reactor recycle loop need to be controlled in the intermediate time scale, i.e.:

$$\mathcal{Y}(\tau_1) = \{1, R_1, 6, 7, 8\}$$

and all but one out of five can be controlled independently since, in $\mathcal{H}_1$, R_2 does not have any auxiliary node connected to it. The following set of potential manipulated inputs is given:

$$\mathcal{U}(\tau_1) = \{1-2, 6-7, 7-8, 8-1, 5-6\}$$

Lastly, the total network enthalpy needs to be controlled in the slow time scale, and this can be achieved using any small external flow.

Simulation study

To further validate the results obtained using the graph-theoretic framework, we now perform numerical simulations using the equations in (2.2), along with the equations of the material dynamics. For simplicity, we assume that there is no phase change in the FEHE, and the holdups of all units are constant. Also, we assume constant heat capacity and density. Reaction rates are computed using the expressions provided in [79]. The nominal values of the state variables and the process parameters are given in Table 2.4.

We considered the open-loop responses of the system by increasing the temperature

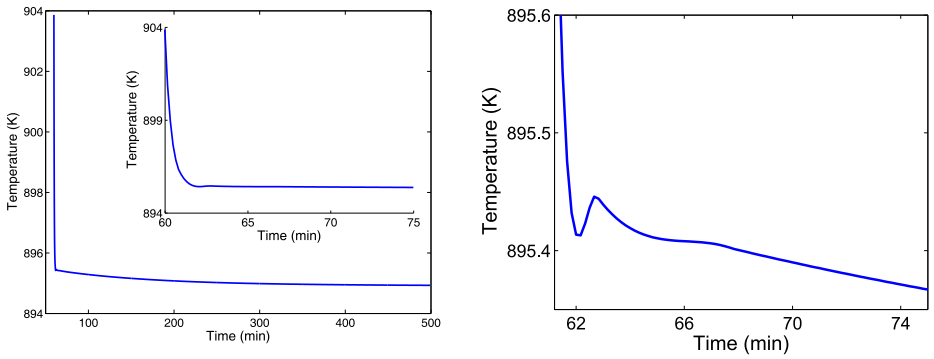
Parameter	Value	Parameter	Value
F_{TF}	0.0033 m^3/s	F_{HF}	0.0356 m^3/s
F_{TR}	9.7×10^{-4} m^3/s	F_q	0.0034 m^3/s
R	0.3087 m^3/s	UA	8.24×10^5 W/K
V_m	0.1 m^3	V_C	14.16 m^3
V_f	8.5 m^3	V_R	110 m^3
V_H	14.16 m^3	V_{cl}	8.5 m^3
V_s	8.5 m^3	V_{cp}	8.5 m^3
T_{TF}	303 K	T_{HF}	303 K
T_{TR}	457 K	$T_{m,s}$	343.91 K
$T_{C,s}$	881.11 K	$T_{f,s}$	896.03 K
$T_{R,s}$	940.60 K	$T_{q,s}$	895.99 K
$T_{H,s}$	384.45 K	$T_{cl,s}$	317.55 K
$T_{s,s}$	317.55 K	$T_{cp,s}$	348.71 K

Table 2.4: Nominal values of the state variables and the process parameters

of the furnace by 1%. Figure 2.6 shows the evolution of the furnace temperature. We note that the furnace temperature quickly drops and then slowly goes back to the steady state. Also, an intermediate time scale dynamics is observed as shown in Figure 2.6b. The separator temperature increases in a slower time scale compared to the furnace, and then returns slowly to the steady state, as shown in Figure 2.7, indicating that it evolves in the intermediate and the slow time scales. Note that the simulation results are consistent with the results obtained using either the detailed mathematical analysis or the graph-theoretic framework.

Let us now proceed to the simulations of the closed-loop responses. Main control objectives for the reaction part of this network are:

- to reject disturbances going into the reactor-FEHE recycle loop in order to prevent the accumulation of energy. It can be done by controlling the exit temperature of the recycle loop, i.e. the temperature of the hot channel of the FEHE, which corresponds to T_5 in the energy flow graph. Note that this objective needs to be addressed in the time scale τ_2, since T_5 is included in $\mathcal{Y}(\tau_2)$. The edge 5-2, i.e.



(a) Overall (b) Intermediate

Figure 2.6: Evolution of the furnace temperature

the duty of the FEHE, can be used as the manipulated input. We introduce the bypass across the hot channel of the FEHE to manipulate the duty of the FEHE indirectly.

- to control the separator temperature to indirectly control the liquid phase composition. Since there is no external energy source/sink for the separator, the inlet temperature of the separator, which corresponds to T_6 in the energy flow graph, needs to be controlled. Note that this objective needs to be addressed in the time scale τ_1, since T_6 is included in $\mathcal{Y}(\tau_1)$. The edge 6-22, i.e. the duty of the cooler, can be used as the manipulated input.

- to control the reactor inlet temperature to facilitate the reactions. This objective can be addressed in the original time scale t using the duty of the furnace as the manipulated input.

- to control the reactor outlet temperature to prevent cocking. In order to achieve it, we introduce a quench stream which is a part of the liquid outlet stream of the separator. We assume that the quench stream is mixed with the reactor outlet

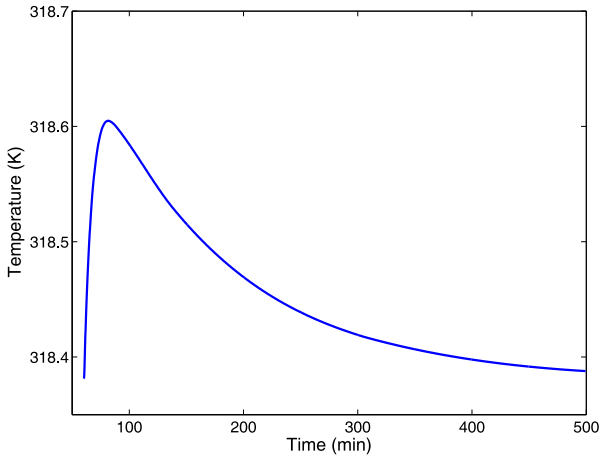


Figure 2.7: Evolution of the separator temperature

stream instantaneously.

The process flow diagram with the above control loops is shown in Figure 2.8. We implement simple proportional and integral (PI) controllers with the following control laws:

$$\text{TC1: } b = b_s + K_{c,1}\left[(T_{h,set} - T_h) + \frac{1}{\tau_{I,1}}\int_0^t (T_{h,set} - T_h)d\tau\right]$$

$$\text{TC2: } Q_{cl} = Q_{cl,s} - K_{c,2}\left[(T_{cl,set} - T_{cl}) + \frac{1}{\tau_{I,2}}\int_0^t (T_{cl,set} - T_{cl})d\tau\right]$$

$$\text{TC3: } Q_f = Q_{f,s} + K_{c,3}\left[(T_{f,set} - T_f) + \frac{1}{\tau_{I,3}}\int_0^t (T_{f,set} - T_f)d\tau\right]$$

$$\text{TC4: } F_q = F_{q,s} - K_{c,4}\left[(T_{q,set} - T_q) + \frac{1}{\tau_{I,4}}\int_0^t (T_{q,set} - T_q)d\tau\right]$$

where b is the bypass ratio around the hot channel of the FEHE, F_q is the volumetric flow rate of the quench stream, and T_q is the reactor outlet temperature after quenching. Different parameters for the controllers are provided in Table 2.5.

We first performed a set-point tracking simulation. At $t = 60$ min, we increased the

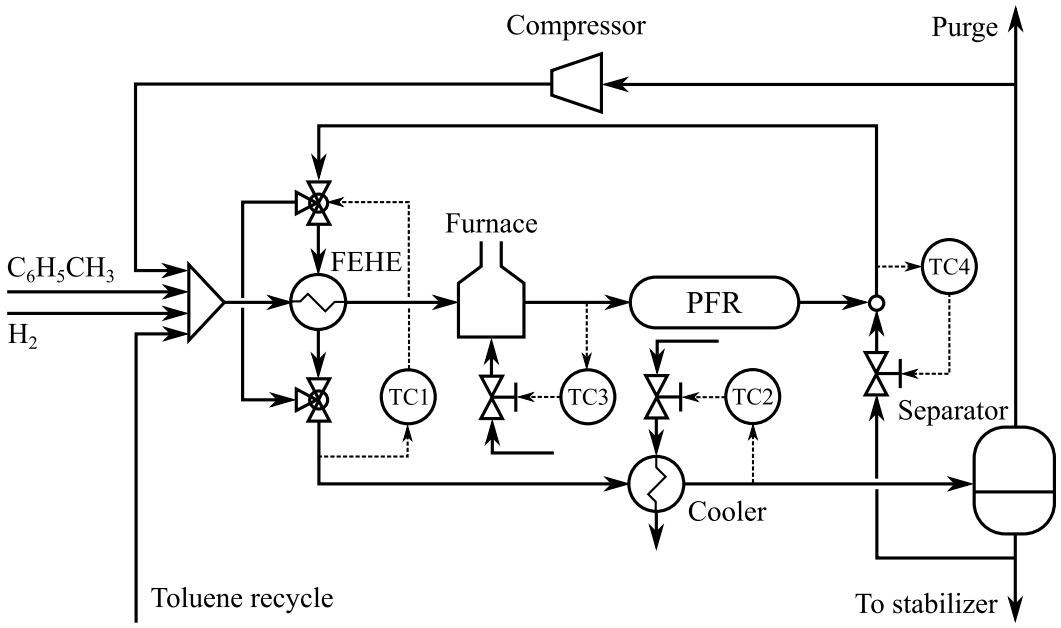
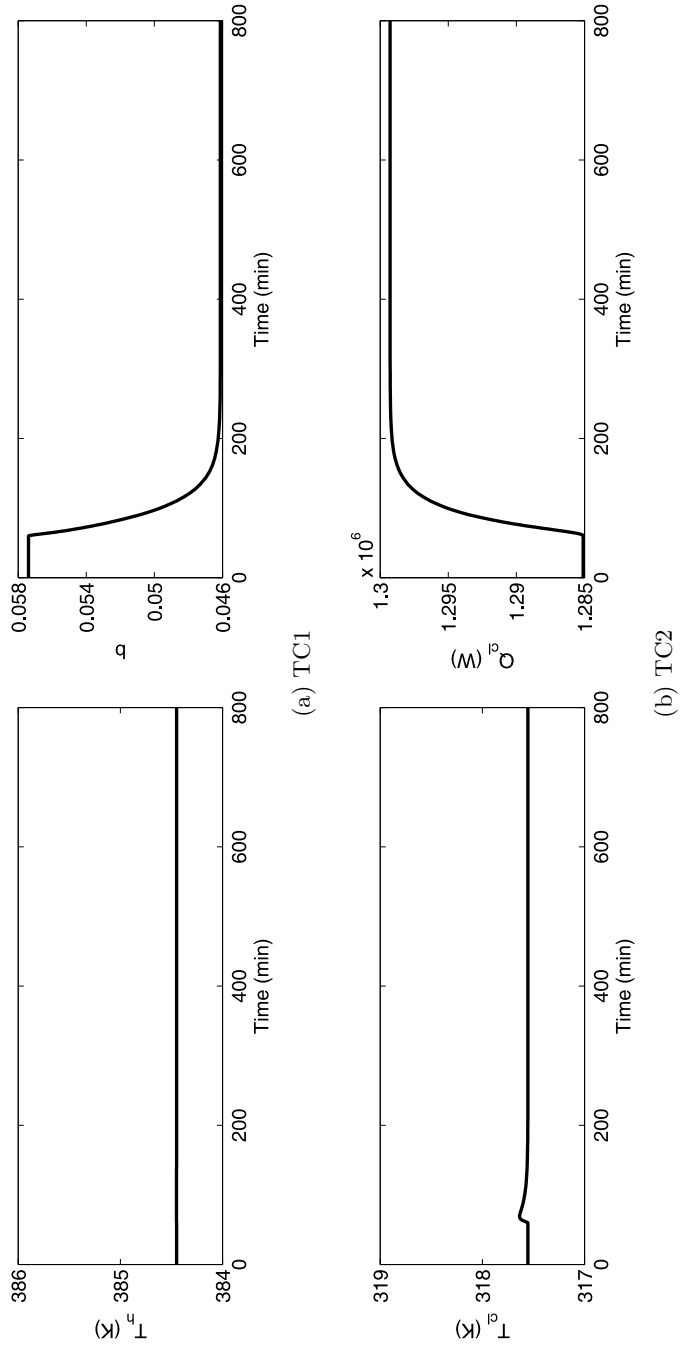


Figure 2.8: Multi-loop control configuration for the HDA process

Parameter	Value	Parameter	Value
b_s	0.0574	$Q_{f,s}$	2.729×10^5 W
$K_{c,1}$	0.03/K	$K_{c,3}$	600 W/K
$\tau_{I,1}$	10 s	$\tau_{I,3}$	240 s
$Q_{cl,s}$	1.285×10^6 W	$F_{q,s}$	3.403×10^{-3} m^3/s
$K_{c,2}$	200 W/K	$K_{c,4}$	4.5×10^{-6} m^3/s/K
$\tau_{I,2}$	50 s	$\tau_{I,4}$	2 s

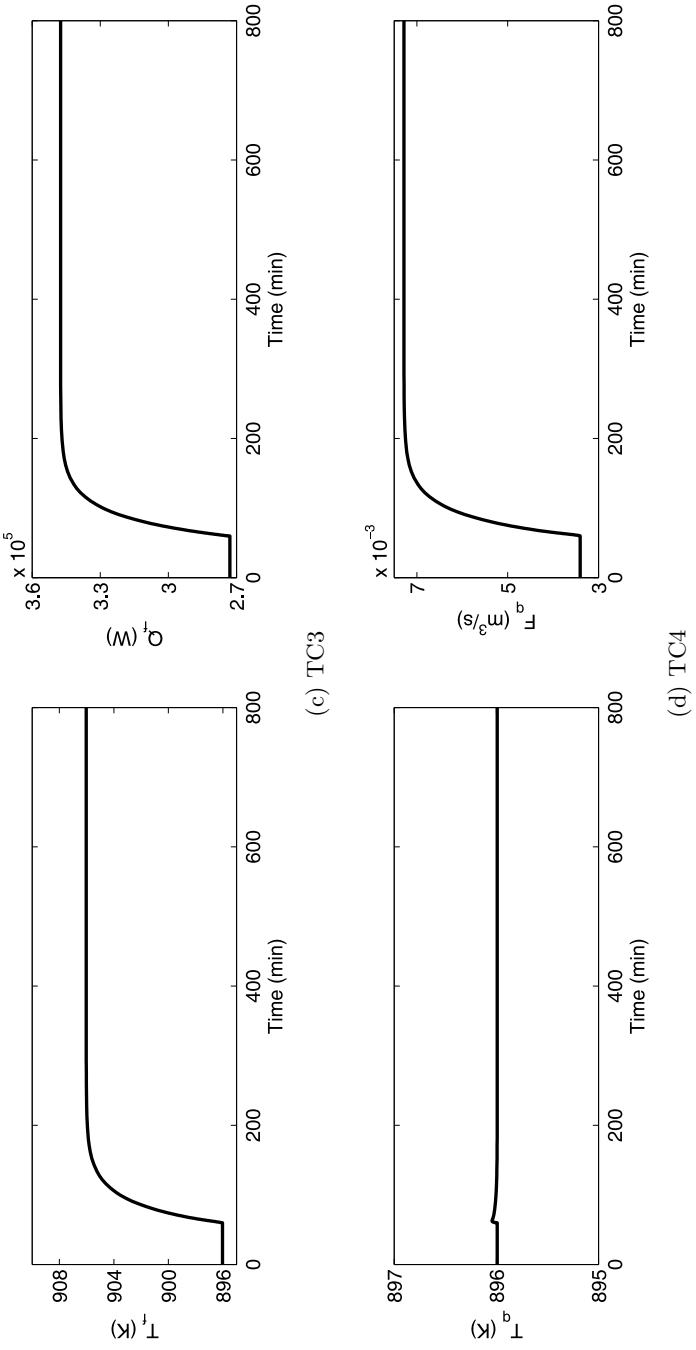
Table 2.5: Controller parameters for the HDA process

set point of T_f by 10 K. The closed-loop responses are shown in Figure 2.8. We note that the different control objectives are achieved in different time scales. For example, T_f is controlled in a slower time scale compared to T_h. This result is consistent with the control hierarchy discussed above. In the second simulation, we introduced a disturbance into the network by increasing the temperature of the hydrogen feed stream by 20 K. The closed-loop responses are shown in Figure 2.8. Note that the proposed control structure rejects the disturbance effectively, showing small changes in all the temperatures.


b
0.058
0.054
0.05
0.046
0
200
400
600
800
Time (min)
T_h (K)
386
385
384
Q_c (W)
x 10^6
1.3
1.295
1.29
1.285
T_c (K)
319
318
317

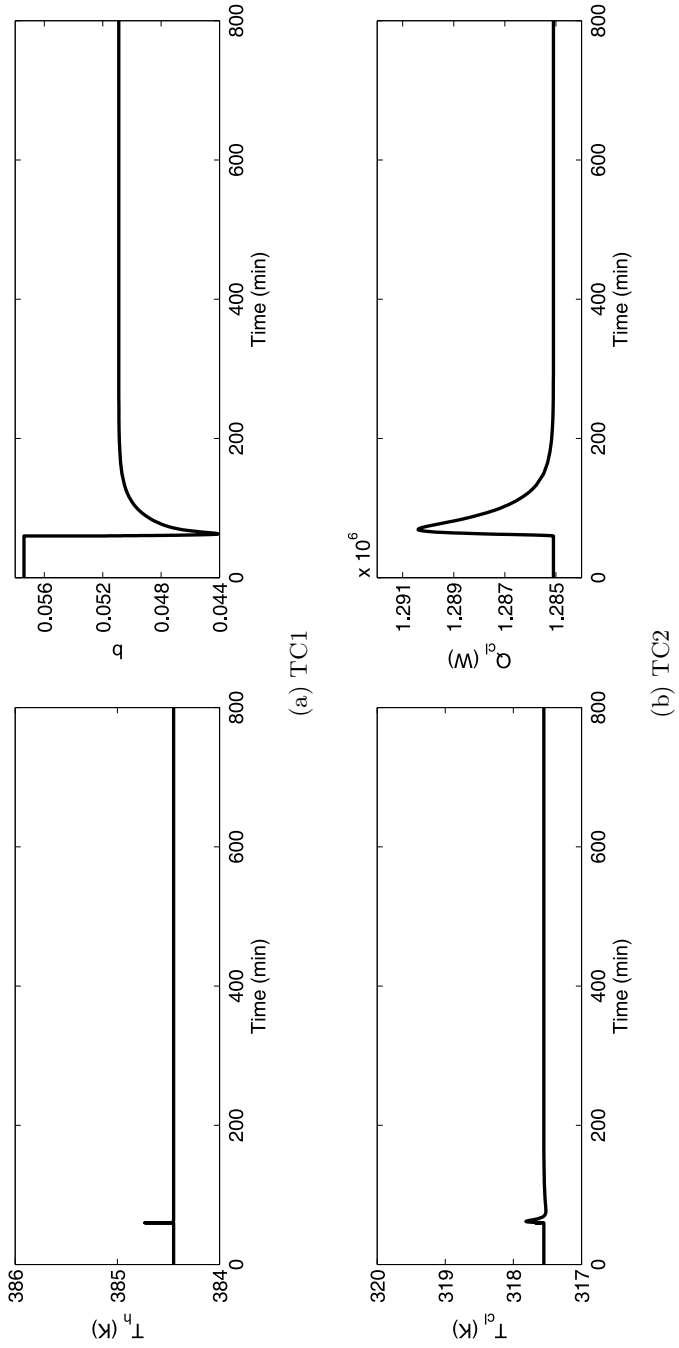
(a) TC1

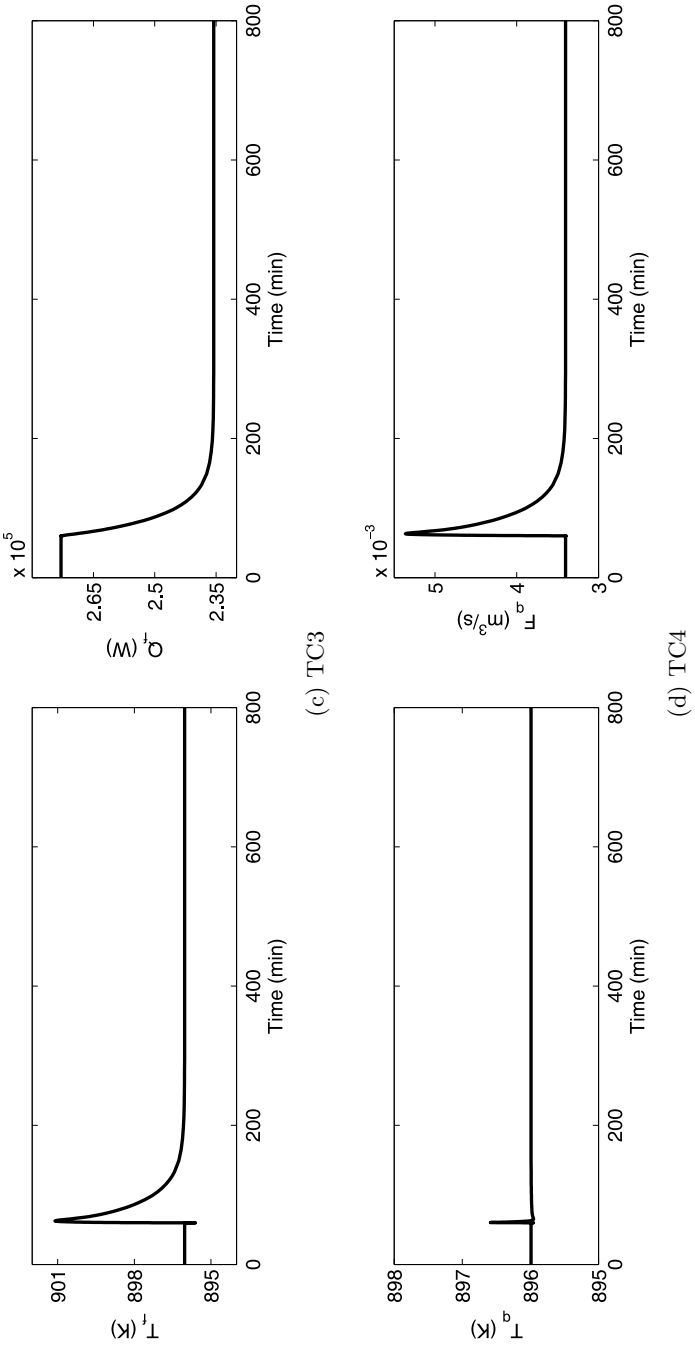
(b) TC2



(c) TC3 (d) TC4

Figure 2.8: Closed-loop responses for set-point tracking


b
0.056
0.052
0.048
0.044
0
200
400
600
800
Time (min)
T_h (K)
386
385
384
(a) TC1
Q_{cl} (W)
x 10^6
1.291
1.289
1.287
1.285
T_{cl} (K)
320
319
318
317
(b) TC2



(c) TC3 (d) TC4

Figure 2.8: Closed-loop responses for disturbance rejection

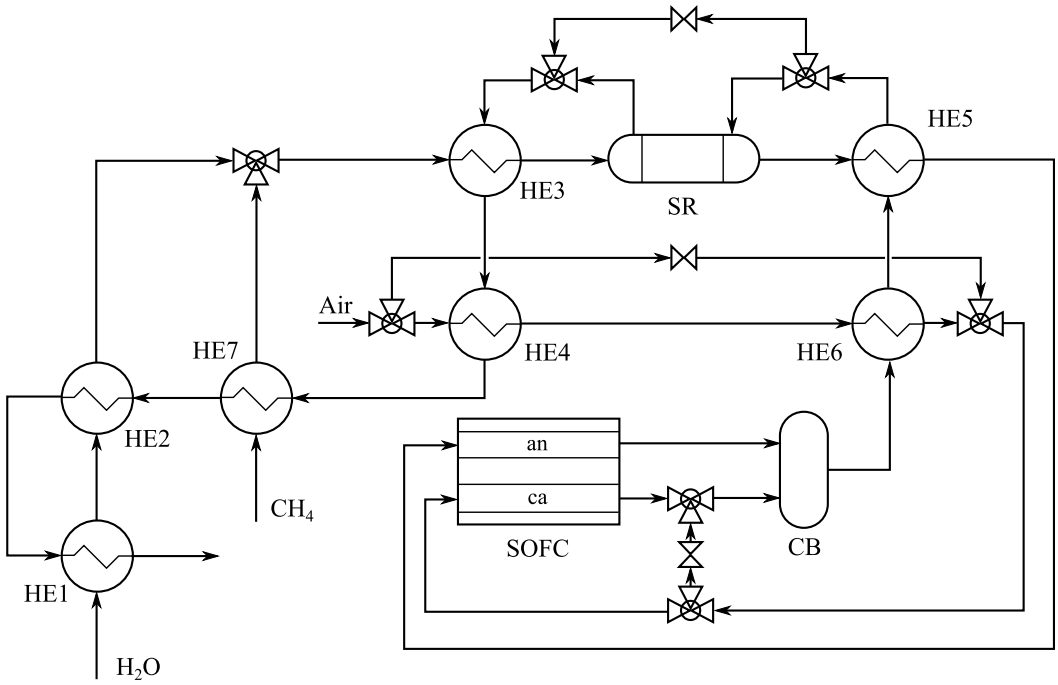


Figure 2.9: Energy integrated SOFC system

2.3.2 Energy integrated solid oxide fuel cell system

Let us now consider an energy integrated solid oxide fuel cell (SOFC) system with an external reformer [80], as shown in Figure 2.9. In this network, both the outlet streams of the SOFC are fed to a catalytic burner to achieve complete combustion of methane, carbon monoxide and unreacted hydrogen. Then, the outlet stream of the burner, which is at high temperature, provides energy to the reformer, where endothermic reactions occur, and the feed streams, resulting in a tightly integrated configuration. The energy flow graph of the network is shown in Figure 2.10, and all the nodes are listed in Table 2.6. The various energy flows of the network span a wide range of values from 0.256 kW to 49.098 kW. The orders of magnitude of the energy flows are determined following a similar procedure used in the previous example, as summarized in Table 2.7. ε_1 and ε_2 are small parameters ($\varepsilon_2 \ll \varepsilon_1 \ll 1$), which are the ratios of the energy input through

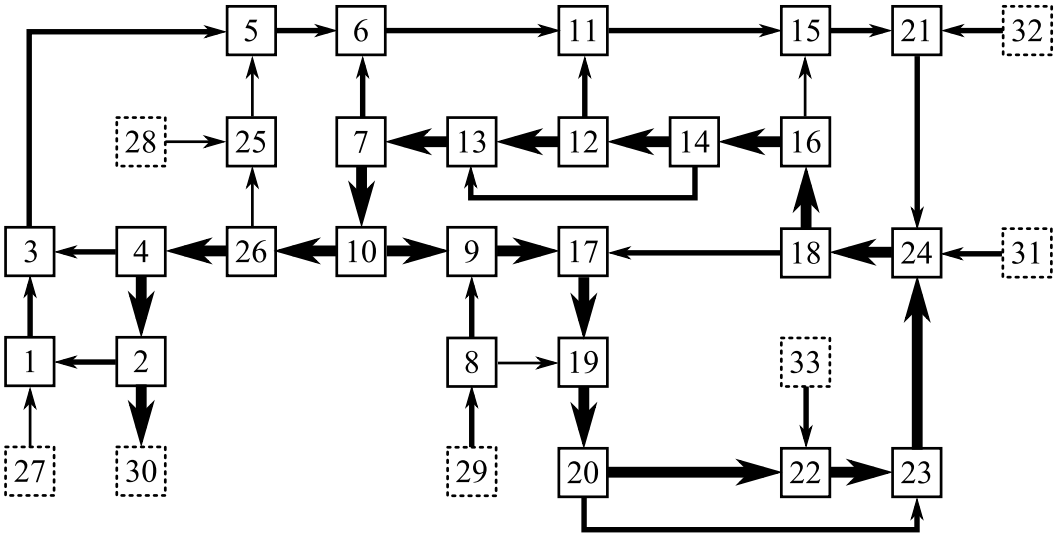


Figure 2.10: Energy flow graph of the energy integrated SOFC system

the methane feed to the energy input through the air feed and the energy of the catalytic burner outlet stream, respectively.

We apply the graph-theoretic reduction to the network using the information in Figure 2.10 and Table 2.7 as inputs. The graph-theoretic algorithm checks the *feasibility* of the network, based on the energy balance around all the normal nodes as well as the composite nodes. Specifically, the highest order of magnitude among the edges entering the node N should match with the highest order of magnitude among the edges leaving the node N in order to have a feasible energy flow structure around the unit N. In this case, the energy flow structure around the recycle, consisting of nodes 24-18-16-16-12-13-7-10-9-17-19-20-22-23, is infeasible since the recycle has an energy output of $\mathcal{O}(1/\varepsilon_2)$, but no energy input of $\mathcal{O}(1/\varepsilon_2)$. This infeasibility may originate from either inappropriately determined orders of magnitude or no clear segregation of energy flows. In order to determine which is the case, we remove the energy feasibility check, and apply the algorithm again.

Figure 2.11a shows the subgraph corresponding to the fast time scale. The units

Normal nodes			
Index	Unit	Index	Unit
1	HE1-Cold	14	Splitter 2
2	HE1-Hot	15	HE5-Cold
3	HE2-Cold	16	HE5-Hot
4	HE2-Hot	17	HE6-Cold
5	Mixer 1	18	HE6-Hot
6	HE3-Cold	19	Mixer 3
7	HE3-Hot	20	Splitter 3
8	Splitter 1	21	SOFC anode
9	HE4-Cold	22	SOFC cathode
10	HE4-Hot	23	Mixer 4
11	Steam reformer (SR)	24	Catalytic burner (CB)
12	SR jacket	25	HE7-Cold
13	Mixer 2	26	HE7-Hot
Auxiliary nodes			
Index	Unit	Index	Unit
27	Water feed stream	31	CB energy input
28	Methane feed stream	32	SOFC anode heat of reaction
29	Air feed stream	33	SOFC cathode heat of reaction
30	HE1 sink		

Table 2.6: Node list of the energy integrated SOFC system

evolving in this time scale are:

$$\mathcal{T}_2 = \{2, 4, 7, 9, 10, 12, 13, 14, 16, 17, 18, 19, 20, 22, 23, 24, 26\}$$

A prototype recycle is identified, and can be simplified as a single composite node R_1. Also, a prototype throughput is identified and clustered to form a composite node T_1. The composite node T_1 is then removed from the original energy flow graph. The subgraph of the intermediate time scale is shown in Figure 2.11b. A set of units evolving in this time scale is identified as

$$\mathcal{T}_1 = \{1, 3, 5, 6, 8, 11, 15, 21\}$$

Lastly, the slow time scale subgraph is shown in Figure 2.11c. The following unit evolves in this time scale: $\mathcal{T}_0 = \{25\}$.

We note that the composite node R_1 has one energy output of $\mathcal{O}(1/\varepsilon_2)$ (i.e. 10-26), and several energy inputs of $\mathcal{O}(1/\varepsilon_1)$ (e.g. 33-22, 21-24). Although none of these energy inputs is of $\mathcal{O}(1/\varepsilon_2)$, the sum of the energy content of the inputs is comparable to the energy content of the flow 10-26, implying that there is no clear segregation of energy flows in the network.

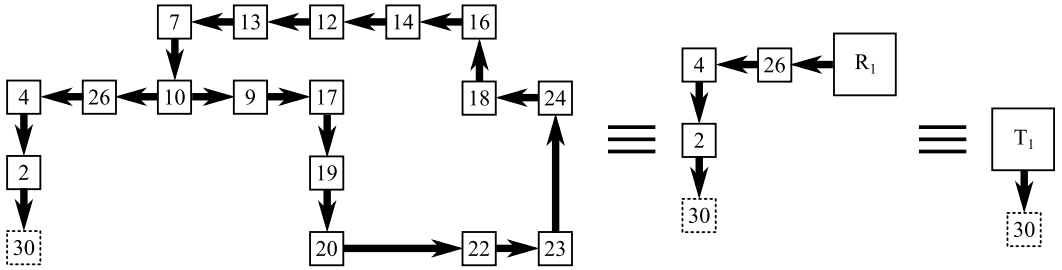
We conclude that, despite the tight energy integration throughout the network, there is no process unit with multi-time scale energy dynamics, as summarized in Table 2.8. The outlet stream of the burner, which has the largest amount of energy among all the process flows, provides energy to other streams through seven heat exchangers and the steam reformer, forming a *pseudo* throughput in the sense that the throughput starts at a process unit, not an energy source. The energy dynamics of the network is dominated by this large energy throughput, hence the absence of time-scale segregation in the energy dynamics.

2.4 Energy integrated distillation columns

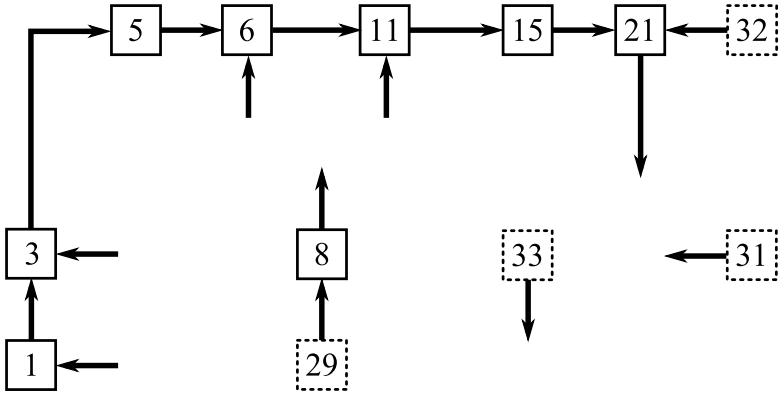
Energy integration in distillation column networks has been an active area of research, as distillation is one of the most energy intensive processes in chemical plants, and has resulted in various thermally coupled configurations [81]. It has been shown that the energy flow structures of such configurations are combinations of energy recycles and energy throughputs, showing a potential for multi-time scale dynamics [43]. However, the time scale properties of such configurations are not clear a priori. In what follows, we apply the graph-theoretic analysis to two different examples of such networks.

Energy flow	mC_pT(kW)	Order of magnitude
28→25	0.256	$\mathcal{O}(1)$
25→5	0.447	$\mathcal{O}(1)$
27→1	1.049	$\mathcal{O}(1/\varepsilon_1)$
1→3	1.313	$\mathcal{O}(1/\varepsilon_1)$
3→5	1.815	$\mathcal{O}(1/\varepsilon_1)$
5→6	2.450	$\mathcal{O}(1/\varepsilon_1)$
6→11	3.901	$\mathcal{O}(1/\varepsilon_1)$
11→15	4.997	$\mathcal{O}(1/\varepsilon_1)$
15→21	5.215	$\mathcal{O}(1/\varepsilon_1)$
21→24	6.231	$\mathcal{O}(1/\varepsilon_1)$
29→8	8.768	$\mathcal{O}(1/\varepsilon_1)$
8→9	7.891	$\mathcal{O}(1/\varepsilon_1)$
9→17	19.303	$\mathcal{O}(1/\varepsilon_2)$
17→19	27.751	$\mathcal{O}(1/\varepsilon_2)$
19→20	28.627	$\mathcal{O}(1/\varepsilon_2)$
20→22	25.764	$\mathcal{O}(1/\varepsilon_2)$
22→23	30.768	$\mathcal{O}(1/\varepsilon_2)$
24→18	49.098	$\mathcal{O}(1/\varepsilon_2)$
18→16	39.934	$\mathcal{O}(1/\varepsilon_2)$
16→14	39.722	$\mathcal{O}(1/\varepsilon_2)$
13→7	35.217	$\mathcal{O}(1/\varepsilon_2)$
7→10	33.821	$\mathcal{O}(1/\varepsilon_2)$
10→26	21.441	$\mathcal{O}(1/\varepsilon_2)$
26→4	21.163	$\mathcal{O}(1/\varepsilon_2)$
4→2	17.195	$\mathcal{O}(1/\varepsilon_2)$
2→30	16.710	$\mathcal{O}(1/\varepsilon_2)$

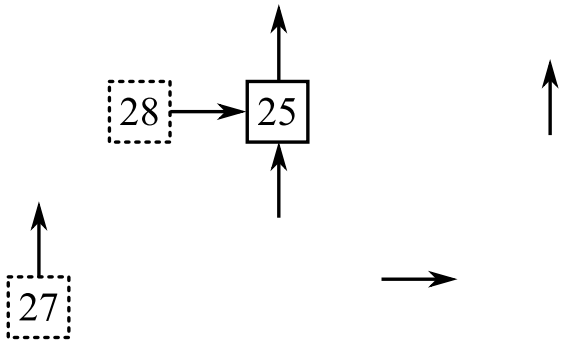
Table 2.7: Summary of orders of magnitude of the energy flows of the energy integrated SOFC system



(a) with the largest energy flows



(b) with the intermediate energy flows



(c) with the smallest energy flows

Figure 2.11: Subgraphs of the energy integrated SOFC energy flow graph

Enthalpy	Evolution in each time scale		
	Fast	Intermediate	Slow
H_1	X	O	X
H_2	O	X	X
H_3	X	O	X
H_4	O	X	X
H_5	X	O	X
H_6	X	O	X
H_7	O	X	X
H_8	X	O	X
H_9	O	X	X
H_{10}	O	X	X
H_{11}	X	O	X
H_{12}	O	X	X
H_{13}	O	X	X
H_{14}	O	X	X
H_{15}	X	O	X
H_{16}	O	X	X
H_{17}	O	X	X
H_{18}	O	X	X
H_{19}	O	X	X
H_{20}	O	X	X
H_{21}	X	O	X
H_{22}	O	X	X
H_{23}	O	X	X
H_{24}	O	X	X
H_{25}	X	X	O
H_{26}	O	X	X

Table 2.8: Summary of time scale analysis for the energy integrated SOFC process

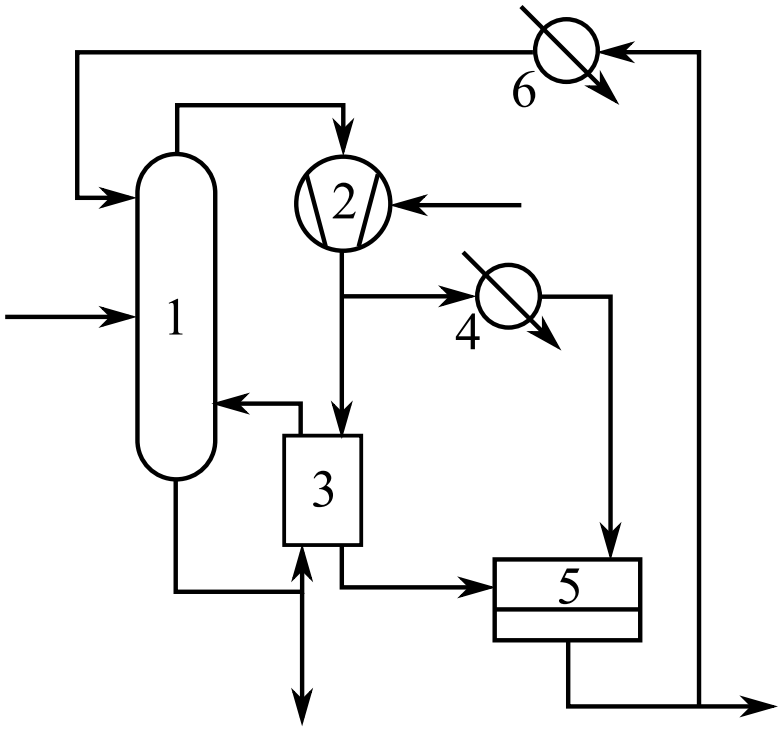


Figure 2.12: Direct vapor recompression distillation (1: Distillation column, 2: Compressor, 3: Combined reboiler-condenser, 4: Trim condenser, 5: Reflux drum, 6: Auxiliary cooler)

2.4.1 Vapor recompression distillation

Vapor recompression distillation (VRD) represents a typical example of energy integrated distillation column networks, where the condenser and the reboiler of the same column are thermally coupled. Let us consider the so-called *direct* VRD configuration, where the recompressed vapor stream is directly used to heat the bottom liquid stream, as shown in Figure 2.12. The top vapor stream of the distillation column is compressed in the compressor to facilitate the heat transfer to the bottoms stream. Then, the majority of the compressed vapor goes through the combined reboiler-condenser to vaporize the bottom liquid stream. The residual vapor is condensed using a trim condenser. An auxiliary cooler is used to regulate the temperature of the reflux. The energy flow

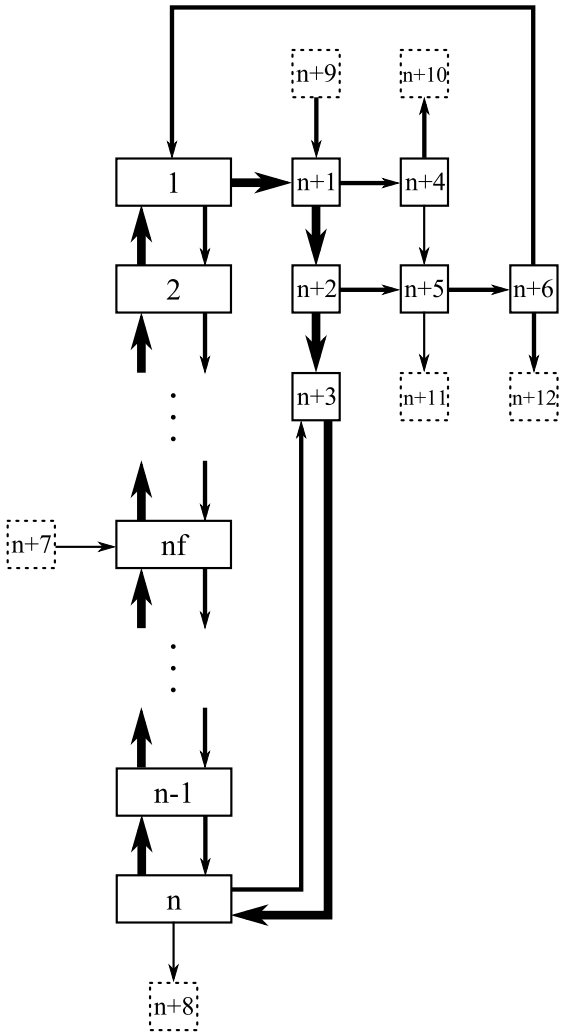


Figure 2.13: Energy flow graph of direct VRD

Normal nodes		Auxiliary nodes	
Index	Unit	Index	Unit
1∼n	Distillation column	n+7	Column feed stream
n+1	Compressor	n+8	Bottoms stream
n+2	RC-Condenser	n+9	Compressor power input
n+3	RC-Reboiler	n+10	Trim condenser sink
n+4	Trim condenser	n+11	Distillate
n+5	Reflux drum	n+12	Auxiliary cooler sink
n+6	Auxiliary cooler		

Table 2.9: Node list of VRD energy flow graph

structure of this network can be transformed into an energy flow graph $\mathcal{G}$ as shown in Figure 2.13. Note that the distillation column is represented by n separate nodes each of which corresponds to each tray of the distillation column with nf being the feed tray. All the nodes of the energy flow graph are listed in Table 2.9. The following aspects of this process lead to the segregation in the energy flows:

- The molar flow rates of the distillate and the bottoms stream are much smaller compared to those of the reflux and the top vapor stream (i.e. there is segregation in the material flows), which results in a difference in the enthalpy content of these streams.

- The partial molar enthalpy of vapor streams is much larger than the partial molar enthalpy of liquid streams, i.e. the latent heat recovered through the combined reboiler-condenser is much larger than the sensible heat of any other energy flow.

In the previous study [82], a detailed mathematical reduction using singular perturbations was performed, and three-time scale dynamics was documented, with the energy dynamics evolving in the fast and the intermediate time scales and the material dynamics evolving in the intermediate and the slow time scales. For illustration, we consider a column with n=95 and nf=47, and we use the order of magnitude information provided

Energy flow	Order of magnitude	Energy flow	Order of magnitude
$i{+}1{\to}i$, $1 \leq i \leq 94$	$\mathcal{O}(1/\varepsilon_2)$	$100{\to}101$	$\mathcal{O}(1/\varepsilon_1)$
$1{\to}96$	$\mathcal{O}(1/\varepsilon_2)$	$101{\to}1$	$\mathcal{O}(1/\varepsilon_1)$
$96{\to}97$	$\mathcal{O}(1/\varepsilon_2)$	$104{\to}96$	$\mathcal{O}(1/\varepsilon_1)$
$97{\to}98$	$\mathcal{O}(1/\varepsilon_2)$	$99{\to}105$	$\mathcal{O}(1/\varepsilon_1)$
$98{\to}95$	$\mathcal{O}(1/\varepsilon_2)$	$101{\to}107$	$\mathcal{O}(1/\varepsilon_1)$
$i{\to}i{+}1$, $1 \leq i \leq 94$	$\mathcal{O}(1/\varepsilon_1)$	$99{\to}100$	$\mathcal{O}(1)$
$95{\to}98$	$\mathcal{O}(1/\varepsilon_1)$	$102{\to}48$	$\mathcal{O}(1)$
$96{\to}99$	$\mathcal{O}(1/\varepsilon_1)$	$100{\to}106$	$\mathcal{O}(1)$
$97{\to}100$	$\mathcal{O}(1/\varepsilon_1)$	$95{\to}103$	$\mathcal{O}(1)$

Table 2.10: Summary of orders of magnitude of the energy flows of the VRD

in [82], summarized in Table 2.10.

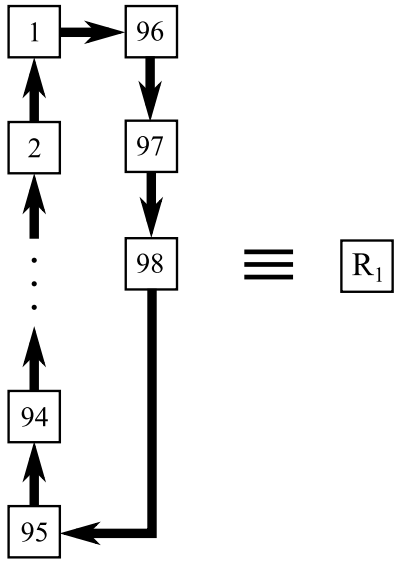
We apply the proposed graph reduction framework to this network, using the information in Figure 2.13 and Table 2.10, focusing on the energy dynamics only. The resulting subgraphs are shown in Figure 2.14. Figure 2.14a shows the subgraph $\mathcal{H}_2$ for the energy flows with the largest magnitude $\mathcal{O}(1/\varepsilon_2)$ (corresponding to the fast time scale). The units evolving in the fast time scale are:

$$\mathcal{T}_2 = \{1, 2, \ldots, 97, 98\}$$

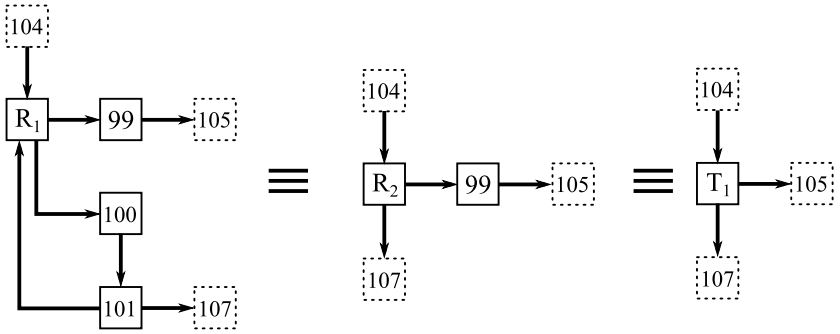
This subgraph contains one prototype recycle, and can be simplified to obtain a single composite recycle node R_1. Figure 2.14b shows the subgraph $\mathcal{H}_1$ for the energy flow with the intermediate magnitude $\mathcal{O}(1/\varepsilon_1)$ (corresponding to the intermediate time scale). The units evolving in the intermediate time scale are:

$$\mathcal{T}_1 = \{R_1, 99, 100, 101\}$$

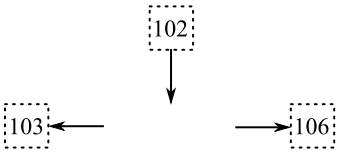
Note that the composite node R_1 is included in the intermediate time scale, implying that the total enthalpy of the recycle (i.e. $H_1 + H_2 + \cdots + H_{97} + H_{98}$) evolves in the



(a) with the largest energy flows



(b) with the intermediate energy flows



(c) with the smallest energy flows

Figure 2.14: Subgraphs of the VRD energy flow graph

intermediate time scale. A prototype recycle is identified, and can be clustered to form a single composite recycle node R_2. Subsequently, a prototype throughput, which can be replaced by a composite throughput node T_1, is identified, and removed from the energy flow graph before we proceed to the slow time scale. Figure 2.14c shows the subgraph $\mathcal{H}_0$ for the energy flow with the smallest magnitude $\mathcal{O}(1)$ (corresponding to the slow time scale). Note that this graph contains only the auxiliary nodes and the small energy flows, implying that no enthalpy evolves in this time scale.

Based on the above result, this network is expected to exhibit two-time scale energy dynamics as summarized in Table 2.11. The canonical forms of the dynamic equations of the original and the reduced models are also obtained. The original model is given as:

$$\frac{d\mathbf{H}}{dt} = \mathbf{g}_0 + \frac{1}{\varepsilon_1}\mathbf{g}_1 + \frac{1}{\varepsilon_2}\mathbf{g}_2 \tag{2.19}$$

with

$$\mathbf{g}_0 = h_{102-48,s} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ k_{102-48}u_{102-48} \\ 0 \\ \vdots \\ 0 \\ -k_{95-103}u_{95-103} \\ 0 \\ 0 \\ 0 \\ -k_{99-100}u_{99-100} \\ k_{99-100}u_{99-100} - k_{100-106}u_{100-106} \\ 0 \end{bmatrix}$$

$$\mathbf{g}_1 = h_{102-48,s} \begin{bmatrix} -k_{1-2}u_{1-2} + k_{101-1}u_{101-1} \\ k_{1-2}u_{1-2} - k_{2-3}u_{2-3} \\ k_{2-3}u_{2-3} - k_{3-4}u_{3-4} \\ \vdots \\ k_{93-94}u_{93-94} - k_{94-95}u_{94-95} \\ k_{94-95}u_{94-95} - k_{95-98}u_{95-98} \\ -k_{96-99}u_{96-99} + k_{104-96}u_{104-96} \\ -k_{97-100}u_{97-100} \\ k_{95-98}u_{95-98} \\ k_{96-99}u_{96-99} - k_{99-105}u_{99-105} \\ k_{97-100}u_{97-100} - k_{100-101}u_{100-101} \\ k_{100-101}u_{100-101} - k_{101-1}u_{101-1} - k_{101-107}u_{101-107} \end{bmatrix}$$

$$\mathbf{g}_2 = h_{102-48,s} \begin{bmatrix} -k1 - 96u_{1-96} + k_{2-1}u_{2-1} \\ -k_{2-1}u_{2-1} + k_{3-2}u_{3-2} \\ -k_{3-2}u_{3-2} + k_{4-3}u_{4-3} \\ \vdots \\ -k_{94-93}u_{94-93} + k_{95-94}u_{95-94} \\ -k_{95-94}u_{95-94} + k_{98-95}u_{98-95} \\ -k_{1-2}u_{1-2} + k_{4-1}u_{4-1} \\ k_{1-96}u_{1-96} - k_{96-97}u_{96-97} \\ k_{96-97}u_{96-97} - k_{97-98}u_{97-98} \\ k_{97-98}u_{97-98} - k_{98-95}u_{98-95} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

k_{i-j} in $\mathbf{g}_0$, $\mathbf{g}_1$ and $\mathbf{g}_2$, which are the $\mathcal{O}(1)$ steady state ratios, are defined as:

$$k_{i-j} = \frac{h_{i-j,s}}{h_{8-1,s}},\ \frac{h_{i-j,s}}{h_{95-98,s}},\ \frac{h_{i-j,s}}{h_{1-96,s}}$$

respectively, whereas u_{i-j} are defined as:

$$u_{i-j} = \frac{h_{i-j}}{h_{i-j,s}}$$

where h_{i-j} represents the energy flow from the i-th unit to the j-th unit, and the subscript s denotes a steady state value. ε_1 and ε_2 are small parameters, which are the ratios of the nominal energy input through the feed stream (i.e. $h_{8-1,s}$) to the

Enthalpy	Evolution in each time scale		
	Fast	Intermediate	Slow
$H_1 \sim H_{95}$	O	O	X
H_{96}	O	O	X
H_{97}	O	O	X
H_{98}	O	O	X
H_{99}	X	O	X
H_{100}	X	O	X
H_{101}	X	O	X

Table 2.11: Summary of time scale analysis for the direct VRD

nominal energy flow from the distillation column to the reboiler (i.e. $h_{95-98,s}$) and to the compressor (i.e. $h_{1-96,s}$), respectively.

The fast time scale dynamics is given as:

$$\frac{d\mathbf{H}_2}{d\tau_2} = \mathbf{F}_2^T \mathbf{g}_2 = \mathbf{g}_2'$$

with

$$\mathbf{F}_2 = \begin{bmatrix} \mathbf{I}_{98\times 98} \\ \mathbf{0}_{3\times 98} \end{bmatrix}$$

where $\mathbf{H}_2 = \{H_1, H_2, \ldots, H_{97}, H_{98}\}$ and $\tau_2 = t/\varepsilon_2$ is a stretched fast time scale. The quasi-steady state constraints, $\mathbf{g}_2' = 0$, are identified as linearly dependent.

The intermediate time scale dynamics is given in DAE form as:

$$\frac{d\mathbf{H}_1}{d\tau_1} = \mathbf{g}_1 + \mathbf{F}_2 \mathbf{B}_2 \bar{\mathbf{z}}_2$$
$$\tilde{\mathbf{g}}_2' = 0$$

with

$$\mathbf{B}_2 = \begin{bmatrix} \mathbf{I}_{97\times 97} \\ -\mathbf{1}_{1\times 97} \end{bmatrix}$$

where $\mathbf{H}_1 = \{H_1, H_2, \ldots, H_{100}, H_{101}\}$. $\tilde{\mathbf{g}}_2'$ represents the subset of linearly independent constraints given as $\mathbf{g}_2' = \mathbf{B}_2\tilde{\mathbf{g}}_2'$. $\bar{\mathbf{z}}_2$ is a vector of algebraic variables, defined by:

$$\lim_{\varepsilon_2 \to 0} \frac{\tilde{\mathbf{g}}_2'}{\varepsilon_2}$$

The quasi-steady state constraints for the intermediate time scale are identified as $\mathbf{g}_1 + \mathbf{F}_2\mathbf{B}_2\bar{\mathbf{z}}_2 = 0$, which are linearly independent. This is consistent with the fact that the energy dynamics of the network does not evolve in the slow time scale.

Note that the above equations have exactly the same canonical forms with the equations derived using the detailed mathematical analysis within the framework of singular perturbations in [82].

Remark 2.2. *The distillation column can be represented as a single node (instead of separate nodes for each tray) to analyze the time scale property of the dynamics of the overall enthalpy of the column. In this case, we obtain one equation for the dynamics of the distillation column, along with 6 equations for the other process units, instead of a set of equations which describe the energy dynamics of each tray of the distillation column.*

Also, the controlled outputs and the potential manipulated inputs available for each time scale can be readily identified. The enthalpies of the distillation column, the compressor and the combined reboiler-condenser need to be controlled in the fast time

scale (τ_2), and all but one out of four units should be controlled, i.e.:

$$\mathcal{Y}(\tau_2) = \{1, 2, \ldots, 97, 98\}$$

since no auxiliary node is connected to R_1 in $\mathcal{H}_2$. Potential manipulated inputs in this time scale include:

$$\mathcal{U}(\tau_2) = \{1-96, 2-1, 3-2, \ldots, 95-94, 96-97, 97-98, 98-95\}$$

The enthalpies of the trim condenser, the reflux drum and the auxiliary cooler as well as the total enthalpy of R_1 need to be controlled in the intermediate time scale, i.e.:

$$\mathcal{Y}(\tau_1) = \{R_1, 99, 100, 101\}$$

The following set of potential manipulated inputs is identified:

$$\mathcal{U}(\tau_1) = \{96-99, 97-100, 99-105, 100-101, 101-1, 101-107, 104-96\}$$

The reduced models obtained above can be used for deriving nonlinear model based controllers for each time scale, resulting in a hierarchical control strategy. Such controllers are well suited for achieving robust transition performance of the overall system (see [82] for example case studies).

2.4.2 Heat integrated network of distillation columns

Let us consider a distillation column network shown in Figure 2.15, which was proposed as a solution to a five-component separation problem [83]. In this configuration, four

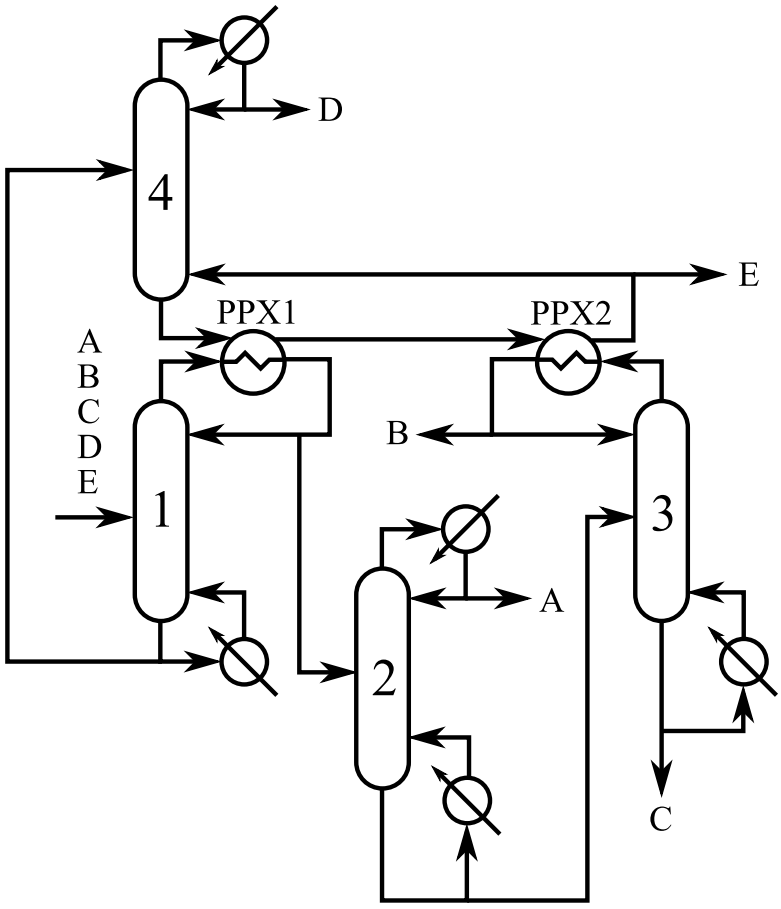


Figure 2.15: Heat integrated distillation column network for five-component separation

distillation columns are used, and the condensers of the first and third columns are combined with the reboiler of the fourth column through two process-to-process heat exchangers (PPX1 and PPX2). The energy flow graph of the network is shown in Figure 2.16. The node list is provided in Table 2.12. The orders of magnitude of the energy flows are determined following the information provided in [43], and are summarized in Table 2.13. We note that the energy flows of the network span three different orders of magnitude. ε_1 and ε_2 are small parameters ($\varepsilon_2 \ll \varepsilon_1 \ll 1$), which are the ratios of the nominal values of the energy input through the feed stream to the energy of the reflux

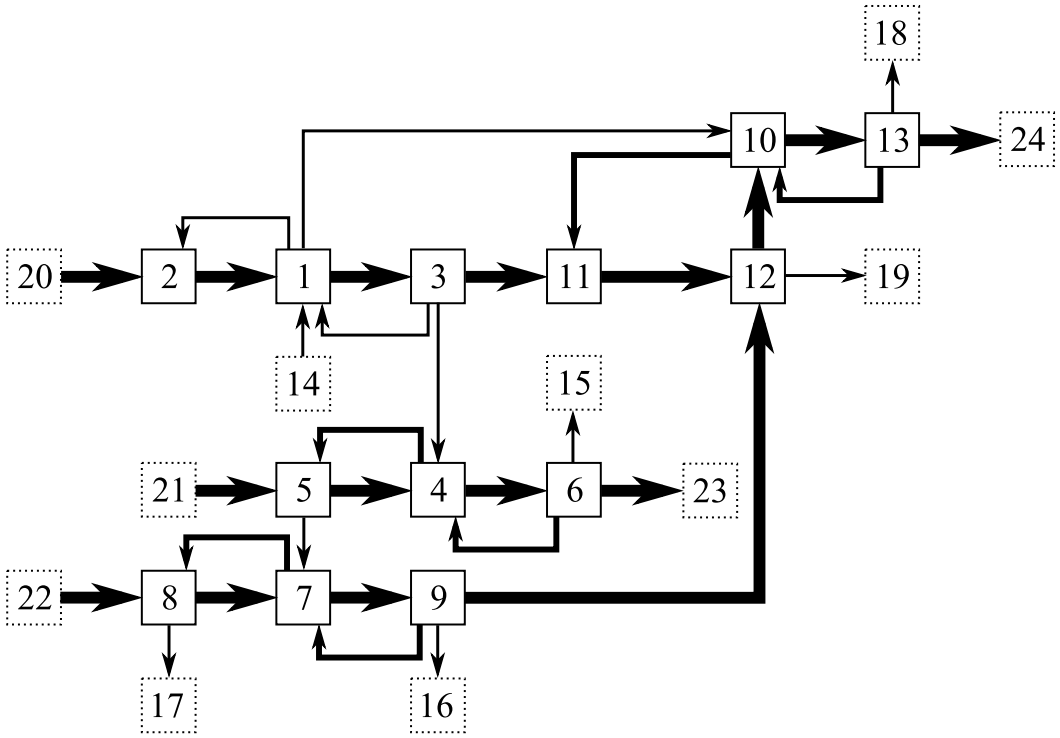


Figure 2.16: Energy flow graph of five-component separation

of the distillation column 2, and the duty of the reboiler of the distillation column 1, respectively:

$$\varepsilon_1 = \frac{h_{14-1,s}}{h_{6-4,s}}$$
$$\varepsilon_2 = \frac{h_{14-1,s}}{h_{20-2,s}}$$

We feed the information in Figure 2.16 and Table 2.13 to the graph reduction algorithm. Figure 2.17 shows the obtained subgraphs of the network corresponding to each time scale. In the fast time scale subgraph, three large throughputs (from the node 21 to the node 23, from the node 22 to the node 24 and from the node 20 to the node 24) are identified, and these throughputs need to be removed from the energy flow graph

Normal nodes		Auxiliary nodes	
Index	Unit	Index	Unit
1	Distillation column 1	14	Feed stream
2	DC1-Reboiler	15	Product stream (A)
3	PPX1-Hot	16	Product stream (B)
4	Distillation column 2	17	Product stream (C)
5	DC2-Reboiler	18	Product stream (D)
6	DC2-Condenser	19	Product stream (E)
7	Distillation column 3	20	DC1-Reboiler source
8	DC3-Reboiler	21	DC2-Reboiler source
9	PPX2-Hot	22	DC3-Reboiler source
10	Distillation column 4	23	DC2-Condenser sink
11	PPX1-Cold	24	DC4-Condenser sink
12	PPX2-Cold		
13	DC4-Condenser		

Table 2.12: Node list of five-component separation energy flow graph

before we proceed to the slower time scales as all the units in the throughput evolve in a single time scale (fast time scale in this case). Note that all the normal nodes are parts of at least one of three throughputs. The intermediate time scale subgraph is empty and thus not shown here. Also note that, in the slow time scale subgraph, we only have the energy flows of the smallest magnitude, and the auxiliary nodes, implying that no unit evolves in those time scales.

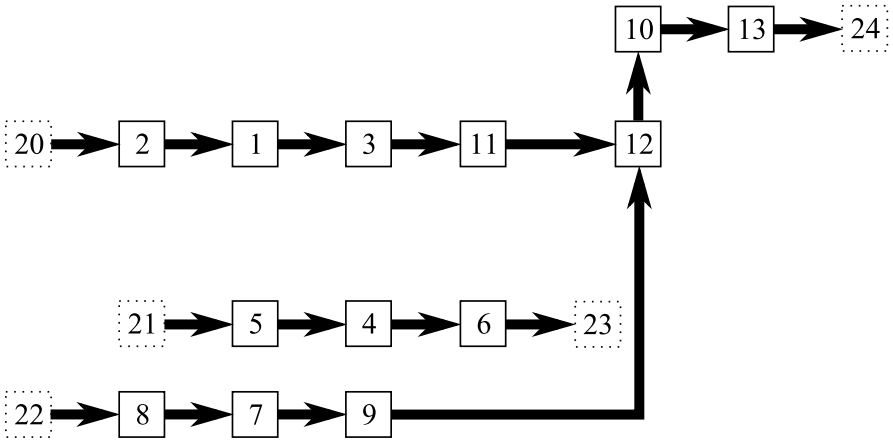
In summary, the energy dynamics of this network is mainly driven by the large energy throughputs, with the energy dynamics of every process unit evolving in the fast time scale. This implies that no time scale hierarchy in the energy dynamics exists that could be used to address energy management. This is a common feature in such networks due to the existence of large energy sources/sinks forming large energy throughputs.

In the cases where we do not have multi-time scale energy dynamics, we cannot implement a hierarchical control strategy as we did for the VRD and HDA examples. For such cases, one can seek a way to *decompose* the network into smaller subnetworks

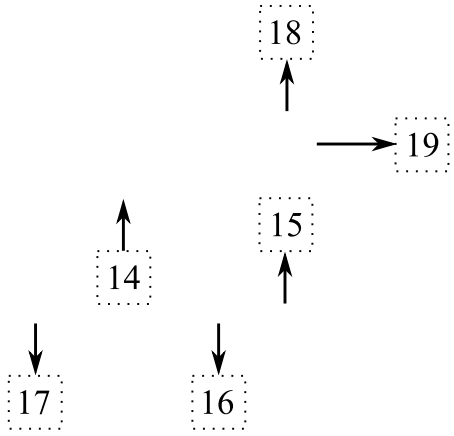
Energy flow	Order of magnitude	Energy flow	Order of magnitude
2→1	$\mathcal{O}(1/\varepsilon_2)$	7→8	$\mathcal{O}(1/\varepsilon_1)$
1→3	$\mathcal{O}(1/\varepsilon_2)$	9→7	$\mathcal{O}(1/\varepsilon_1)$
5→4	$\mathcal{O}(1/\varepsilon_2)$	10→11	$\mathcal{O}(1/\varepsilon_1)$
4→6	$\mathcal{O}(1/\varepsilon_2)$	13→10	$\mathcal{O}(1/\varepsilon_1)$
8→7	$\mathcal{O}(1/\varepsilon_2)$	14→1	$\mathcal{O}(1)$
7→9	$\mathcal{O}(1/\varepsilon_2)$	1→2	$\mathcal{O}(1)$
3→11	$\mathcal{O}(1/\varepsilon_2)$	3→1	$\mathcal{O}(1)$
9→12	$\mathcal{O}(1/\varepsilon_2)$	1→10	$\mathcal{O}(1)$
11→12	$\mathcal{O}(1/\varepsilon_2)$	6→15	$\mathcal{O}(1)$
12→10	$\mathcal{O}(1/\varepsilon_2)$	9→16	$\mathcal{O}(1)$
10→13	$\mathcal{O}(1/\varepsilon_2)$	8→17	$\mathcal{O}(1)$
4→5	$\mathcal{O}(1/\varepsilon_1)$	13→18	$\mathcal{O}(1)$
6→4	$\mathcal{O}(1/\varepsilon_1)$	12→19	$\mathcal{O}(1)$

Table 2.13: Summary of orders of magnitude of the energy flows of five-component separation network

such that the units in the same subnetwork interact strongly with each other, while the units in different subnetworks interact weakly. Then, separate controllers for each subnetwork can be designed, resulting in a quasi-decentralized control structure.



(a) with the largest energy flows



(b) with the smallest energy flows

Figure 2.17: Subgraphs of the five-component separation energy flow graph

CHAPTER 3

Hierarchical control of complex material integrated process networks

3.1 Introduction

Material integration is a key enabler of process efficiencies in chemical plants. Despite the economic benefits that result from the efficient utilization of material resources, several operational and control challenges emerge; for example, disturbances occurring at one process unit may propagate in the plant, while there are reduced degrees of freedom for control due to the tight coupling between units. The design of control systems for tightly integrated plants is a major problem in process control. Tight integration limits the effectiveness of decentralized control approaches traditionally followed in plant-wide control [37, 38]. Alternative approaches that account for the networked nature in such plants have also been proposed [2, 16, 17, 60–66].

In our previous work, we have proposed a hierarchical control framework for tightly integrated plants that feature a segregation in the magnitude of recycle and throughput

flows [39]. This is predicated on the fact that large recycle flowrates induce a separation in the time scales of the response of individual units in a recycle loop (fast) and the overall recycle network (slow) [30]. It was further shown that if in addition to recycle, a small purge stream is present, an additional, slower time scale for the evolution of the purged species emerges [40,41]. Furthermore, it was shown that the flowrates of different magnitudes act in different time scales resulting in a transparent choice of possible manipulated inputs for addressing the control objectives in the different layers of the hierarchy. Singular perturbations was shown to be a natural mathematical framework for documenting these facts rigorously.

The focus of this chapter is on complex material integrated plants, which may consist of multiple interconnected recycle structures. Such plants may exhibit dynamics in multiple (more than two) time scales, which may not be apparent by simple observation. Although a singular perturbation analysis is in principle possible, it becomes cumbersome as process complexity increases. Our goal is to develop an efficient analysis framework which identifies the time scales where each process variable evolves, and the possible manipulated inputs in each time scale. To this end, we base our analysis on an abstracted representation of the material flow structure in the plant using a novel graph formalism. A graph-theoretic algorithm is then used to essentially mimic the steps of the singular perturbation based analysis and reduction. The result is an automated procedure that can be used efficiently for complex integrated plants. Two examples are used to illustrate the application of this procedure.

3.2 Analysis of multi-time scale dynamics: singular perturbations based framework

The mathematical structure of the dynamic models of processes with a single recycle loop was analyzed in [30,40,41]. It was shown that, in the case of large recycle flowrate, the material balance model takes the form:

$$\frac{d\mathbf{x}}{dt} = \mathbf{g}_s(\mathbf{x}, \mathbf{u}_s) + \frac{1}{\varepsilon}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l \tag{3.1}$$

where we have explicitly separated terms corresponding to the small inlet/outlet flowrates and slow chemical reactions, from those corresponding to the large recycle and internal flowrates. Here, $\mathbf{x}$ is the vector of material balance variables, $\mathbf{g}_s$ and $\mathbf{g}_l$ are appropriately defined vector functions, ε is a small parameter which can be defined as a ratio of the nominal flowrate of an *external* stream (e.g. feed stream, product stream) to the nominal flowrate of a large *internal* stream, and $\mathbf{u}_s$ and $\mathbf{u}_l$ are the vectors of scaled input variables corresponding to the flowrates of small external streams and large internal streams, respectively. When a small purge exists in addition, the following description results:

$$\frac{d\mathbf{x}}{dt} = \mathbf{g}_s(\mathbf{x}, \mathbf{u}_s) + \frac{1}{\varepsilon}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l + \varepsilon_p\mathbf{g}_p(\mathbf{x})u_p \tag{3.2}$$

where $\mathbf{g}_p$ is a vector function, ε_p is a small parameter defined as a ratio of the nominal flowrate of the small purge stream to the nominal flowrate of an external stream, and u_p represents the scaled flowrate of the small purge stream.

The above are singular perturbation models in non-standard form, where the different time scales cannot be readily associated with specific variables. The fast dynamics of the model of Eq.(3.1) can be obtained by defining a stretched time scale $\tau = t/\varepsilon$ and

taking the limit $\varepsilon \to 0$:

$$\frac{d\mathbf{x}}{d\tau} = \mathbf{g}_l(\mathbf{x})\mathbf{u}_l \tag{3.3}$$

The quasi-steady state constraints of the above equation, i.e. $0 = \mathbf{g}_l(\mathbf{x})\mathbf{u}_l$, are not linearly independent, and thus specify an equilibrium manifold in which the slow dynamics evolve. The description of the slow dynamics can be obtained by taking the same limit in the original time scale t, to obtain the following differential algebraic equation (DAE) system:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{g}_s(\mathbf{x}, \mathbf{u}_s) + \mathbf{B}(\mathbf{x})\mathbf{z} \\ 0 &= \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l \end{aligned} \tag{3.4}$$

where $0 = \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l$ represent a linearly independent subset of the quasi-steady state constraints, expressed as $\mathbf{g}_l(\mathbf{x})\mathbf{u}_l = \mathbf{B}(\mathbf{x})\tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l$. $\mathbf{z}$ are algebraic variables which represent the terms $\lim_{\varepsilon\to 0} \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l/\varepsilon$, and correspond to the finite, yet indeterminate, scaled differences of large flows in the limit as the recycle becomes infinite.

The above reduction framework lends itself to a hierarchical control framework:

- process unit level control is addressed in the fast time scale using $\mathbf{u}_l$ as manipulated inputs; the specification of $\mathbf{u}_l$ based on a control law results in the model of Eq.(3.4) depending only on $\mathbf{u}_s$.

- network level control is then addressed in the slow time scale using $\mathbf{u}_s$ as manipulated inputs.

In the case of a small purge stream, a similar reduction procedure can be followed

to obtain the description of the fast dynamics:

$$\frac{d\mathbf{x}}{d\tau} = \mathbf{g}_l(\mathbf{x})\mathbf{u}_l \tag{3.5}$$

and the dynamics after the fast boundary layer:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \varepsilon_p \mathbf{g}_p(\mathbf{x})u_p + \mathbf{g}_s(\mathbf{x},\mathbf{u}_s) + \mathbf{B}_2(\mathbf{x})\mathbf{z}_2 \\ 0 &= \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l \end{aligned} \tag{3.6}$$

where τ, $\tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l$, $\mathbf{B}_2(\mathbf{x})$ and $\mathbf{z}_2$ can be defined similarly as in the previous case. With $\mathbf{u}_l$ being specified by a control law, the constraints in Eq.(3.6) are differentiated to evaluate the algebraic variables $\mathbf{z}_2$:

$$\mathbf{z}_2 = -(\mathcal{L}_{\mathbf{B}_2}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l)^{-1}\left\{\varepsilon_p \mathcal{L}_{\mathbf{g}_p(\mathbf{x})u_p}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l + \mathcal{L}_{\mathbf{g}_s(\mathbf{x},\mathbf{u}_s)}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l\right\} \tag{3.7}$$

where $\mathcal{L}$ is a compact Lie derivative notation [41]. The model of Eq.(3.6) can be rewritten using the above solution for $\mathbf{z}_2$ as follows:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \varepsilon_p \mathbf{g}'_p(\mathbf{x})u_p + \mathbf{g}'_s(\mathbf{x},\mathbf{u}_s) \\ 0 &= \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l \end{aligned} \tag{3.8}$$

where $\mathbf{g}'_p$ and $\mathbf{g}'_s$ are adjusted vector functions which can be obtained by collecting the terms of the same magnitude:

$$\begin{aligned} \mathbf{g}'_p(\mathbf{x})u_p &= \mathbf{g}_p(\mathbf{x})u_p - \mathbf{B}_2(\mathcal{L}_{\mathbf{B}_2}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l)^{-1}\left\{\mathcal{L}_{\mathbf{g}_p(\mathbf{x})u_p}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l\right\} \\ \mathbf{g}'_s(\mathbf{x},\mathbf{u}_s) &= \mathbf{g}_s(\mathbf{x},\mathbf{u}_s) - \mathbf{B}_2(\mathcal{L}_{\mathbf{B}_2}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l)^{-1}\left\{\mathcal{L}_{\mathbf{g}_s(\mathbf{x},\mathbf{u}_s)}\mathbf{g}_l(\mathbf{x})\mathbf{u}_l\right\} \end{aligned}$$

By taking the limit $\varepsilon_p \to 0$, the model of the intermediate dynamics can be obtained:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{g}'_s(\mathbf{x}, \mathbf{u}_s) \\ 0 &= \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l \end{aligned} \tag{3.9}$$

A one-dimensional equilibrium manifold, in which a slower dynamics evolves, is specified by the quasi-steady state constraints, $0 = \mathbf{g}'_s(\mathbf{x}, \mathbf{u}_s)$, which are not linearly independent. Finally, we define a compressed, slow time scale $\theta = \varepsilon_p t$, and take the limit $\varepsilon_p \to 0$ to obtain the following DAE system:

$$\begin{aligned} \frac{d\mathbf{x}}{d\theta} &= \mathbf{g}'_p(\mathbf{x})u_p + \mathbf{B}_1(\mathbf{x})\mathbf{z}_1 \\ 0 &= \tilde{\mathbf{g}}_l(\mathbf{x})\mathbf{u}_l \\ 0 &= \tilde{\mathbf{g}}'_s(\mathbf{x}, \mathbf{u}_s) \end{aligned} \tag{3.10}$$

with $0 = \tilde{\mathbf{g}}'_s(\mathbf{x}, \mathbf{u}_s)$ being the linearly independent subset of the constraints defined as $\mathbf{g}'_s(\mathbf{x}, \mathbf{u}_s) = \mathbf{B}_1(\mathbf{x})\tilde{\mathbf{g}}'_s(\mathbf{x}, \mathbf{u}_s)$, and $\mathbf{z}_1$ being algebraic variables representing the finite, yet indeterminate terms $\lim_{\varepsilon_1 \to 0} \tilde{\mathbf{g}}'_s(\mathbf{x}, \mathbf{u}_s)/\varepsilon_p$, which correspond to the scaled differences of small flows in the limit as the purge stream becomes infinitely small.

In this case, an additional layer can be added to the hierarchical control structure mentioned above, which regulates the inventory of the purged species in the recycle loop in a slower time scale using u_p as manipulated inputs. Note that $\mathbf{u}_l$ and $\mathbf{u}_s$ have to be specified by a control law so that the model of Eq.(3.10) depends only on u_p.

Let us now consider the more general case of multiple interconnected recycle structures, where different material flows span m orders of magnitude. We assume that the flows of the smallest magnitude are of $\mathcal{O}(1)$, and that t represents the time scale where the slowest dynamics evolve (rather than the process time scale). The mathematical

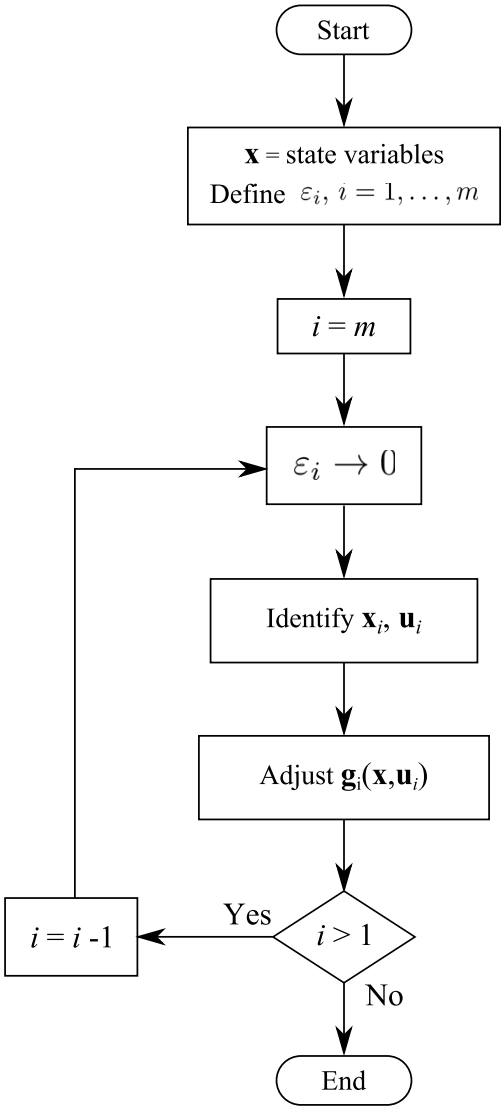


Figure 3.1: Flowchart of the analysis of multi-time scale dynamics using singular perturbations

model of such structures then can be cast in a singularly perturbed form as follows:

$$\frac{d\mathbf{x}}{dt} = \frac{1}{\varepsilon_1}\mathbf{g}_1(\mathbf{x}, \mathbf{u}_1) + \cdots + \frac{1}{\varepsilon_m}\mathbf{g}_m(\mathbf{x}, \mathbf{u}_m) \tag{3.11}$$

or in a more compact form:

$$\frac{d\mathbf{x}}{dt} = \sum_{i=1}^{m} \frac{1}{\varepsilon_i}\mathbf{g}_i(\mathbf{x}, \mathbf{u}_i) \tag{3.12}$$

where ε_i are small parameters, which can be defined as ratios of nominal flowrates of different orders of magnitude, such that $\varepsilon_i \ll 1$, $\varepsilon_{i+1}/\varepsilon_i \ll 1$ and $\varepsilon_1 \approx 1$. $\mathbf{u}_i$ represent the vectors of scaled manipulated inputs which correspond to the flowrates of $\mathcal{O}(1/\varepsilon_i)$. The reduced order dynamic models for each time scale $\tau_i = t/\varepsilon_i$ can be obtained through a sequential application of singular perturbations. Figure 3.1 shows a flowchart which summarizes such an analysis. We take the limit $\varepsilon_i \to 0$, starting from $i = m$, proceeding all the way to $i = 1$. In the time scale τ_i, we identify the following sets:

- $\mathbf{x}_i$: the material balance variables evolving in the time scale τ_i, which may/may not evolve in the subsequent time scales.

- $\mathbf{u}_i$: potential manipulated inputs which can be used to address the control objectives in the time scale τ_i.

In this case of multiple recycle loops, it is possible that the material flow structures around some large recycles form large material throughputs, if the magnitudes of the flowrates of the external streams are the same as those of the flowrates of the internal streams. The quasi-steady state constraints obtained for such recycles are linearly independent, specifying a set of equilibrium points (rather than equilibrium manifolds), i.e. the dynamics of such recycles will not be accompanied by a slower evolution. To take this into account, the elements of $g_{i'}$ $(i' = 1, \ldots, i-1)$, which contribute to the dynamics of

such recycles, are set to zero. In the case of large material recycles, $g_{i'}$ $(i' = 1, \ldots, i-1)$ are adjusted accordingly as discussed in the case of a small purge stream.

The above method becomes cumbersome with increasing size of networks. To this end, in what follows, we develop a generic, scalable graph theory based framework to automate the major steps of the above singular perturbations based framework.

3.3 Graph theoretic analysis of multi-time scale material dynamics

In this section, we propose a graph representation of process networks, which will be used as the input to the graph-theoretic algorithm that will be developed in the subsequent subsection.

3.3.1 Process flow graph

In our previous work [54,77], the time scale properties of the energy dynamics of process networks were analyzed on the basis of the concept of an *energy flow graph*, which captures the energy flow structure in such networks. This concept can be extended to represent the material flow structure of process networks by constructing multiple graphs each of which captures the flow structure of each chemical species. However, the dynamics of the holdups of each species cannot be analyzed independently. For example, in [42], it was shown that large flowrates of solvent act as carriers of solute that exists in small quantities, affecting the time scale in which the holdups of solute evolve. Thus, the analysis of the time scale properties of material dynamics on the basis of multiple graphs is not straightforward.

Motivated by this, we propose a new graph representation to capture the material flow structure in process networks. Specifically, a process network, which involves n_s

chemical species, is represented as a *process flow graph*, which is a weighted graph $\mathcal{G}(V, E)$, where V and E represent the set of nodes and the set of edges, respectively. In order to capture different elements of process networks (e.g. material sources/sinks, chemical reactions) properly, we define the following subsets of V and E:

- V_{normal}: the set of *normal nodes* which represent individual process units. There are n_s+1 material balance variables (n_s species holdups and total hold up) related to each normal node.
- V_{source}: the set of *auxiliary nodes* which represent *material sources* (e.g. feed stream sources). There are n_s (at most) material balance variables related to each source node, which have a potential to be used as manipulated inputs.
- V_{sink}: the set of *auxiliary nodes* which represent *material sinks* (e.g. product stream sinks).
- $V_{reaction}$: the set of *auxiliary nodes* which represent *chemical reactions.*
- E_{normal}: the set of *normal edges* which represent *material flows.*
- $E_{auxiliary}$: the set of *auxiliary edges* which represent *formation/consumption of chemical species.*
- Edge weights: each edge has n_s weights.

For a normal edge, the weights represent the orders of magnitude of different chemical species flowrates. For example, if the flowrates of species A and B in a certain stream are of $\mathcal{O}(1/\varepsilon_2)$ and $\mathcal{O}(1/\varepsilon_1)$, respectively, the weights of the edge, corresponding to that stream, will be (2,1). For an auxiliary edge, the weights represent the order of magnitude of the chemical reaction rate. For example, if the rate of reaction $A \rightarrow B$ is of $\mathcal{O}(1/\varepsilon_2)$, the weights of the edge, corresponding to that reaction, will be (-2,2). A

Subset	Schematic representation
V_{normal}	Solid rectangular node
V_{source}	Dashed circular node
V_{sink}	Dashed rectangular node
$V_{reaction}$	Solid circular node
E_{normal}	Solid arrow
$E_{auxiliary}$	Dashed arrow

Table 3.1: Schematic representation of different types of nodes and edges

negative value is used to represent the consumption rate, while a positive value is used to represent the formation rate.

Different types of nodes and edges will be distinguished schematically as summarized in Table 3.1. Also, material flows of different orders of magnitude will be distinguished using edges of different thickness (the larger the order, the thicker the edge). If m is so large that different magnitudes cannot be distinguished pictorially using different line thickness, other suitable indicators can be used to represent flows of different orders of magnitude.

The graph $\mathcal{G}(V, E)$ is initialized (i.e. each node/edge is classified into the appropriate subset) using Algorithm 1. Indegree and outdegree of node V_i are defined as the number of edges going into V_i and the number of edges emanating from V_i, respectively.

Example. Let us consider the network shown in Figure 3.2, where we have a gas-phase reactor where the reaction $A \rightarrow B$ occurs, and a condenser to separate the product B from the other species. N_A, N_B and N_I represent the interphase transfer rate of species A, B and I, respectively. We assume that a small quantity of inert I is introduced into the network at the flowrate $F_{I,in}$, and removed from the network through a purge stream of flowrate P. We also assume that the recycle flowrate R is much larger than the feed stream flowrate F_{in} or the product stream flowrate L. Two small parameters

Algorithm 1 InitializeGraph($\mathcal{G}(V, E)$)

```
for j = 1 : size(E) do
  if Any value of E_j is negative then
    Put E_j in E_auxiliary
  else
    Put E_j in E_normal
  end if
end for
for k = 1 : size(V) do
  if Indegree(V_i)=0 then
    if V_k is connected to any auxiliary edge then
      Put V_k in V_reaction
    else
      Put V_k in V_source
    end if
  else if Outdegree(V_k)=0 then
    Put V_k in V_sink
  else
    Put V_k in V_normal
  end if
end for
```

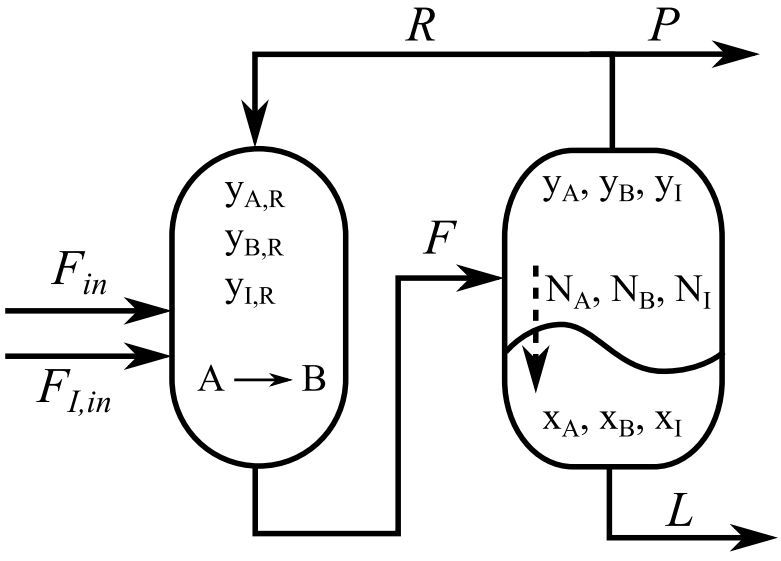


Figure 3.2: A simple reactor-separator network

can be defined as below:

$$\varepsilon_1 = \frac{P_s}{F_{in,s}},\ \varepsilon_2 = \frac{P_s}{R_s}$$

with $\varepsilon_2 \ll \varepsilon_1 \ll 1$. The subscript s denotes steady state value.

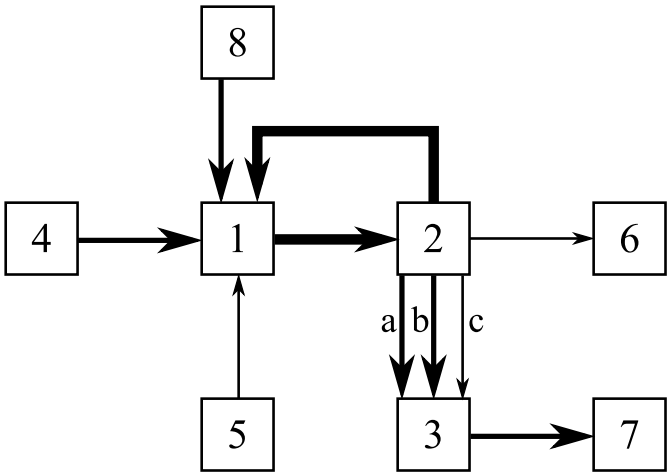
The network shown in Figure 3.2 can be represented as a process flow graph as shown in Figure 3.3a. The node list is provided in Table 3.2. The edge weights are determined based on the information provided in [41], and summarized in Table 3.3. The graph after the initialization is shown in Figure 3.3b.

3.3.2 Graph-theoretic algorithm

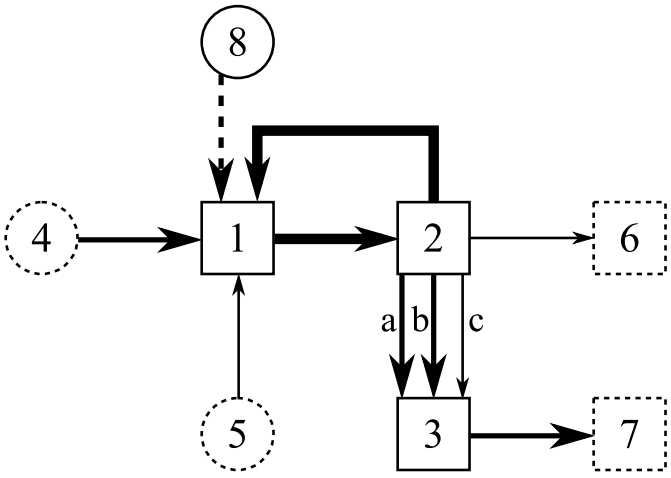
Now, let us describe the graph-theoretic algorithm, which mimics the singular perturbations, to extract the time scale properties of such networks, and to design hierarchical control structures. Specifically, we present the subroutines to mimic the following steps:

- take the limit $\varepsilon_i \to 0$
- identify $\mathbf{x}_i$ and $\mathbf{u}_i$
- identify the linear dependency/independency of the quasi-steady state constraints
- adjust the vector functions $\mathbf{g}_i(\mathbf{x})$

Algorithm 2 outlines the procedure for simplifying a complex process flow graph and formulating a hierarchical control design by exploiting the multi-time scale dynamics. The input to the algorithm is the process flow graph $\mathcal{G}(V, E)$. Since one seeks the evolution of the system for times $t = 0 \to \infty$, the algorithm begins with the largest order of magnitude material flows (i.e. material flows of $\mathcal{O}(1/\varepsilon_m)$) and proceeds to the smallest.



(a) Before initialization



(b) After initialization

Figure 3.3: Process flow graph of the reactor-separator network

Node	Unit
Normal nodes	
1	Reactor
2	Separator (gas phase)
3	Separator (liquid phase)
Source/sink/reaction nodes	
4	Feed stream source
5	Inert source
6	Purge stream sink
7	Product stream sink
8	$A \rightarrow B$

Table 3.2: Node list of $\mathcal{G}$ of the reactor-separator network

Algorithm 2 MaterialDynamicsAnalysis($\mathcal{G}(V, E)$)

1: InitializeGraph($\mathcal{G}(V, E)$)
2: $W = [m, m-1, \ldots, 1]$
3: **for** i=1:m **do**
4: $\mathcal{M}_i$=InducedSubgraph($W[i]$)
5: Put normal nodes of $ReducedNodeList_i$ in $\mathcal{N}_i$
6: $[\mathcal{T}_i, \mathcal{U}_i, \mathcal{Y}_i]$ =DynamicsAndControl($\mathcal{M}_i$)
7: $\mathcal{M}_{i,reduced}$ =GraphReduction($\mathcal{M}_i$)
8: PostProcess($\mathcal{G}, \mathcal{M}_{i,reduced}$)
9: **end for**
10: **return** $\mathcal{T}, \mathcal{U}, \mathcal{Y}$

Algorithm 3 describes the procedure of the subroutine InducedSubgraph. In the singular perturbations based framework, in order to obtain the description of the dynamics in the i-th time scale, one defines a stretched time scale $\tau_i = t/\varepsilon_i$ and takes the limit $\varepsilon_i \rightarrow 0$. The resulting equations will include only the terms of $\mathcal{O}(1/\varepsilon_i)$. In the graph theory based framework, the algorithm identifies the subset of E, $ReducedEdgeList_i$, which includes the edges of $\mathcal{O}(1/\varepsilon_i)$ only. Then, the nodes, which are connected to any edge in $ReducedEdgeList_i$, are taken to form a subset of V, $ReducedNodeList_i$. An *induced subgraph* $\mathcal{M}_i$ is the graph which can be defined by $ReducedEdgeList_i$ and $ReducedNodeList_i$. $\mathcal{N}_i$ is the set of nodes whose dynamics evolve in the time scale τ_i.

Edge	Weights (A, B, I)	Edge	Weights (A, B, I)
4→1	(2,0,0)	2→3 (a)	(2,0,0)
5→1	(0,0,1)	2→3 (b)	(0,2,0)
8→1	(-2,2,0)	2→3 (c)	(0,0,1)
1→2	(3,3,1)	2→6	(1,1,1)
2→1	(3,3,1)	3→7	(2,2,1)

Table 3.3: Edge list of $\mathcal{G}$ of the reactor-separator network

Algorithm 3 InducedSubgraph(i)

Define $ReducedEdgeList_i$
for $j = 1 : \text{size}(E)$ **do**
  if $\max_l(E_j[l]) = i$ **then**
    Put E_j in $ReducedEdgeList_i$
  end if
end for
Define $ReducedNodeList_i$
for $k = 1 : \text{size}(V)$ **do**
  if V_k is connected to any edge in $ReducedEdgeList_i$ **then**
    Put V_k in $ReducedNodeList_i$
  end if
end for
$\mathcal{M}_i = \mathcal{G}(ReducedNodeList_i, ReducedEdgeList_i)$
return $\mathcal{M}_i$

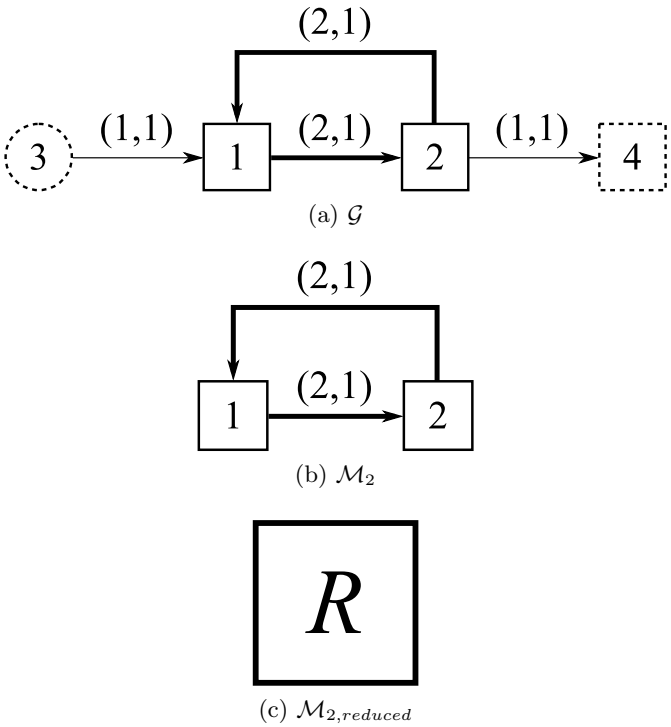
The procedure of the subroutine DynamicsAndControl is described in Algorithm 4. If the equation, which describes the dynamics of a certain variable, includes any term of $\mathcal{O}(1/\varepsilon_i)$, that variable evolves in the time scale τ_i. Similarly, if a certain node appears in $\mathcal{M}_i$, state variables related to that node have a potential to evolve in the time scale τ_i. In order to identify the set of state variables evolving in the time scale τ_i, $\mathcal{T}_i$, the algorithm checks, for each node in $\mathcal{M}_i$, the compositions of the edges connected to the corresponding node. As any state variable that evolves in the time scale τ_i can be controlled in the same time scale, the set of controlled outputs that can be regulated in the time scale τ_i, $\mathcal{Y}_i$, is the same as $\mathcal{T}_i$. All the edges appearing in $\mathcal{M}_i$ (essentially, flowrates of the streams represented by the edges) have a potential to be used as the

Algorithm 4 DynamicsAndControl($\mathcal{M}_i$)

for $k = 1 : \text{size}(ReducedNodeList_i)$ **do**
  if $ReducedNodeList_i[k]$ is not a partial recycle/throughput **then**
    Put $M(ReducedNodeList_i[k])$ to $\mathcal{T}_i$
    Put $M(ReducedNodeList_i[k])$ to $\mathcal{Y}_i$
  end if
  for $l = 1 : n_s$ **do**
    if l-th weight of any edge connected to $ReducedNodeList_i[k]$ is larger than 0 **then**
      Put $c_l(ReducedNodeList_i[k])$ to $\mathcal{T}_i$
      Put $c_l(ReducedNodeList_i[k])$ to $\mathcal{Y}_i$
    end if
  end for
end for
for $j = 1 : \text{size}(ReducedEdgeList_i)$ **do**
  Put $ReducedEdgeList_i[j]$ to $\mathcal{U}_i$
  if $ReducedEdgeList_i[j]$ is connected to a source node **then**
    Let that source node be denoted by S
    for $l = 1 : n_s$ **do**
      if l-th weight of $ReducedEdgeList_i[j]$ is larger than 0 **then**
        Put $c_l(S)$ to $\mathcal{U}_i$
      end if
    end for
  end if
end for

manipulated input to address the control objectives of the time scale τ_i. Thus, they are included in $\mathcal{U}_i$, the set of potential manipulated inputs available in the time scale τ_i. Also, if an edge represents a feed stream, the composition of that stream is available as potential manipulated input, and is included in $\mathcal{U}_i$.

The subroutine GraphReduction identifies all the *basic building blocks* in $\mathcal{M}_i$, which correspond to the prototype networks discussed in the previous section, i.e. networks with large material recycles and networks with small purge streams. These basic building blocks will be referred to as *pure recycles* and *partial recycles*, respectively. Then, the basic building blocks are replaced with the composite nodes to generate a *reduced*



(a) $\mathcal{G}$

(b) $\mathcal{M}_2$

(c) $\mathcal{M}_{2,reduced}$

Figure 3.4: Simple network with a pure recycle

subgraph $\mathcal{M}_{i,reduced}$. This subroutine employs a cycle detection algorithm [84] to identify recycles in the graphs.

Figure 3.4a shows a process flow graph for a pure recycle connecting two normal nodes 1 and 2. The pair on each edge represents the edge weights, which are assigned as discussed in the previous subsection. The induced subgraph $\mathcal{M}_2$ is shown in Figure 3.4b. Note that $\mathcal{M}_2$ represents a closed system. Thus, the overall network variables (e.g. $M_1 + M_2$, where M_k is the total holdup of k-th node) do not evolve in the time scale τ_2.

Figure 3.5a shows a partial recycle connecting two normal nodes 1 and 2. The induced subgraph $\mathcal{M}_2$ is shown in Figure 3.5b. Note that, although $\mathcal{M}_2$ represents an open system, no external flow introduces/removes species 2 from the recycle loop consisting of nodes 1 and 2. Thus, $c_{1,1} + c_{1,2}$ ($c_{l,k}$ represents the concentration of l-th

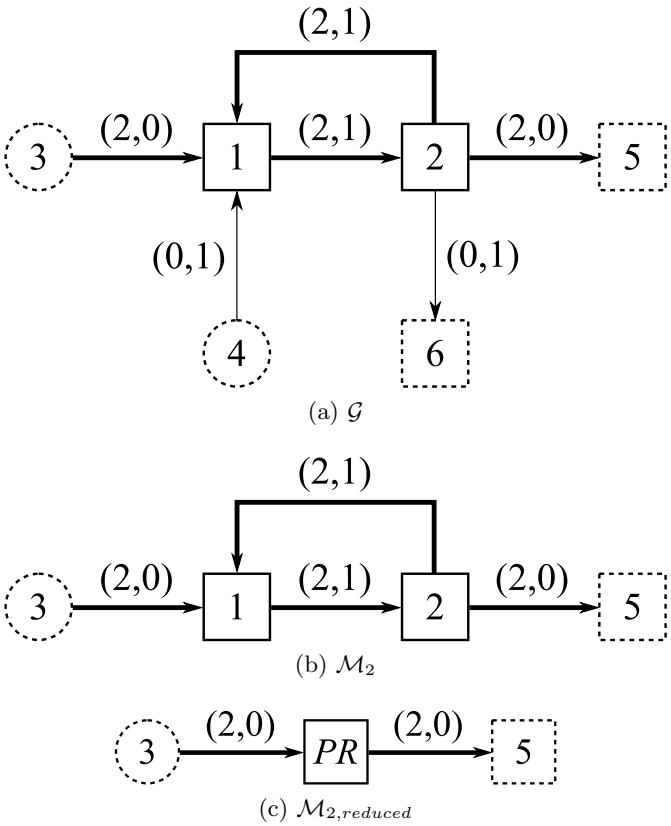


Figure 3.5: Simple network with a partial recycle

chemical species in k-th node), i.e. the overall holdup of species 1 in the recycle loop, evolves in the fast time scale, while $c_{2,1} + c_{2,2}$, i.e. the overall holdup of species 2 in the recycle loop, does not.

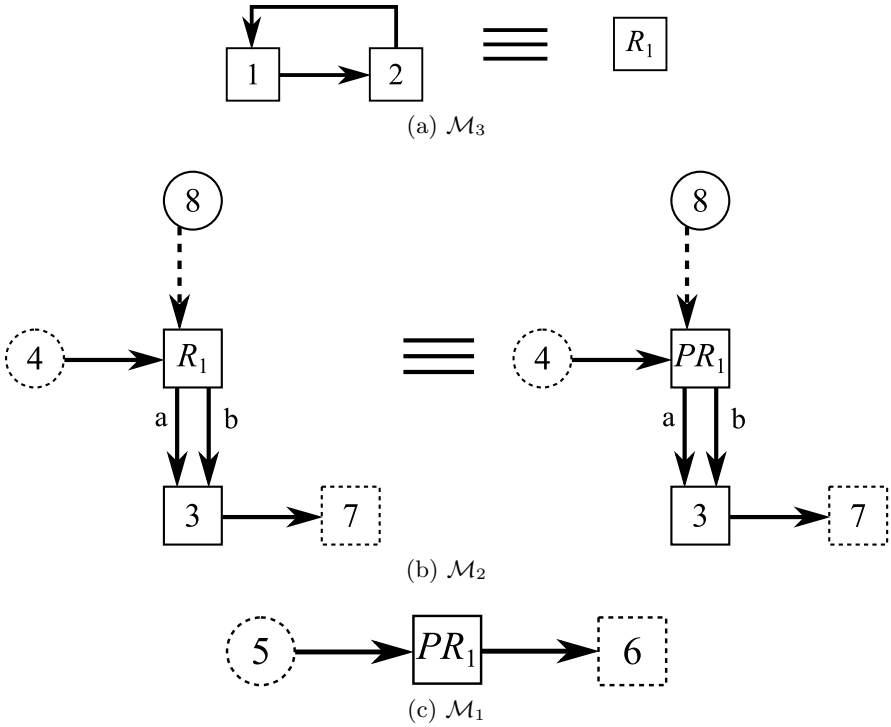
All the constituents of each recycle loop are clustered into a single composite node C. Then, the algorithm discriminates partial recycles from pure recycles by comparing the compositions of the internal flows and the external flows. Let us define two vectors, $\mathbf{t}$ and $\mathbf{r}$, for a recycle loop. The l-th element of $\mathbf{t}$, t_l, is equal to 1 if any large external flow contains the l-th chemical species, and 0 otherwise. Similarly, the l-th element of $\mathbf{r}$, r_l, is equal to 1 if any large internal flow contains the l-th chemical species, and 0 otherwise. Using $\mathbf{t}$ and $\mathbf{r}$, the following information can be inferred:

- the composite node is a pure recycle if $\mathbf{t} \equiv 0$. It will be labeled as R.
- the composite node is a partial recycle if $t_l \neq r_l$ for some l, and $\mathbf{t} \not\equiv 0$. It will be labeled as PR.
- if $t_l \neq r_l$ for a certain species, the overall holdup of that species in the recycle loop evolves in a slower time scale

Nodes 1 and 2 in Figure 3.4b can be clustered into a single composite node, and labeled as R to obtain $\mathcal{M}_{2,reduced}$ as shown in Figure 3.4c, since $\mathbf{t} \equiv 0$. Nodes 1 and 2 in Figure 3.5b can be clustered into a single composite node, and labeled as PR to obtain $\mathcal{M}_{2,reduced}$, which is shown in Figure 3.5c, as $t_2 \neq r_2$. Note that, as mentioned in the previous section, the material flow structure around a recycle loop can form a large material throughput. In such case, $t_l = r_l$, $\forall l$, and the composite node will be labeled as T.

The subroutine PostProcess removes all the constituents of throughputs from $\mathcal{G}$ as the dynamics of throughputs evolve in a single time scale. For partial recycles, the chemical species whose dynamics are accompanied by a slower evolution in the subsequent time scales are identified. Then, the weights of the edges in $\mathcal{G}$, which are connected to the constituents of partial recycles, are modified accordingly so that the dynamics of species for which $t_l = r_l$ are not affected by the smaller flows.

Example (continued)**.** Figure 3.6a shows the subgraph $\mathcal{M}_3$ of the process flow graph shown in Figure 3.3b. The nodes evolving in the time scale τ_3 are $\mathcal{N}_3 = \{1, 2\}$. One recycle is identified in this subgraph, which is determined to be a pure recycle, and is simplified as the composite node R_1. The subgraph $\mathcal{M}_2$ is shown in Figure 3.6b. A subset of nodes which evolve in the time scale τ_2 is given by $\mathcal{N}_2 = \{1, 2, 3\}$. In this subgraph, one recycle is identified, which is determined to be a partial recycle, and is labeled as PR_1. The constituents of throughputs (i.e. node 3) are removed from the



(a) $\mathcal{M}_3$

(b) $\mathcal{M}_2$

(c) $\mathcal{M}_1$

Figure 3.6: Subgraphs of the reactor-separator network

original graph. For PR_1, $t_l = r_l$ for A and B, while $t_l \neq r_l$ for I. Thus, the weights of the edges, which are connected to PR_1 in the subsequent time scale's subgraph, need to be changed so that the dynamics of A and B are not affected by the smaller flows. The subgraph $\mathcal{M}_1$ is shown in Figure 3.6c. Note that the edge weights of the edge $2 \rightarrow 6$ are changed from (1,1,1) to (0,0,1). The nodes evolving in the time scale τ_1 (=t) are $\mathcal{N}_1 = \{1, 2\}$.

Based on the above result, this network is expected to exhibit three-time scale material dynamics as summarized in Table 3.4. Note that this result matches the singular perturbations analysis provided in [41]. Also, potential manipulated inputs and controlled outputs for each time scale are listed in Table 3.5. Note that the variables in each brace cannot be controlled simultaneously, since the quasi-steady state constraints for

State	Evolution in		
	τ_3	τ_2	t
M_1	O	O	
$c_{A,1}$	O	O	
$c_{B,1}$	O	O	
$c_{I,1}$	O		O
M_2	O	O	
$c_{A,2}$	O	O	
$c_{B,2}$	O	O	
$c_{I,2}$	O		O
M_3		O	
$c_{A,3}$		O	
$c_{B,3}$		O	
$c_{I,3}$		O	

Table 3.4: Summary of the dynamics of the reactor-separator network

Time scale	$\mathcal{U}$	$\mathcal{Y}$
τ_3	F_{1-2}, F_{2-1}	$\{M_1, M_2\}$, $\{c_{A,1}, c_{A,2}\}$, $\{c_{B,1}, c_{B,2}\}$, $\{c_{I,1}, c_{I,2}\}$
τ_2	F_{4-1}, $F_{2-3(a)}$, $F_{2-3(b)}$, F_{3-7}, $c_{A,4}$	$(M_1 + M_2)$, $(c_{A,1} + c_{A,2})$, $(c_{B,1} + c_{B,2})$, M_3, $c_{A,3}$, $c_{B,3}$, $c_{I,3}$
t	F_{5-1}, F_{2-6}, $c_{I,5}$	$(c_{I,1} + c_{I,2})$

Table 3.5: Potential manipulated inputs and controlled outputs in each time scale: reactor-separator network

the variables are not linearly independent. Only all but one variables can be controlled independently.

Remark 3.1. *The proposed graph-theoretic algorithm focuses on the dynamics and control of material balance variables only. A framework, which combines the present work with the analysis of multi-time scale energy dynamics, will be the subject of the future work.*

In what follows, we apply the above algorithm to more realistic example process networks to further illustrate the application of the proposed framework.

3.4 Case study

3.4.1 HMF production process

Process description

Let us consider a process proposed in [85], in which 5-hydroxymethylfurfural (HMF) is produced from fructose. Figure 3.7 shows the process flow diagram of the network. A mixture of fructose and catalyst are fed to the continuously stirred biphasic tank reactor, where a complex chemical reaction system is present. The following simplified kinetic model was assumed:

$$\begin{aligned}
\text{Fructose} &\rightarrow \text{HMF} \\
\text{Fructose} &\rightarrow \text{BPA} \\
\text{HMF} &\rightarrow \text{Levulinic acid} + \text{Formic acid} \\
\text{HMF} &\rightarrow \text{BPB}
\end{aligned}$$

In this model, intermediate species between fructose and HMF are neglected, and the byproducts formed from fructose and HMF are lumped into groups BPA and BPB, respectively. The main product HMF is extracted to the organic phase, which is composed of 7:3 methyl iso-butylketone:2-butanol, from the aqueous phase where the reactions occur. In the liquid-liquid extractor, the remaining HMF is recovered from the aqueous phase outlet stream of the reactor. The stream leaving the aqueous phase of the extractor, which mainly consists of water, is recycled back to the reactor, while a part of it is purged to prevent the accumulation of byproducts. The organic phase outlet streams of the reactor and the extractor are fed to the evaporator to produce the purified HMF, and are recycled back to the reactor and the extractor. In order to maintain the inventory of the solvent, a small amount of fresh solvent is fed to the organic phase of the

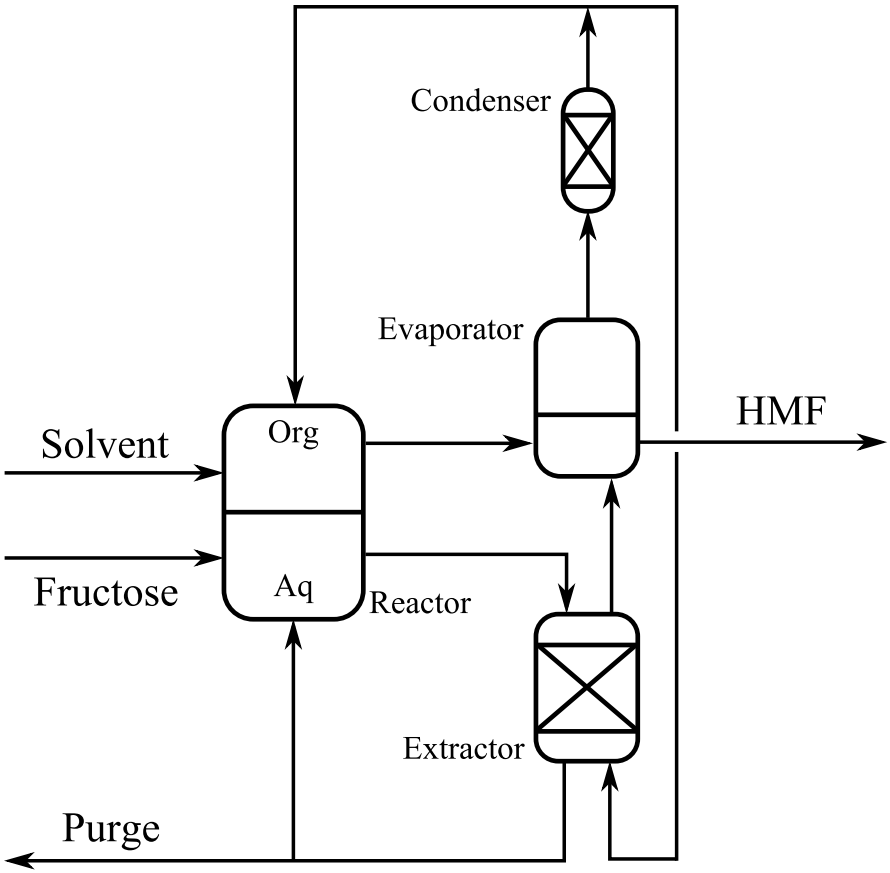


Figure 3.7: Process flow diagram of the HMF production process

reactor.

The presence of a purge stream and multiple recycle loops along with the segregation of material flows (as documented in [42]) suggests that there would be a possibility for the network to exhibit multi-time scale material dynamics. In what follows, we apply the proposed graph-theoretic algorithm to this process network.

Graph-theoretic analysis

The HMF production process, which is shown in Figure 3.7, can be represented as a process flow graph as shown in Figure 3.8. Table 3.6 contains the list of the nodes. Based

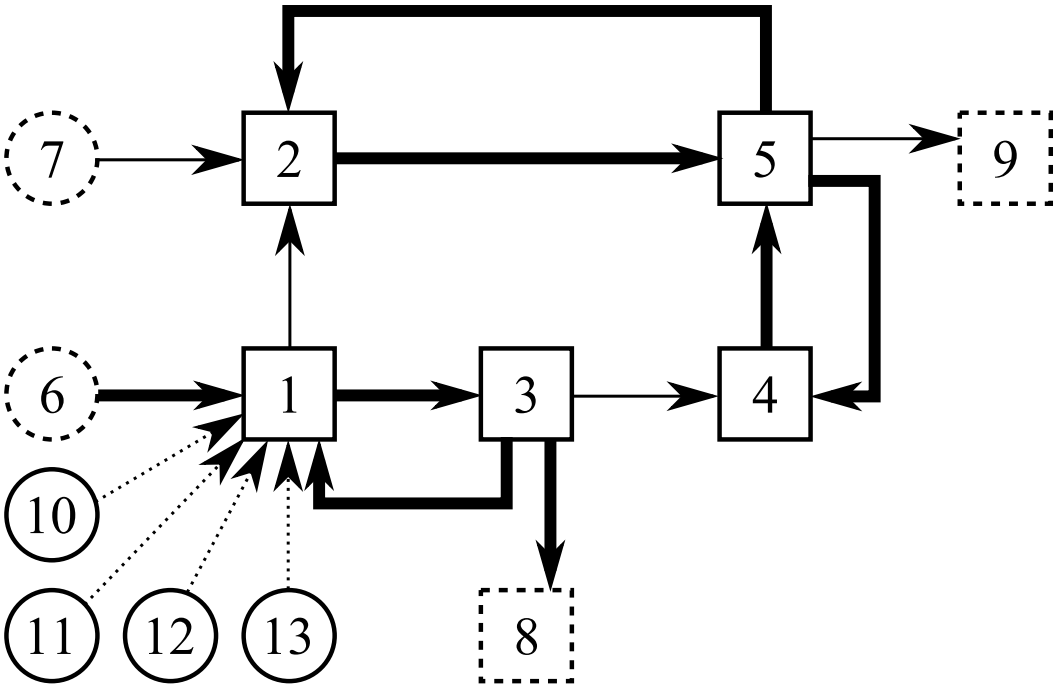
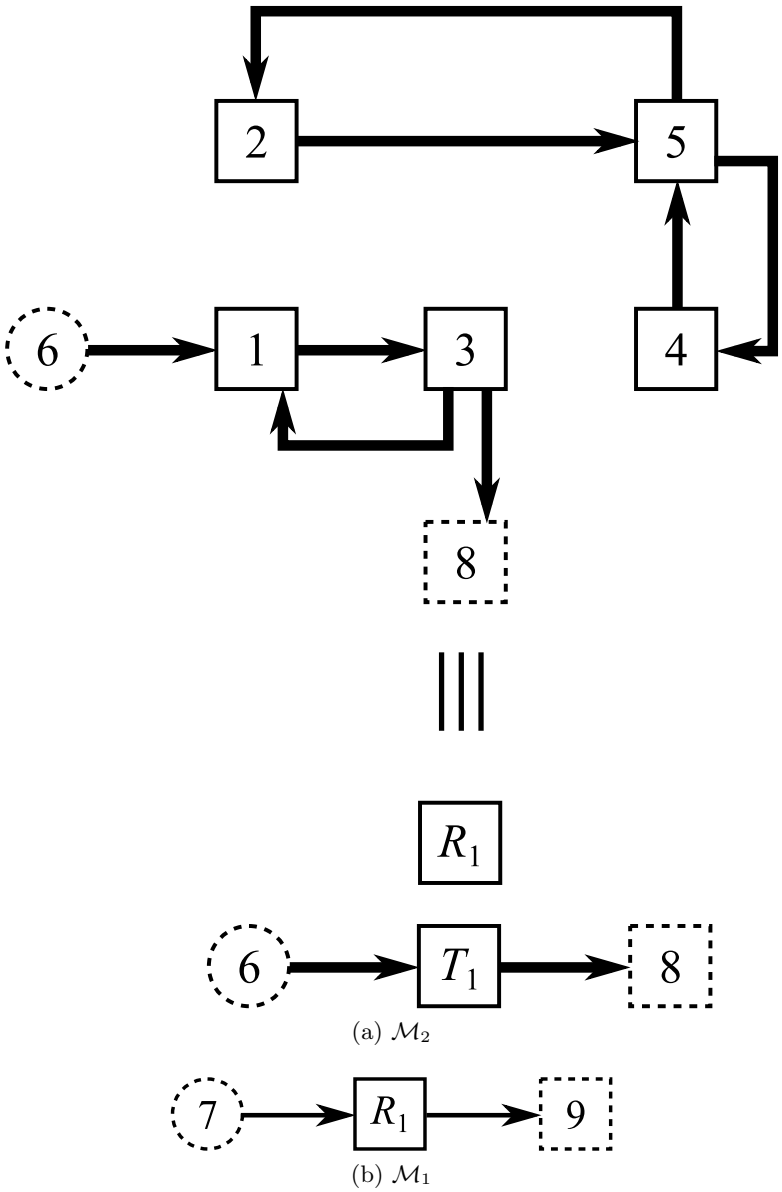


Figure 3.8: Process flow graph of the HMF production process

on the information provided in [42], the edge weights are determined, and summarized in Table 3.7. Different chemical species are indexed as shown in Table 3.8. Using the information in Figure 3.8 and Table 3.7, the proposed graph-theoretic algorithm is applied to this process network.

Figure 3.9a shows the subgraph $\mathcal{M}_2$ obtained from the process flow graph shown in Figure 3.8. The nodes evolving in the time scale τ_2 are $\mathcal{N}_2 = \{1, 2, 3, 4, 5\}$. Three recycles are identified in this subgraph. Two recycles (which are connected with each other through node 5) are determined to be pure recycles, and are simplified as the composite node R_1. The other recycle is determined to be a throughput, and is simplified as the composite node T_1. All the constituents of T_1 are removed from the original graph. The subgraph $\mathcal{M}_1$, which corresponds to the time scale t, is shown in Figure 3.9b. The nodes which evolve in this time scale are $\mathcal{N}_1 = \{2, 4, 5\}$.



(a) $\mathcal{M}_2$

(b) $\mathcal{M}_1$

Figure 3.9: Subgraphs of the HMF production process

Normal nodes		Source/sink nodes	
Node	Unit	Node	Unit
1	Reactor (aq)	6	Fructose feed
2	Reactor (org)	7	Solvent feed
3	Extractor (aq)	8	Purge
4	Extractor (org)	9	Product
5	Evaporator		
Reaction nodes			
Node	Reaction		
10	Fructose → HMF		
11	Fructose → BPA		
12	HMF → Levulinic acid + Formic acid		
13	HMF → BPB		

Table 3.6: Node list of $\mathcal{G}$ of the HMF production process

Edge	Weights	Edge	Weights
1→2	(0,1,0,0,0,0,0,0)	5→4	(0,1,0,0,0,0,0,2)
1→3	(1,1,1,1,1,1,2,0)	5→9	(0,1,0,0,0,0,0,0)
2→5	(0,1,0,0,0,0,0,2)	6→1	(1,0,0,0,0,0,2,0)
3→1	(1,1,1,1,1,1,2,0)	7→2	(0,0,0,0,0,0,0,1)
3→4	(0,1,0,0,0,0,0,0)	10→1	(-1,1,0,0,0,0,1,0)
3→8	(1,1,1,1,1,1,2,0)	11→1	(-1,0,0,0,1,0,0,0)
4→5	(0,1,0,0,0,0,0,2)	12→1	(0,-1,1,1,0,0,-1,0)
5→2	(0,1,0,0,0,0,0,2)	13→1	(0,-1,0,0,0,1,0,0)

Table 3.7: Edge list of $\mathcal{G}$ of the HMF production process

Index	Species	Index	Species
1	Fructose	5	BPA
2	HMF	6	BPB
3	Levulinic acid	7	Water
4	Formic acid	8	Organic solvent

Table 3.8: Index list for the chemical species: HMF production process

Based on these results, the HMF production process network is expected to exhibit two-time scale material dynamics. The state variables, which evolve in the fast and slow time scales, are summarized in Table 3.9 and Table 3.10, respectively. Note that this result matches the result provided in [42], which was obtained via the singular

State	Node				
	1	2	3	4	5
M	O	O	O	O	O
c_1	O		O		
c_2	O	O	O	O	O
c_3	O		O		
c_4	O		O		
c_5	O		O		
c_6	O		O		
c_7	O		O		
c_8		O		O	O

Table 3.9: Summary of the dynamics of the HMF production process in the time scale τ_2

State	Node				
	1	2	3	4	5
M		O		O	O
c_1					
c_2		O		O	O
c_3					
c_4					
c_5					
c_6					
c_7					
c_8		O		O	O

Table 3.10: Summary of the dynamics of the HMF production process in the time scale t

perturbations based analysis.

The controlled outputs and the potential manipulated inputs available in each time scale are also identified, and they are summarized in Table 3.11. On the aqueous part of the network, all the control objectives need to be addressed in the fast time scale. On the organic part of the network, the material balance variables of the individual process units need to be controlled, while the network level control objectives need to be addressed in the slow time scale. A hierarchical control structure, based on this

	$\mathcal{U}$	$\mathcal{Y}$
τ	F_{1-3}, F_{2-5}, F_{3-1}, F_{3-8}, F_{4-5}, F_{5-2}, F_{5-4}, F_{6-1}	M_1, M_3, $c_{1,1}$, $c_{1,3}$, $c_{2,1}$, $c_{2,3}$, $c_{3,1}$, $c_{3,3}$, $c_{4,1}$, $c_{3,3}$, $c_{4,3}$, $c_{5,1}$, $c_{5,3}$, $c_{6,1}$, $c_{6,3}$, $c_{7,1}$, $c_{7,3}$, $\{M_2, M_4, M_5\}$, $\{c_{2,2}, c_{2,4}, c_{2,5}\}$, $\{c_{8,2}, c_{8,4}, c_{8,5}\}$
t	F_{7-2}, F_{5-9}	$(M_2 + M_4 + M_5)$, $(c_{2,2} + c_{2,4} + c_{2,5})$, $(c_{8,2} + c_{8,4} + c_{8,5})$

Table 3.11: Potential manipulated inputs and controlled outputs in each time scale: HMF production process

decomposition, was developed and simulated in [42].

3.4.2 Vinyl acetate process

Process description

Let us now consider a vinyl acetate monomer process (VAc process) proposed in [86], whose process flow diagram is shown in Figure 3.10. In this process, acetic acid, ethylene and oxygen are fed into the reactor where the following chemical reactions occur:

$$2C_2H_4 + 2CH3COOH + O_2 \rightarrow 2CH_2 = CHOCOCH_3 + 2H_2O$$

$$C_2H_4 + 3O \rightarrow 2CO_2 + 2H_2O$$

It is assumed that the fresh ethylene feed contains a small amount of ethane, which is an inert species. The outlet stream of the reactor is then cooled using a process-to-process heat exchanger and a cooler. In the separator, the liquid phase of the reactor outlet stream is separated from the vapor phase, and it is fed into the distillation column. Vinyl acetate is recovered from the vapor outlet of the separator through the absorber, and it is also fed into the distillation column. The vapor outlet of the absorber, which consists mainly of the unreacted ethylene, is processed by the CO_2 removal units, and is recycled back into the reactor. In the distillation column, the unreacted acetic acid

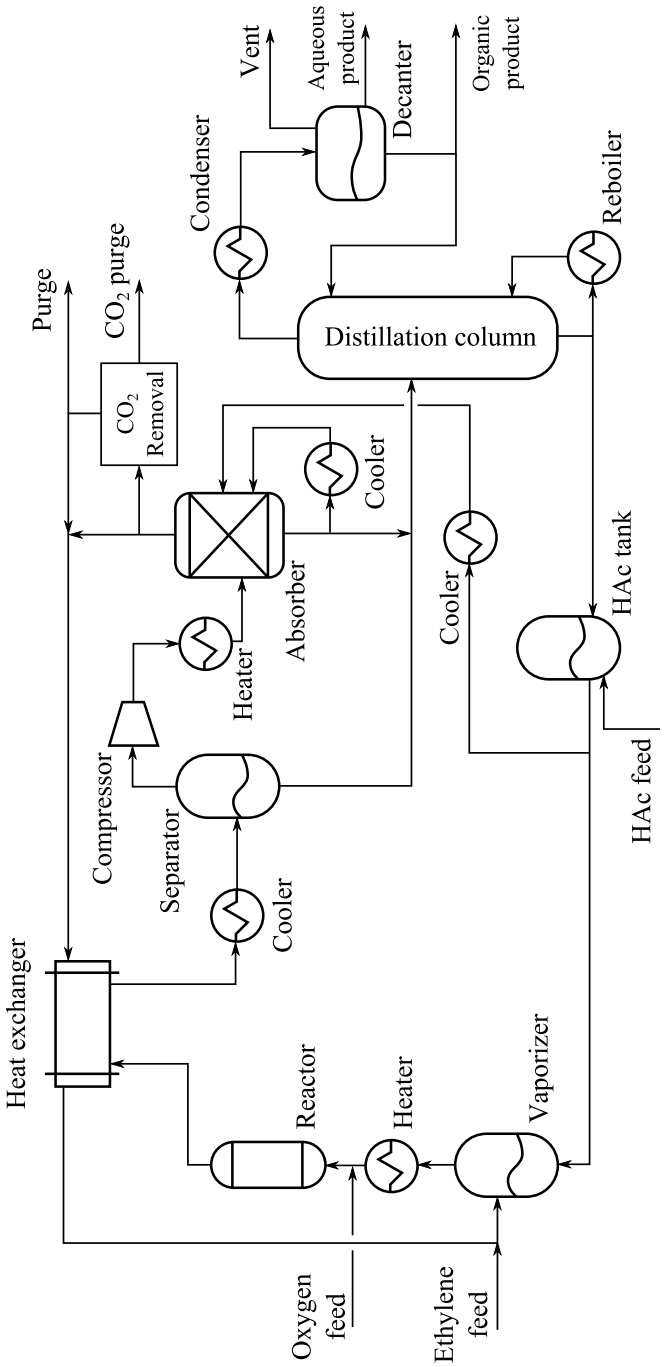


Figure 3.10: Process flow diagram of the VAc process

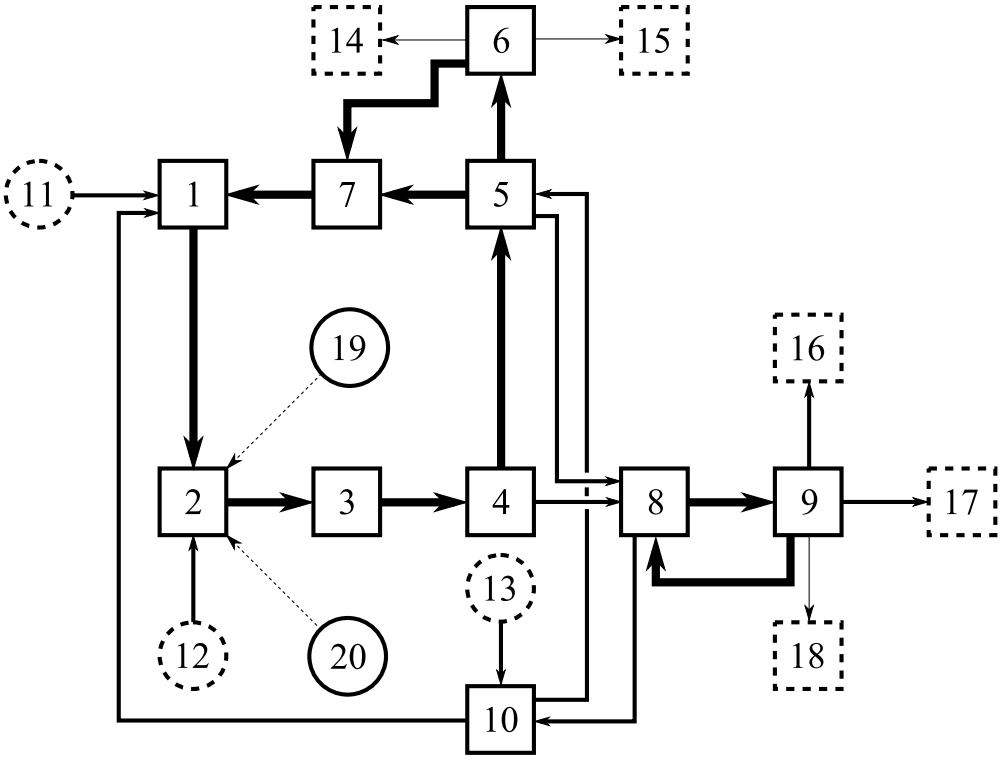


Figure 3.11: Process flow graph of the VAc process

is separated from the products, and is recycled back into the reactor. The inert species, ethylene, and the impurity species, CO_2, are removed from the process using the purge streams.

We note the following features of the process, which give rise to a potential of multi-time scale material dynamics:

- A large amount of unreacted reactants (i.e. ethylene and acetic acid) is recycled.
- There exist small purge streams.

Normal nodes			
Node	Unit	Node	Unit
1	Vaporizer	6	CO_2 removal unit
2	Reactor	7	FEHE-Cold
3	FEHE-Hot	8	Column
4	Separator	9	Decanter
5	Absorber	10	HAc tank
Source/sink nodes			
Node	Unit	Node	Unit
11	Ethylene feed	15	CO_2 purge
12	O_2 feed	16	Organic product
13	HAc feed	17	Aqueous product
14	Purge	18	Vent
Reaction nodes			
Node	Reaction		
19	$2C_2H_4 + 2CH_3COOH + O_2 \rightarrow 2CH_2{=}CHOCOCH_3 + 2H_2O$		
20	$C_2H_4 + 3O_2 \rightarrow 2CO_2 + 2H_2O$		

Table 3.12: Node list of $\mathcal{G}$ of the VAc process

Graph-theoretic analysis

Figure 3.11 shows the process flow graph of the VAc process, and the list of the nodes is tabulated in Table 3.12. The edge weights are determined using the steady state information documented in [86], and summarized in Table 3.13. The index list for different chemical species is provided in Table 3.14. The proposed graph-theoretic algorithm is applied to this process using the information in Figure 3.11 and Table 3.13 as inputs.

The subgraph $\mathcal{M}_3$, with the largest magnitude $\mathcal{O}(1/\varepsilon_3)$ flows, is shown in Figure 3.12a. The nodes evolving in the time scale τ_3 are $\mathcal{N}_3 = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$. In this subgraph, three recycles (two of which are interconnected) are identified, which are determined to be pure recycles, and are simplified as the composite nodes R_1 and R_2. Figure 3.12b shows the subgraph $\mathcal{M}_2$. A subset of the nodes which evolve in the time scale τ_2 is given by $\mathcal{N}_2 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. This subgraph contains four recycles which are determined to be partial recycles, and can be simplified to obtain the

Edge	Weights	Edge	Weights
1→2	(2,1,3,3,1,1,2)	10→1	(0,0,0,0,1,1,2)
2→3	(2,1,3,3,2,2,2)	10→5	(0,0,0,0,1,1,2)
3→4	(2,1,3,3,2,2,2)	9→16	(0,0,0,0,2,1,1)
4→5	(2,1,3,3,2,1,1)	9→17	(0,0,0,0,1,2,1)
5→6	(2,1,3,3,1,1,1)	11→1	(0,0,2,1,0,0,0)
5→7	(2,1,3,3,1,1,1)	12→2	(2,0,0,0,0,0,0)
6→7	(2,1,3,3,1,1,1)	13→10	(0,0,0,0,0,0,2)
7→1	(2,1,3,3,1,1,1)	6→14	(1,1,1,1,1,1,1)
8→9	(0,0,0,0,3,2,1)	6→15	(0,1,0,0,0,0,0)
9→8	(0,0,0,0,3,1,1)	9→18	(0,0,0,0,1,1,1)
4→8	(0,0,0,0,2,2,2)	19→2	(-1,0,-1,0,1,1,-1)
5→8	(0,0,0,0,2,1,2)	20→2	(-1,1,-1,0,0,1,0)
8→10	(0,0,0,0,1,1,2)		

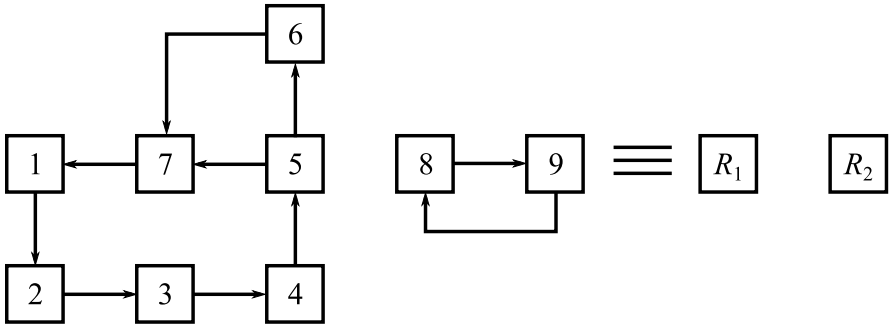
Table 3.13: Edge list of $\mathcal{G}$ of the VAc process

Index	Species	Index	Species
1	O_2	5	CH_2=$CHOCOCH_3$ (VAc)
2	CO_2	6	H_2O
3	C_2H_4	7	CH_3COOH (HAc)
4	C_2H_6		

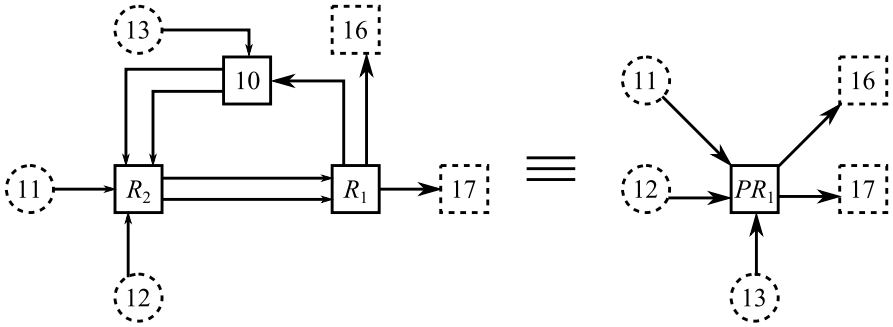
Table 3.14: Index list for the chemical species: VAc process

composite node PR_1. For PR_1, the internal and external compositions, i.e. $\mathbf{r}$ and $\mathbf{t}$, are given as $\mathbf{r} = [1, 1, 1, 1, 1, 1, 1]$ and $\mathbf{t} = [1, 0, 1, 1, 1, 1, 1]$, respectively. Thus, only the dynamics of the species 2 (i.e. CO_2) will be accompanied by a slower evolution in the subsequent time scale. The subgraph $\mathcal{M}_1$, corresponding to the time scale t, is shown in Figure 3.12c. The nodes evolving in this time scale are $\mathcal{N}_1 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Note that the weights of the edges, which are of $\mathcal{O}(1)$, and connected to PR_1, are modified accordingly as summarized in Table 3.15.

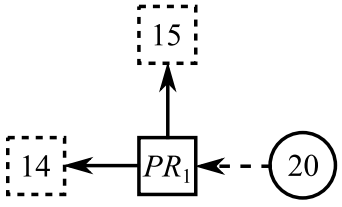
The VAc process is expected to exhibit three-time scale dynamics based on the results above. The state variables evolving in the time scales τ_3, τ_2 and t are summarized in Table 3.16, Table 3.17 and Table 3.18, respectively.



(a) $\mathcal{M}_3$



(b) $\mathcal{M}_2$



(c) $\mathcal{M}_1$

Figure 3.12: Subgraphs of the VAc process

Edge	Original weights	Modified weights
9→14	(1,1,1,1,1,1,1)	(0,1,0,0,0,0,0)
9→15	(0,1,0,0,0,0,0)	(0,1,0,0,0,0,0)
9→18	(0,0,0,0,1,1,1)	Removed
19→2	(-1,0,-1,0,1,1,-1)	Removed
20→2	(-1,1,-1,0,0,1,0)	(0,1,0,0,0,0,0)

Table 3.15: Original and modified weights of the edges of $\mathcal{O}(1)$

State	Node									
	1	2	3	4	5	6	7	8	9	10
M	O	O	O	O	O	O	O	O	O	
c_1	O	O	O	O	O	O	O			
c_2	O	O	O	O	O	O	O			
c_3	O	O	O	O	O	O	O			
c_4	O	O	O	O	O	O	O			
c_5	O	O	O	O	O	O	O	O	O	
c_6	O	O	O	O	O	O	O	O	O	
c_7	O	O	O	O	O	O	O	O	O	

Table 3.16: Summary of dynamics of the VAc process in the time scale τ_3

State	Node									
	1	2	3	4	5	6	7	8	9	10
M	O	O	O	O	O	O	O	O	O	O
c_1	O	O	O	O	O	O	O			
c_2										
c_3	O	O	O	O	O	O	O			
c_4	O	O	O	O	O	O	O			
c_5	O	O	O	O	O	O	O	O	O	O
c_6	O	O	O	O	O	O	O	O	O	O
c_7	O	O	O	O	O	O	O	O	O	O

Table 3.17: Summary of dynamics of the VAc process in the time scale τ_2

Also, the potential manipulated inputs and the controlled outputs in each time scale are identified as tabulated in Table 3.19. In the fast time scale, the material balance variables of the individual process units (except for the HAc tank) need to be controlled. The material balance variables associated with the HAc tank need to be controlled in

State	Node									
	1	2	3	4	5	6	7	8	9	10
M										
c_1										
c_2	O	O	O	O	O	O	O	O	O	O
c_3										
c_4										
c_5										
c_6										
c_7										

Table 3.18: Summary of dynamics of the VAc process in the time scale t

the intermediate time scale. Also, the network level dynamics of all the species (except for CO_2) needs to be controlled in this time scale. In the slow time scale, the total holdup of CO_2 in the network needs to be controlled.

Numerical simulations

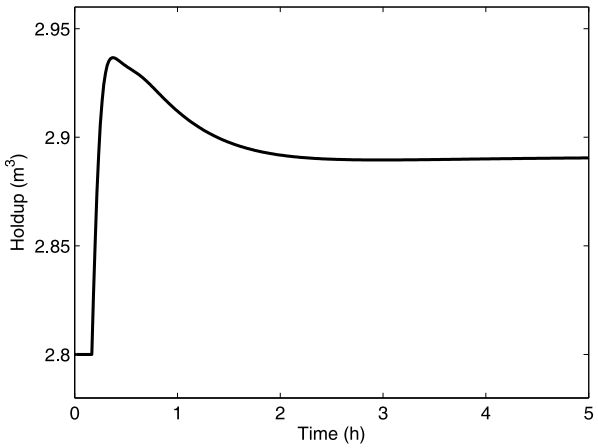
Let us now perform numerical simulations to validate the results obtained above using the mathematical model developed in [87]. P controllers were implemented to control the holdups of the separator, the absorber and the decanter to ensure the stability of the network.

We apply a disturbance to the network by increasing the flowrate of the vaporizer inlet from 2.2 kmol/min to 2.64 kmol/min at t=10 min. Figure 3.13a and Figure 3.13b show the evolutions of the total holdup of the vaporizer and the liquid holdup of the separator, respectively. Note that both holdups increase very quickly and then slowly go back to their new steady states. The evolutions of the species holdups of O_2 and CO_2 in the liquid phase of the separator are shown in Figure 3.14a and Figure 3.14b, respectively. We note that the holdup of CO_2 keeps decreasing slowly even when the holdup of O_2 is at its new steady state, suggesting a slower evolution of CO_2 holdup. Note that the simulation results are consistent with the results obtained in the previous

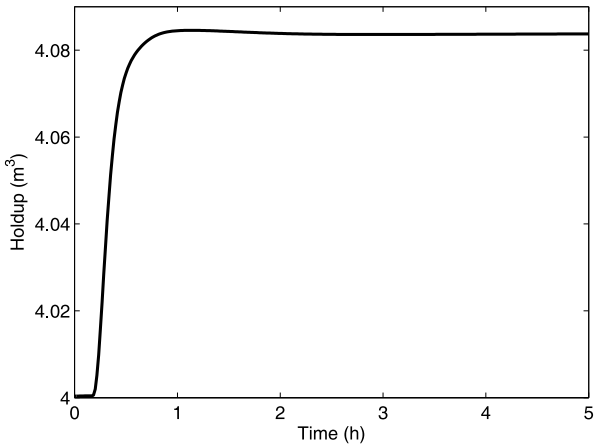
	$\mathcal{U}$	$\mathcal{Y}$
τ_3	F_{1-2}, F_{2-3}, F_{3-4}, F_{4-5}, F_{5-6}, F_{5-7}, F_{6-7}, F_{7-1}, F_{8-9}, F_{9-8}	$\{M_1, M_2, M_3, M_4, M_5, M_6, M_7\}$, $\{c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}, c_{1,6}, c_{1,7}\}$, $\{c_{2,1}, c_{2,2}, c_{2,3}, c_{2,4}, c_{2,5}, c_{2,6}, c_{2,7}\}$, $\{c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}, c_{3,6}, c_{3,7}\}$, $\{c_{4,1}, c_{4,2}, c_{4,3}, c_{4,4}, c_{4,5}, c_{4,6}, c_{4,7}\}$, $\{c_{5,1}, c_{5,2}, c_{5,3}, c_{5,4}, c_{5,5}, c_{5,6}, c_{5,7}\}$, $\{c_{6,1}, c_{6,2}, c_{6,3}, c_{6,4}, c_{6,5}, c_{6,6}, c_{6,7}\}$, $\{c_{7,1}, c_{7,2}, c_{7,3}, c_{7,4}, c_{7,5}, c_{7,6}, c_{7,7}, \}$, $\{M_8, M_9\}$, $\{c_{5,8}, c_{5,9}\}$, $\{c_{6,8}, c_{6,9}\}$, $\{c_{7,8}, c_{7,9}\}$
τ_2	F_{4-8}, F_{5-8}, F_{8-10}, F_{10-1}, F_{10-5}, F_{11-1}, F_{12-2}, F_{13-10}, F_{9-16}, F_{9-17}, $c_{3,11}$, $c_{4,11}$, $c_{1,12}$, $c_{7,13}$	M_{10}, $c_{5,10}$, $c_{6,10}$, $c_{7,10}$, $(M_8 + M_9)$, $(c_{5,8} + c_{5,9})$, $(c_{6,8} + c_{6,9})$, $(c_{7,8} + c_{7,9})$, $(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7)$, $(c_{1,1} + c_{1,2} + c_{1,3} + c_{1,4} + c_{1,5} + c_{1,6} + c_{1,7})$, $(c_{3,1} + c_{3,2} + c_{3,3} + c_{3,4} + c_{3,5} + c_{3,6} + c_{3,7})$, $(c_{4,1} + c_{4,2} + c_{4,3} + c_{4,4} + c_{4,5} + c_{4,6} + c_{4,7})$, $(c_{5,1} + c_{5,2} + c_{5,3} + c_{5,4} + c_{5,5} + c_{5,6} + c_{5,7})$, $(c_{6,1} + c_{6,2} + c_{6,3} + c_{6,4} + c_{6,5} + c_{6,6} + c_{6,7})$, $(c_{7,1} + c_{7,2} + c_{7,3} + c_{7,4} + c_{7,5} + c_{7,6} + c_{7,7})$
t	F_{6-14}, F_{6-15}, F_{20-2}	$(c_{2,1}+c_{2,2}+c_{2,3}+c_{2,4}+c_{2,5}+c_{2,6}+c_{2,7}+c_{2,8}+c_{2,9}+c_{2,10})$

Table 3.19: Potential manipulated inputs and controlled outputs in each time scale: VAc process

subsection using the graph theory based analysis.

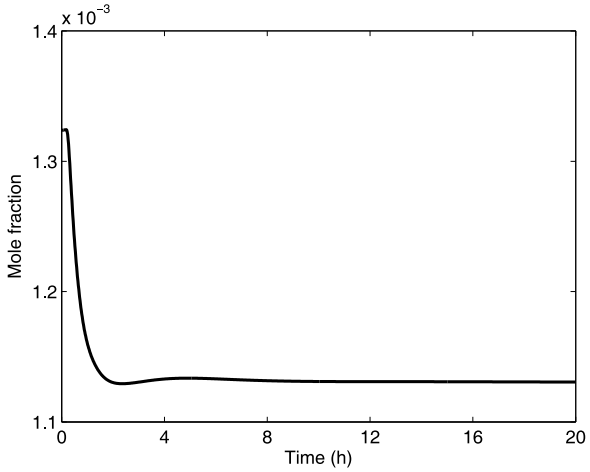


(a) Vaporizer

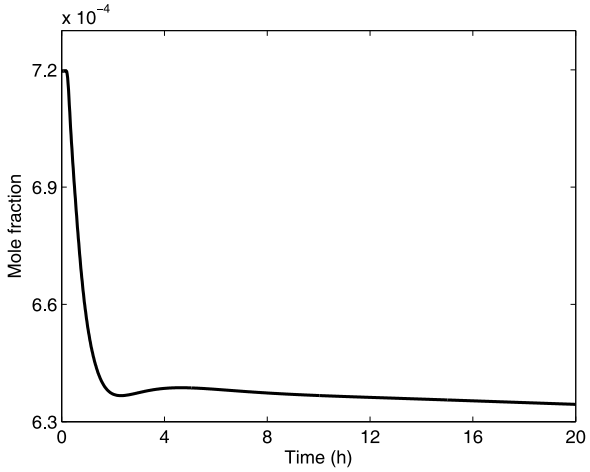


(b) Separator

Figure 3.13: Evolution of the total holdup



(a) O_2



(b) CO_2

Figure 3.14: Evolution of the species holdup in the liquid phase of the separator

CHAPTER 4

Automated synthesis of control configurations based on structural coupling*

4.1 Introduction

Control structure design is a classic problem in control that has received a lot of attention in the literature (e.g. [1]). In process control in particular, this problem has been studied extensively in the context of plant-wide control design (e.g. [2]). The problem of control structure selection becomes particularly challenging for tightly integrated process networks, which are the rule rather than the exception in modern chemical and energy plants, and also in the context of smart manufacturing [89]. Integration results in significant economic benefits, but also limits the available degrees of freedom and leads

*Reprinted with permission from Seongmin Heo, W. Alex Marvin, and Prodromos Daoutidis, *Chemical Engineering Science*, http://dx.doi.org/10.1016/j.ces.2015.03.049, 2015 [88].

to significant interactions that need to be addressed in controller design (including the selection of the control configuration). In a series of papers [30, 31, 40], we have documented that whenever integration results in large rates of recovery and recycle of material and/or energy (compared to the input/output flows), a time-scale hierarchy develops: individual units evolve in a fast time scale (and are affected by the large internal and recycle flows) and slower network-level dynamics emerge (that are affected by the small external flows). This particular feature can be exploited for the selection of manipulated inputs acting in appropriate time scales to address individual process control objectives and network level objectives, in the context of hierarchical control, as discussed in Chapter 2 and Chapter 3.

However, integrated process networks are not necessarily characterized by a segregation of material and/or energy flows. The design of control systems for such networks is a challenging, open problem, that has been addressed, for example, using passivity-based control (e.g. [61,66,90]), distributed control (e.g. [91–93]) and quasi-decentralized control (e.g. [94,95]). Central to this problem is the pairing of manipulated inputs and controlled outputs, either in the form of single-input single-output controllers or multivariable ones. A promising approach to this end is to exploit the extensive work in network theory and graph theory towards identifying "communities" (of manipulated inputs and controlled outputs, in our case) whose members interact strongly among them, yet are weakly coupled to the rest of the network members (e.g. [56, 96]). This community detection problem can be pursued using optimization of clustering parameters that capture the network connectivity (e.g. [97, 98]) or through spectral graph theory (e.g. [99]). The former approaches are generally more easily scalable to larger networks.

For the pairing of manipulated inputs and controlled outputs, which we will refer

to as the control configuration synthesis problem, a meaningful approach is to seek potential inputs and outputs that are strongly connected topologically. One measure of topological closeness is the concept of relative degree [69]. Relative degree essentially captures the directness of the effect of an input on an output, or the physical closeness between the two variables, and can be used to identify input/output clusters with favorable "structural coupling" in the above sense [69, 100]. Its generic calculation requires only structural information on the dynamic interactions in the network, and can be automated on the basis of an equation graph that captures these dynamic interactions.

In this chapter, we develop a method to systematically synthesize control configurations with favorable structural coupling, using relative degree as a measure of such coupling. Initially, we formulate an integer optimization problem to identify optimal distributions of inputs and outputs (essentially decentralized control configurations) that minimize the overall structural coupling in the network. We then propose a hierarchical clustering procedure which allows identifying groups of inputs and outputs that are strongly connected topologically, and are thus block decentralized control configuration candidates. The proposed approach is flexible as it allows generating control configurations that span the entire gamut from fully decentralized ones to the fully centralized one. It can also be automated for ease of implementation. Its application is illustrated through a case study on an integrated energy system.

4.2 Relative degree as a measure of structural coupling

Let us consider a general state space model of the form:

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{n_u} g_i(x) u_i \\ y_j &= h_j(x),\ j = 1, \ldots, n_y \end{aligned} \tag{4.1}$$

where $x \in \mathbb{R}^{n_x}$ denotes the state variables, and $u_i, y_j \in \mathbb{R}$ denote the input variables and the output variables, respectively. f, g_i are vector fields on $\mathbb{R}^{n_x}$, and h_j are scalar fields on $\mathbb{R}^{n_x}$.

The relative degree between u_i and y_j, r_{ij}, is defined as the smallest integer that satisfies [101]:

$$\mathcal{L}_{g_i}\mathcal{L}_f^{r_{ij}-1}h_j(x) \not\equiv 0 \tag{4.2}$$

where $\mathcal{L}$ represents the Lie derivative, defined as:

$$\mathcal{L}_f h(x) = \frac{\partial h(x)}{\partial x} f(x)$$

In [69], it was shown that the relative degree can be interpreted as a measure of how *direct* the effect of an input is on an output, as it represents the number of integrations needed for an input to affect an output. It was also argued that the relative degree can be used as a measure of *physical closeness* between an input and an output as it captures the sluggishness of the input/output response (in effect representing an apparent dead time). Also, it was shown that the generic calculation of relative degree requires only structural information of a process, i.e. knowledge of the interdependencies among the process variables. This can be used to construct an *equation graph* where nodes represent the state, input and output variables, and edges represent the interdependencies among the process variables. Edges are added based on the following rules:

- there is an edge from node x_k to node x_l if $\partial f_l(x)/\partial x_k \not\equiv 0$
- there is an edge from node u_i to node x_l if $g_{il}(x) \not\equiv 0$
- there is an edge from node x_k to node y_j if $\partial h_j(x)/\partial x_k \not\equiv 0$

where $f_l(x)$, $g_{il}(x)$ are the l-th element of $f(x)$, $g_i(x)$, respectively.

In such a graph, a *path* is defined as an open walk of nodes and edges of a graph such that no node is repeated. The *length* of a path is the number of edges contained in the path. An *input-to-output path* (IOP) is a path which starts from an input node and terminates at an output node. The relative degree r_{ij} is then related to the length of the shortest IOP connecting u_i and y_j of the equation graph as follows [69]:

$$r_{ij} = l_{ij} - 1 \tag{4.3}$$

Based on the above, relative degree can be used as a measure of structural coupling, i.e. coupling among the input/output variables based on their structural interdependencies, to provide guidelines for the design and evaluation of multi-loop control configurations. The procedure described in [69] involves the following steps:

1. Compute the relative degrees between all the inputs and the outputs to form a relative degree matrix (RDM) whose elements are the relative degrees between the inputs and the outputs:

$$M_r = \begin{bmatrix} r_{11} & \cdots & r_{1n_y} \\ \vdots & \ddots & \vdots \\ r_{n_u1} & \cdots & r_{n_un_y} \end{bmatrix}$$

2. Rearrange the outputs such that the minimum relative degree in each column of the RDM falls on the major diagonal. Then, the diagonal elements of the rearranged RDM represent the relative degrees between the input/output pairs forming the control configuration with a favorable structural coupling. The off-diagonal relative degrees in a row capture the coupling between a specific input and the other outputs, while the off-diagonal relative degrees in a column capture the coupling between a specific output and the other inputs.

3. Evaluate the overall structural coupling for the particular input/output assignment by comparing the diagonal and the off-diagonal relative degrees. Specifically, the differences between off-diagonal and diagonal relative degrees in a row (i.e. $r_{ij} - r_{ii}$) and in a column (i.e. $r_{ji} - r_{ii}$) provide a measure of the overall structural coupling. The larger these differences are, the more favorable the control configuration is.

The above procedure is time consuming to apply to large scale process networks as each step needs to be executed manually. Also, only fully decentralized control configurations are considered. In what follows, we exploit the graph theoretic interpretation of relative degrees to propose a generally applicable and easy to automate framework to synthesize optimal control configurations (ranging from fully decentralized to fully centralized) for process networks.

4.3 Control configuration synthesis based on structural coupling

In this section, we develop an optimization-based framework to synthesize control configurations based on structural coupling. We define a control configuration synthesis problem as a distribution problem, where the inputs and the outputs are distributed to the controllers. The assumption is that the controllers are square, i.e. the same numbers of inputs and outputs are distributed to each controller. We define the size of a controller as the number of inputs (or outputs) that are distributed to the controller. Initially, we limit the size of all the controllers to 1 (i.e. we consider fully decentralized control configurations only), and formulate an optimization problem to generate such decentralized control configurations so as to minimize the overall structural coupling between the different loops. Starting from such decentralized control configurations, we

then describe the synthesis of block decentralized control configurations by hierarchical clustering of the input/output pairs.

4.3.1 Synthesis of decentralized control configurations

Let us consider a system with n_u inputs and n_y outputs, and the corresponding RDM, M_r. We assume that we have more inputs than outputs which is typically the case in practical problems (i.e. $n_u > n_y$). For a given input/output assignment, consider the RDM rearranged such that the relative degrees between the chosen input/output pairs fall on the main diagonal. The structural coupling of this configuration can be evaluated using the differences between the diagonal and off-diagonal relative degrees of the first n_y rows of the RDM, and noting that the relative degrees in the remaining $n_u - n_y$ rows do not affect the structural coupling of the system as they represent the relative degrees between the unselected inputs (i.e. the inputs that are not distributed to any controller) and the outputs. The overall structural coupling for the given control configuration can thus be evaluated based on the value of the following term:

$$J_{DC} = \sum_{i=1}^{n_y} \sum_{j=1}^{n_y} (r_{ij} - r_{ii}) \tag{4.4}$$

The larger the above term is (i.e. the smaller the diagonal relative degrees and the larger the off-diagonal ones), the more favorable the control configuration is.

We note that the relative degrees in a column can be classified into three classes, diagonal ($\mathbf{r}_d$), off-diagonal ($\mathbf{r}_o$) and the remaining ones ($\mathbf{r}_r$). The relative degree r_{ij} is:

- in $\mathbf{r}_d$ if u_i and y_j are distributed to the same controller
- in $\mathbf{r}_r$ if u_i is not distributed to any controller
- in $\mathbf{r}_o$ otherwise

Let us now define two different sets of binary variables that will allow us to capture this classification without rearranging the RDM:

$$p_{ij} = \begin{cases} 1, & \text{if } u_i \text{ is distributed to the same controller with } y_j \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

$$q_i = \begin{cases} 0, & \text{if } u_i \text{ is not distributed to any controller} \\ 1, & \text{otherwise} \end{cases} \tag{4.6}$$

Note that, by definition, p_{ij} cannot be equal to 1 if q_i is equal to 0, leaving three possible combinations of p_{ij} and q_i, corresponding to the classes of relative degrees defined above:

$$\mathbf{r}_d = \{r_{ij} \in M_r | p_{ij} = 1, q_i = 1\}$$

$$\mathbf{r}_o = \{r_{ij} \in M_r | p_{ij} = 0, q_i = 1\}$$

$$\mathbf{r}_r = \{r_{ij} \in M_r | p_{ij} = 0, q_i = 0\}$$

Also note that, for systems with more outputs than inputs, q_j, instead of q_i, can be defined such that q_j is equal to 0 if y_j is not distributed to any controller.

The optimization problem for decentralized control configuration synthesis corresponds to an exhaustive search over all possible input/output distributions such that the term in Eq.(4.4) is maximized. Using the binary variables defined in Eqs.(4.5)-(4.6), this optimization problem can be stated as follows:

$$\text{maximize } J_{DC} = \sum_{i=1}^{n_u} \sum_{j=1}^{n_y} r_{ij} q_i - n_y \sum_{i=1}^{n_u} \sum_{j=1}^{n_y} r_{ij} p_{ij} \tag{4.7}$$

where the first and the second terms of the objective function represent the corresponding terms in Eq.(4.4), and the decision variables are p_{ij} and q_i. We now describe a set of constraints to complete the problem.

The number of input/output pairs needs to be equal to the number of outputs to ensure that all the outputs are distributed to a controller:

$$\sum_{i,j} p_{ij} = n_y \tag{4.8}$$

The following set of constraints states that any input or output can be distributed to one controller at most:

$$\sum_{j} p_{ij} \leq 1 \ \forall i \tag{4.9}$$

$$\sum_{i} p_{ij} \leq 1 \ \forall j \tag{4.10}$$

Finally, since any input u_i can be distributed to at most one output y_i:

$$q_i \leq \sum_{j} p_{ij} \ \forall i \tag{4.11}$$

This formulation (Eqs.(4.7)-(4.11)) corresponds to a bipartite matching problem [102], where a set of variables is divided into two disjoint subsets (in this case, the set of inputs and the set of outputs), and the variables from each set are matched so as to minimize/maximize the objective function. For such a problem, the existence of a global optimum is guaranteed, although in general there does not exist a unique solution [102]. This optimization problem can be solved within the environment of GAMS using the CPLEX Optimizer [103]. We can also find alternative solutions with the same objective value by excluding the optimal solutions that have been already found and resolving

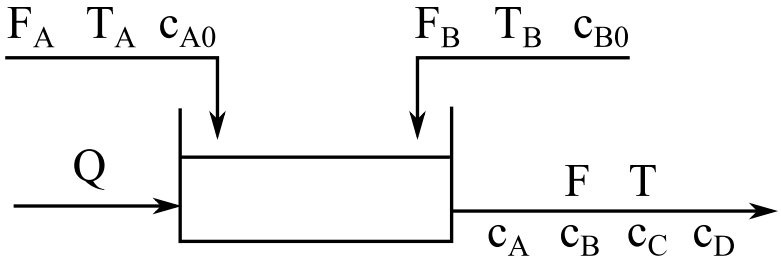


Figure 4.1: A simple CSTR

the optimization problem.

The output of the optimization problem is the $n_u \times n_y$ matrix P_{opt} whose i, j-th element is p_{ij}. Note that P_{opt} is a variant of a permutation matrix whose i-th row sum is 1 if u_i is distributed to a controller, and 0 otherwise. Using P_{opt}, we can obtain the *optimal* RDM ($n_y \times n_y$), $M_{r,opt}$, as:

$$M_{r,opt} = P_{opt}^T M_r \tag{4.12}$$

Note that, by premultiplying M_r by P_{opt}^T, we eliminate the $n_u - n_y$ rows of M_r, whose elements represent the relative degrees between the unselected inputs and the outputs, while rearranging the remaining n_y rows with respect to the optimal decentralized control configuration.

Example. For a simple illustration, let us consider a continuous stirred tank reactor (CSTR) shown in Figure 4.1. Feed streams of A and B with molar flow rates F_A, F_B, temperatures T_A, T_B, and concentrations c_{A0}, c_{B0}, respectively, are fed to the reactor, where a reaction $A + B \rightarrow C + D$ occurs, to produce a product stream with molar flow rate F, temperature T, and concentrations c_A, c_B, c_C, c_D. We assume that the reaction is endothermic, and heat is supplied to the reactor at a rate Q to facilitate the reaction.

Assuming constant density ρ and heat capacity C_p, the energy/material balances of

the system can be described by the following equations:

$$\begin{aligned}
\frac{dM}{dt} &= F_A + F_B - F \\
\frac{dc_A}{dt} &= \frac{F_A}{M}(c_{A0} - c_A) - \frac{F_B}{M}c_A - kc_Ac_B\exp\left(\frac{-E}{RT}\right) \\
\frac{dc_B}{dt} &= \frac{F_B}{M}(c_{B0} - c_B) - \frac{F_A}{M}c_B - kc_Ac_B\exp\left(\frac{-E}{RT}\right) \\
\frac{dc_C}{dt} &= -\frac{F_A + F_B}{M}c_C + kc_Ac_B\exp\left(\frac{-E}{RT}\right) \\
\frac{dT}{dt} &= \frac{F_A}{M}(T_A - T) + \frac{F_B}{M}(T_B - T) + \frac{Q}{MC_p} - \frac{\Delta H}{\rho C_p}kc_Ac_B\exp\left(\frac{-E}{RT}\right)
\end{aligned}$$

where M is the molar holdup of the reactor, and k and E are the rate constant and the activation energy of the reaction, respectively. ΔH represents the heat of reaction.

We define the following sets of the inputs and the outputs:

$$\mathbf{u} = \{F_A, F_B, F, Q\}$$

$$\mathbf{y} = \{M, c_A, c_C, T\}$$

Then, the equation graph of the system can be obtained as shown in Figure 4.2. The RDM of the system can be computed as follows:

$$M_r = \begin{array}{c} \\ F_A \\ F_B \\ F \\ Q \end{array} \begin{array}{c} \begin{array}{cccc} M & c_A & c_C & T \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ \infty & 2 & 2 & 1 \end{bmatrix} \end{array}$$

and the shortest IOPs between all input/output pairs are provided in Table 4.1.

By solving the optimization problem defined by Eqs.(4.7)-(4.11), we get two optimal

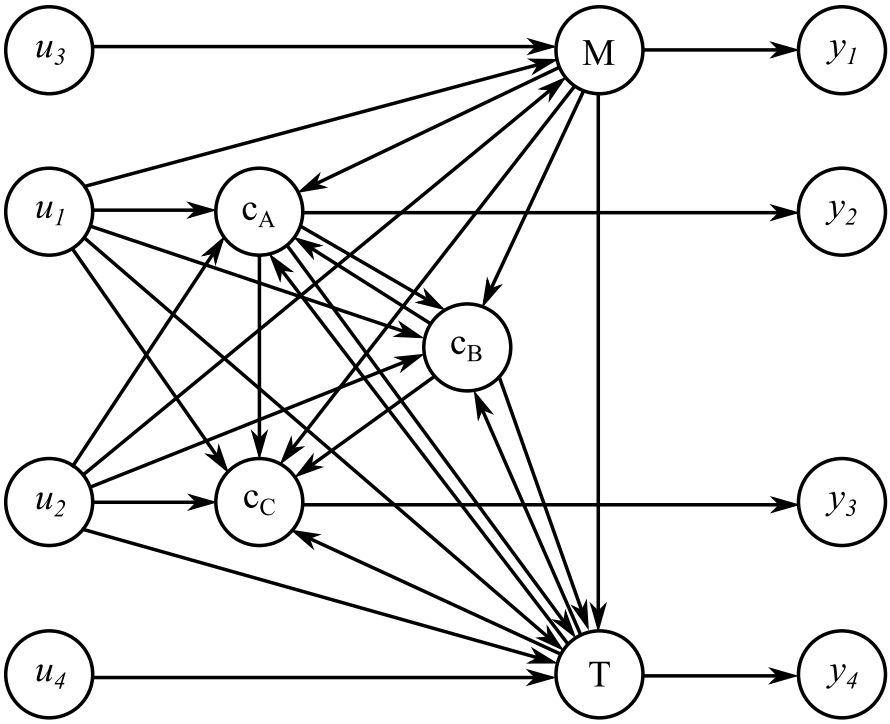


Figure 4.2: Equation graph of the simple CSTR example

solutions as follows:

$$P_{opt,1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_{opt,2} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

	$y_1(M)$	$y_2(c_A)$
$u_1(F_A)$	$u_1 \to M \to y_1$	$u_1 \to c_A \to y_2$
$u_2(F_B)$	$u_2 \to M \to y_1$	$u_2 \to c_A \to y_2$
$u_3(F)$	$u_3 \to M \to y_1$	$u_3 \to M \to c_A \to y_2$
$u_4(Q)$	No path	$u_4 \to T \to c_A \to y_2$
	$y_3(c_C)$	$y_4(T)$
$u_1(F_A)$	$u_1 \to c_C \to y_3$	$u_1 \to T \to y_4$
$u_2(F_B)$	$u_2 \to c_C \to y_3$	$u_2 \to T \to y_4$
$u_3(F)$	$u_3 \to M \to c_C \to y_3$	$u_3 \to M \to T \to y_4$
$u_4(Q)$	$u_4 \to T \to c_C \to y_3$	$u_4 \to T \to y_4$

Table 4.1: Shortest paths between the inputs and the outputs of the simple CSTR example

which translate into the following optimal RDMs:

$$M_{r,opt,1} = \begin{array}{c} \\ F \\ F_B \\ F_A \\ Q \end{array} \begin{array}{c} \begin{array}{cccc} M & c_A & c_C & T \end{array} \\ \begin{bmatrix} 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ \infty & 2 & 2 & 1 \end{bmatrix} \end{array}$$

$$M_{r,opt,2} = \begin{array}{c} \\ F \\ F_A \\ F_B \\ Q \end{array} \begin{array}{c} \begin{array}{cccc} M & c_A & c_C & T \end{array} \\ \begin{bmatrix} 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ \infty & 2 & 2 & 1 \end{bmatrix} \end{array}$$

In both control configurations, the molar holdup and the temperature of the reactor is controlled by manipulating the outlet flowrate and the heat input rate, respectively. The flowrates of the feed streams of A and B are respectively manipulated to control the concentrations of C and A of the outlet stream in the control configuration given

by $M_{r,opt,1}$, and to control the concentrations of A and C of the outlet stream in the control configuration given by $M_{r,opt,2}$.

4.3.2 Hierarchical clustering of input/output pairs

In this subsection, we propose a systematic way to cluster the input/output pairs hierarchically, by considering the structural coupling between different input/output *pair*s (rather than simply inputs and outputs). The goal is to generate a hierarchy of block decentralized control configurations with increasing controller sizes, whereby in each level we form clusters of input/output pairs which are structurally close in a sense that will be made precise shortly. The result of clustering can be visualized as a dendrogram where each level of the dendrogram represents a block decentralized control configuration.

Let us consider the following optimal RDM obtained as a solution to the optimization problem discussed above (in the case of multiple optimal solutions, each one can be considered separately):

$$M_{r,opt} = \begin{array}{c} \\ u_1 \\ \vdots \\ u_i \\ \vdots \\ u_j \\ \vdots \\ u_{n_y} \end{array} \begin{array}{c} \begin{array}{ccccccc} y_1 & \cdots & y_i & \cdots & y_j & \cdots & y_{n_y} \end{array} \\ \begin{bmatrix} r_{11} & \cdots & r_{1i} & \cdots & r_{1j} & \cdots & r_{1n_y} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{i1} & \cdots & r_{ii} & \cdots & r_{ij} & \cdots & r_{in_y} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{j1} & \cdots & r_{ji} & \cdots & r_{jj} & \cdots & r_{jn_y} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{n_y1} & \cdots & r_{n_yi} & \cdots & r_{n_yj} & \cdots & r_{n_yn_y} \end{bmatrix} \end{array} \tag{4.13}$$

Two input/output pairs, $\{u_i, y_i\}$ and $\{u_j, y_j\}$, will be structurally close if the inputs

u_i, u_j have comparable structural effect on the outputs y_i, y_j, i.e. the values of the off-diagonal relative degrees r_{ij} and r_{ji} are close to the values of the diagonal relative degrees r_{ii} and r_{jj}. Motivated by this, we define a distance between the two input/output pairs as the sum of the *largest* differences between the off-diagonal and diagonal relative degrees:

$$\begin{aligned} d(\{u_i, y_i\}, \{u_j, y_j\}) &= (\max\{r_{ij}, r_{ji}\} - r_{ii}) + (\max\{r_{ij}, r_{ji}\} - r_{jj}) \\ &= 2\max\{r_{ij}, r_{ji}\} - r_{ii} - r_{jj} \end{aligned} \tag{4.14}$$

Proposition 4.3.1. The distance defined in Eq.(4.14) is a pseudo-semi-metric [104], satisfying the following properties:

$$\begin{aligned} d(\{u_i, y_i\}, \{u_j, y_j\}) &\geq 0 \\ d(\{u_i, y_i\}, \{u_i, y_i\}) &= 0 \\ d(\{u_i, y_i\}, \{u_j, y_j\}) &= d(\{u_j, y_j\}, \{u_i, y_i\}) \end{aligned}$$

Proof. The objective function value computed from the above optimal RDM is:

$$\begin{aligned} J_{DC,opt} =& \sum_{k=1}^{n_y}\sum_{l=1}^{n_y}(r_{kl} - r_{ll}) \\ =& [(r_{11} - r_{11}) + \cdots + (r_{1i} - r_{11}) + \cdots + (r_{1j} - r_{11}) + \cdots + (r_{1n_y} - r_{11})] + \cdots \\ &+ [(r_{i1} - r_{ii}) + \cdots + (r_{ii} - r_{ii}) + \cdots + (r_{ij} - r_{ii}) + \cdots + (r_{in_y} - r_{ii})] + \cdots \\ &+ [(r_{j1} - r_{jj}) + \cdots + (r_{ji} - r_{jj}) + \cdots + (r_{jj} - r_{jj}) + \cdots + (r_{jn_y} - r_{jj})] + \cdots \\ &+ [(r_{n_y1} - r_{n_yn_y}) + \cdots + (r_{n_yi} - r_{n_yn_y}) + \cdots \\ &\qquad + (r_{n_yj} - r_{n_yn_y}) + \cdots + (r_{n_yn_y} - r_{n_yn_y})] \end{aligned}$$

Let us swap the i, j-th rows to get the following RDM:

$$M_r' = \begin{array}{c} \\ u_1 \\ \vdots \\ u_j \\ \vdots \\ u_i \\ \vdots \\ u_{n_y} \end{array} \begin{array}{c} \begin{array}{ccccccc} y_1 & \cdots & y_i & \cdots & y_j & \cdots & y_{n_y} \end{array} \\ \begin{bmatrix} r_{11} & \cdots & r_{1i} & \cdots & r_{1j} & \cdots & r_{1n_y} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{j1} & \cdots & r_{ji} & \cdots & r_{jj} & \cdots & r_{jn_y} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{i1} & \cdots & r_{ii} & \cdots & r_{ij} & \cdots & r_{in_y} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{n_y1} & \cdots & r_{n_yi} & \cdots & r_{n_yj} & \cdots & r_{n_yn_y} \end{bmatrix} \end{array}$$

The objective function value computed from the above RDM is:

$$\begin{aligned} J_{DC}' =& [(r_{11} - r_{11}) + \cdots + (r_{1i} - r_{11}) + \cdots + (r_{1j} - r_{11}) + \cdots + (r_{1n_y} - r_{11})] + \cdots \\ &+ [(r_{j1} - r_{ji}) + \cdots + (r_{ji} - r_{ji}) + \cdots + (r_{jj} - r_{ji}) + \cdots + (r_{jn_y} - r_{ji})] + \cdots \\ &+ [(r_{i1} - r_{ij}) + \cdots + (r_{ii} - r_{ij}) + \cdots + (r_{ij} - r_{ij}) + \cdots + (r_{in_y} - r_{ij})] + \cdots \\ &+ [(r_{n_y1} - r_{n_yn_y}) + \cdots + (r_{n_yi} - r_{n_yn_y}) + \cdots \\ &\qquad + (r_{n_yj} - r_{n_yn_y}) + \cdots + (r_{n_yn_y} - r_{n_yn_y})] \end{aligned}$$

The difference between the objective function values is:

$$\begin{aligned} J_{DC,opt} - J_{DC}' &= [(r_{i1} - r_{ii}) + \cdots + (r_{ii} - r_{ii}) + \cdots + (r_{ij} - r_{ii}) + \cdots + (r_{in_y} - r_{ii}) \\ &\quad + (r_{j1} - r_{jj}) + \cdots + (r_{ji} - r_{jj}) + \cdots + (r_{jj} - r_{jj}) + \cdots + (r_{jn_y} - r_{jj})] \\ &\quad - [(r_{j1} - r_{ji}) + \cdots + (r_{ji} - r_{ji}) + \cdots + (r_{jj} - r_{ji}) + \cdots + (r_{jn_y} - r_{ji}) \\ &\quad + (r_{i1} - r_{ij}) + \cdots + (r_{ii} - r_{ij}) + \cdots + (r_{ij} - r_{ij}) + \cdots + (r_{in_y} - r_{ij})] \\ &\quad = n_y(r_{ij} + r_{ji} - r_{ii} - r_{jj}) \end{aligned}$$

Recall that $M_{r,opt}$ is the optimal RDM, implying that the above difference should be greater than or equal to 0, i.e.:

$$n_y(r_{ij} + r_{ji} - r_{ii} - r_{jj}) \geq 0$$

Then,

$$d(\{u_i, y_i\}, \{u_j, y_j\}) = 2\max\{r_{ij}, r_{ji}\} - r_{ii} - r_{jj} \geq r_{ij} + r_{ji} - r_{ii} - r_{jj} \geq 0$$

The second and third properties can be shown easily from Eq.(4.14). □

The main idea behind the proposed clustering procedure is to identify the input/output pairs that are structurally close (i.e. have a small distance) and merge them to form a cluster of input/output pairs that will form a new controller block in a block decentralized configuration. At each level of clustering, the number of controllers is decreased by 1 (although more general formulations are possible), and the clustering procedure stops when there is only one controller (i.e. when we have fully centralized control configuration).

Let us first consider the merging of individual input/output pairs. Note that there is always the possibility that there can be multiple pairs with identical distance. Let us consider the following example RDM:

$$M_{ex} = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \end{array} \begin{array}{c} \begin{array}{ccc} y_1 & y_2 & y_3 \end{array} \\ \begin{bmatrix} 1 & 5 & 5 \\ 4 & 3 & 6 \\ 3 & 4 & 3 \end{bmatrix} \end{array} \tag{4.15}$$

Note that $d(\{u_1, y_1\}, \{u_2, y_2\}) = d(\{u_2, y_2\}, \{u_3, y_3\}) = d(\{u_1, y_1\}, \{u_3, y_3\}) = 6$,

i.e. all three input/output pairs are equally spaced in terms of the distance defined in Eq.(4.14). However, different pairs of input/output pairs (e.g. $\{u_1, y_1\}/\{u_2, y_2\}$ and $\{u_1, y_1\}/\{u_3, y_3\}$) have different structural coupling. In such a case, we need to impose additional criteria to determine the input/output pairs to be merged. A natural first such criterion is to use the sum of the *smallest* differences between off-diagonal and diagonal relative degrees, i.e.:

$$\delta(\{u_i, y_i\}, \{u_j, y_j\}) = 2\min\{r_{ij}, r_{ji}\} - r_{ii} - r_{jj} \tag{4.16}$$

Additionally, we could use the actual values of the off-diagonal elements to further discriminate between different pairs of input/output pairs. To this end, we define:

$$\Delta(\{u_i, y_i\}, \{u_j, y_j\}) = \max\{r_{ij}, r_{ji}\} \tag{4.17}$$

We can now define the distance matrices, D_d, D_δ and D_Δ, whose i, j-th elements are $d(\{u_i, y_i\}, \{u_j, y_j\})$, $\delta(\{u_i, y_i\}, \{u_j, y_j\})$ and $\Delta(\{u_i, y_i\}, \{u_j, y_j\})$, respectively. We can also define a *distance triplet* as (d, δ, Δ). For the example in Eq.(4.15), D_d, D_δ and D_Δ take the form:

$$D_d = \begin{array}{c} \\ \{u_1, y_1\} \\ \{u_2, y_2\} \\ \{u_3, y_3\} \end{array} \begin{array}{c} \begin{array}{ccc} \{u_1, y_1\} & \{u_2, y_2\} & \{u_3, y_3\} \end{array} \\ \begin{bmatrix} 0 & 6 & 6 \\ 6 & 0 & 6 \\ 6 & 6 & 0 \end{bmatrix} \end{array}$$

$$D_\delta = \begin{array}{c} \\ \{u_1, y_1\} \\ \{u_2, y_2\} \\ \{u_3, y_3\} \end{array} \begin{array}{c} \begin{array}{ccc} \{u_1, y_1\} & \{u_2, y_2\} & \{u_3, y_3\} \end{array} \\ \begin{bmatrix} 0 & 4 & 2 \\ 4 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix} \end{array}$$

$$D_\Delta = \begin{array}{c} \\ \{u_1, y_1\} \\ \{u_2, y_2\} \\ \{u_3, y_3\} \end{array} \begin{array}{c} \begin{array}{ccc} \{u_1, y_1\} & \{u_2, y_2\} & \{u_3, y_3\} \end{array} \\ \begin{bmatrix} 0 & 5 & 5 \\ 5 & 0 & 6 \\ 5 & 6 & 0 \end{bmatrix} \end{array}$$

Note that $\delta(\{u_1, y_1\}, \{u_2, y_2\}) > \delta(\{u_2, y_2\}, \{u_3, y_3\}) = \delta(\{u_1, y_1\}, \{u_3, y_3\})$. Furthermore, $\Delta(\{u_2, y_2\}, \{u_3, y_3\}) > \Delta(\{u_1, y_1\}, \{u_3, y_3\})$. Based on these criteria, $\{u_1, y_1\}$ and $\{u_3, y_3\}$, with the distance triplet of (6,2,5), are the input/output pairs to be merged into a cluster.

We can now proceed to discuss the merging of input/output clusters. Let us consider the set of all optimal input/output pairs, $IO_i = \{u_i, y_i\}$, and denote it as IO. Any subset of this set will be an input/output cluster, and be denoted by $C(A)$ where A is the set of indices of the input/output pairs in the cluster (e.g. $C(\{k, l, m\}) = \{IO_k, IO_l, IO_m\}$). Note that a single input/output pair can be considered as a (trivial) cluster by itself (e.g. $C(\{k\}) = \{IO_k\}$). The closeness between input/output clusters can be assessed through an exhaustive search over all possible pairs of input/output pairs from the two clusters, such that their distance is maximized. This leads naturally to the following definition of the distance between clusters $C(A)$ and $C(A')$:

$$d(C(A), C(A')) = \max\{d(IO_k, IO_l) | k \in A, l \in A'\} \tag{4.18}$$

Note that, in this case too, there is the possibility that there can be multiple pairs of input/output clusters with the same distance. These can be further discriminated using the criteria that we defined previously. Specifically, we can define:

$$\delta(C(A), C(A')) = \delta(IO_k, IO_l) \tag{4.19}$$

$$\Delta(C(A), C(A')) = \Delta(IO_k, IO_l) \tag{4.20}$$

where IO_k, IO_l are the input/output pairs identified in Eq.(4.18). The corresponding distance matrices, D_d, D_δ and D_Δ, can then be formed at each step of the procedure and used to determine the clusters to be merged.

For the example in Eq.(4.15), the distances d, δ and Δ between the newly formed cluster, $C(\{1,3\})$, and $C(\{2\})$ can be evaluated yielding the following distance matrices:

$$D_d^{(1)} = \begin{array}{c} \\ C(\{1,3\}) \\ C(\{2\}) \end{array} \begin{array}{c} \begin{array}{cc} C(\{1,3\}) & C(\{2\}) \end{array} \\ \begin{bmatrix} 0 & 6 \\ 6 & 0 \end{bmatrix} \end{array}$$

$$D_\delta^{(1)} = \begin{array}{c} \\ C(\{1,3\}) \\ C(\{2\}) \end{array} \begin{array}{c} \begin{array}{cc} C(\{1,3\}) & C(\{2\}) \end{array} \\ \begin{bmatrix} 0 & 4 \\ 4 & 0 \end{bmatrix} \end{array}$$

$$D_\Delta^{(1)} = \begin{array}{c} \\ C(\{1,3\}) \\ C(\{2\}) \end{array} \begin{array}{c} \begin{array}{cc} C(\{1,3\}) & C(\{2\}) \end{array} \\ \begin{bmatrix} 0 & 5 \\ 5 & 0 \end{bmatrix} \end{array}$$

where the superscript represents the number of updates in the calculation of the distance matrices or equivalently the number of mergings implemented. Note that at the i-th

update, the distance matrices are of dimension $(n_y - i) \times (n_y - i)$, i.e. they contain $n_y - i$ clusters.

The procedure for hierarchical clustering of the input/output pairs described above can be summarized and generalized as follows:

1. At each level of clustering (including the initial optimal decentralized configurations), compute the distance matrices, D_d, D_δ and D_Δ.

2. Find the smallest off-diagonal value in D_d (i.e. the shortest distance between distinct input/output clusters). If there exist multiple pairs of input/output clusters with the smallest value, compare the following values in order:

   (a) δ

   (b) Δ

   Choose the pair with the smallest distance triplet values and merge the clusters to form a new cluster.

3. Repeat steps 1 and 2. Terminate when a single cluster is formed.

Remark 4.1. *There is the possibility that there can be multiple pairs of the input/output clusters with the same distance triplet values. In such case, each choice can be considered separately, proceeding to the subsequent levels to complete the clustering. Multiple dendrograms will be generated to summarize the results in this case.*

Example (continued)**.** Let us consider one of the optimal RDMs obtained for the CSTR

example:

$$M_{r,opt,1} = \begin{array}{c} \\ F \\ F_B \\ F_A \\ Q \end{array} \begin{array}{c} \begin{array}{cccc} M & c_A & c_C & T \end{array} \\ \begin{bmatrix} 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ \infty & 2 & 2 & 1 \end{bmatrix} \end{array}$$

The initial input/output clusters are formed as follows:

$$C(\{1\}) = \{IO_1\} = \{F, M\}$$
$$C(\{2\}) = \{IO_2\} = \{F_B, c_A\}$$
$$C(\{3\}) = \{IO_3\} = \{F_A, c_B\}$$
$$C(\{4\}) = \{IO_4\} = \{Q, T\}$$

The corresponding distance matrices are:

$$D_d = \begin{array}{c} \\ C(\{1\}) \\ C(\{2\}) \\ C(\{3\}) \\ C(\{4\}) \end{array} \begin{array}{c} \begin{array}{cccc} C(\{1\}) & C(\{2\}) & C(\{3\}) & C(\{4\}) \end{array} \\ \begin{bmatrix} 0 & 2 & 2 & \infty \\ 2 & 0 & 0 & 2 \\ 2 & 0 & 0 & 2 \\ \infty & 2 & 2 & 0 \end{bmatrix} \end{array}$$

$$D_\delta = \begin{array}{c} \\ C(\{1\}) \\ C(\{2\}) \\ C(\{3\}) \\ C(\{4\}) \end{array} \begin{array}{c} \begin{array}{cccc} C(\{1\}) & C(\{2\}) & C(\{3\}) & C(\{4\}) \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$D_\Delta = \begin{array}{c} \\ C(\{1\}) \\ C(\{2\}) \\ C(\{3\}) \\ C(\{4\}) \end{array} \begin{array}{c} \begin{array}{cccc} C(\{1\}) & C(\{2\}) & C(\{3\}) & C(\{4\}) \end{array} \\ \begin{bmatrix} 0 & 2 & 2 & \infty \\ 2 & 0 & 1 & 2 \\ 2 & 1 & 0 & 2 \\ \infty & 2 & 2 & 0 \end{bmatrix} \end{array}$$

The smallest off-diagonal element in D_d is $D_d(2,3) = D_d(3,2) = 0$. Thus, we merge $C(\{2\})$ and $C(\{3\})$ to form a new cluster $C(\{2,3\})$. The updated distance matrices are:

$$D_d^{(1)} = \begin{array}{c} \\ C(\{1\}) \\ C(\{2,3\}) \\ C(\{4\}) \end{array} \begin{array}{c} \begin{array}{ccc} C(\{1\}) & C(\{2,3\}) & C(\{4\}) \end{array} \\ \begin{bmatrix} 0 & 2 & \infty \\ 2 & 0 & 2 \\ \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$D_\delta^{(1)} = \begin{array}{c} \\ C(\{1\}) \\ C(\{2,3\}) \\ C(\{4\}) \end{array} \begin{array}{c} \begin{array}{ccc} C(\{1\}) & C(\{2,3\}) & C(\{4\}) \end{array} \\ \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix} \end{array}$$

$$D_{\Delta}^{(1)} = \begin{array}{c} \\ C(\{1\}) \\ C(\{2,3\}) \\ C(\{4\}) \end{array} \begin{array}{c} \begin{array}{ccc} C(\{1\}) & C(\{2,3\}) & C(\{4\}) \end{array} \\ \begin{bmatrix} 0 & 2 & \infty \\ 2 & 0 & 2 \\ \infty & 2 & 0 \end{bmatrix} \end{array}$$

The distance triplets between $C(\{1\})$ and $C(\{2,3\})$ and between $C(\{2,3\})$ and $C(\{4\})$ are equal to (2,0,2). Therefore, we can choose to merge either, and the results can be summarized as dendrograms shown in Figure 4.3, along with the distance triplet values between the clusters to be merged at each level of the dendrogram. Note that these triplets are identical for both dendrograms, indicating similar coupling characteristics. We can now draw a horizontal line at any height to cut the dendrogram, resulting in a block decentralized control configuration. The number of intersections between the horizontal line and the vertical lines represent the number of controllers. For example, the dendrogram shown in Figure 4.3a can be cut at a height between $(2,1,2)$ and $(\infty,2,2)$, to generate a block decentralized control configuration with IO_1, IO_2, IO_3 assigned to one controller, and IO_4 assigned to the other. Note that such a control configuration is reasonable since the first controller addresses the control of material balance variables, while the energy balance variable is assigned to the other controller.

Remark 4.2. *In the case of multiple dendrograms, the distance triplet values at each level are not always the same for all the dendrograms. Different dendrograms can be evaluated noting that larger distance triplet values imply more weakly coupled block decentralized control configurations.*

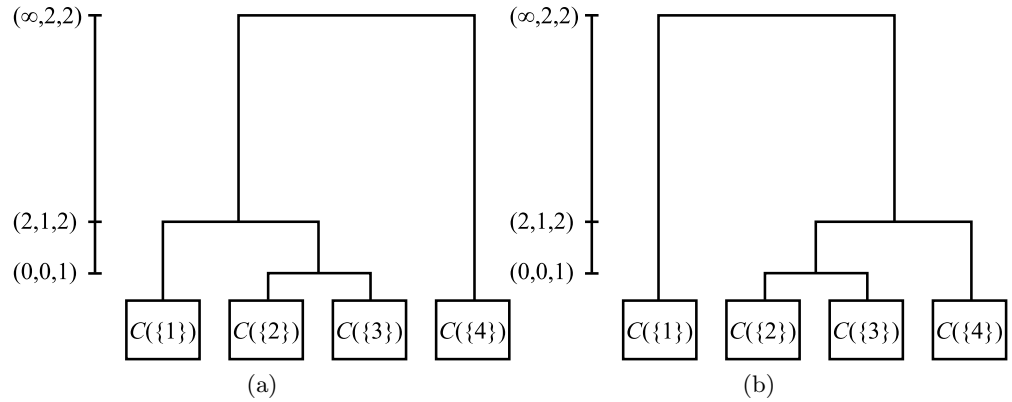


Figure 4.3: Hierarchical clustering results for the simple CSTR example

4.3.3 Algorithm for automation of synthesis procedure

The proposed framework can be automated using the algorithm provided below.

Algorithm: *AutomatedControlConfigurationSynthesis*

1: $\mathcal{EG}$ = ConstructEquationGraph
2: $[M_r, SIOP]$ = Floyd-Warshall($\mathcal{EG}$)
3: $[J_{DC,opt}, P_{opt}]$ = DesignDC(M_r)
4: Add P_{opt} to ODC
5: $P'_{opt} = P_{opt}$
6: **while** $J_{DC} = J_{DC,opt}$ **do**
7: Exclude P'_{opt}
8: $[J_{DC}, P'_{opt}]$ = DesignDC(M_r)
9: **if** $J_{DC} = J_{DC,opt}$ **then**
10: Add P'_{opt} to ODC
11: **end if**

12: **end while**

13: **for** $i = 1$ to Size(ODC) **do**

14: $M_{r,opt}$ =ODC$[i]^T M_r$

15: HC = HierarchicalClustering($M_{r,opt}$)

16: Add HC to DEND

17: **end for**

18: **return** ODC, DEND

The input to the algorithm is the equations which describe the dynamic behavior of the energy/material balances variables. By default, the material/energy flow rates are marked as the manipulated inputs, and the molar holdups, temperatures and concentrations of the process units are marked as the controlled outputs. An additional input can be provided to the algorithm to define specific sets of manipulated inputs and controlled outputs. The outputs of the algorithm are the optimal decentralized control configurations, and the dendrograms.

The subroutine *ConstructEquationGraph* configures the equation graph of the given network, $\mathcal{EG}$, from the input. The RDM of the network (M_r), and the shortest IOPs between all input/output pairs (SIOP) are computed using the Floyd-Warshall algorithm [102]. The Floyd-Warshall algorithm initially sets the length of the path between different nodes to 1 if the nodes are directly connected, otherwise to a very large finite integer, which, in our case, is chosen to be 10001, so that the relative degree of 10^4 represents a relative degree of infinity.

The subroutine *DesignDC* computes the optimal objective function value, $J_{DC,opt}$, and P_{opt} by solving the optimization problem defined by Eqs.(4.7)-(4.11). P_{opt} is added to *ODC*, the set of optimal decentralized control configurations. The *while* loop is added to find the alternative solutions. Then, for each configuration in *ODC*, hierarchy

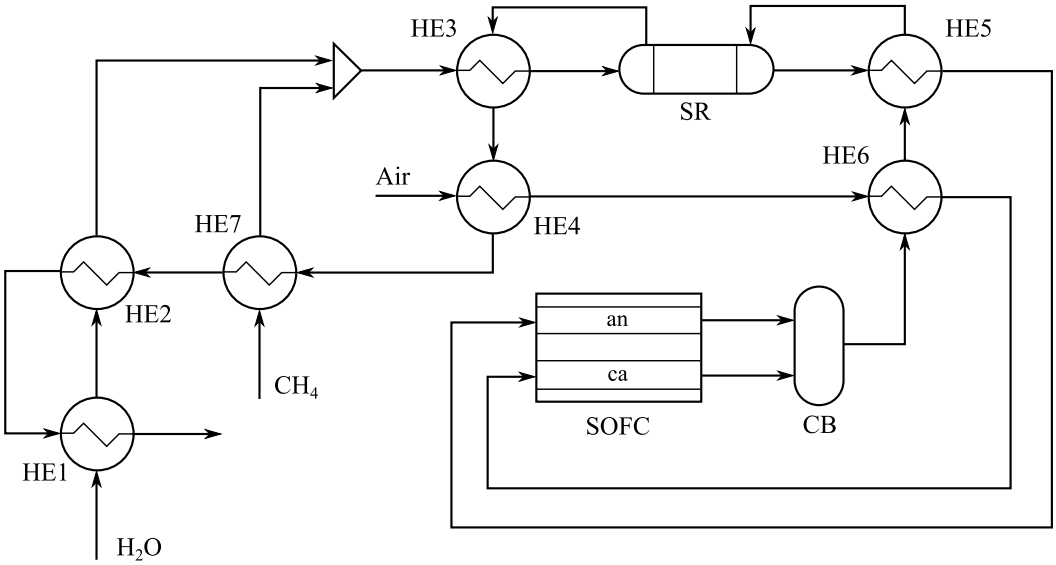


Figure 4.4: Process flow diagram of the energy integrated SOFC system

of input/output clusters *HC*, which is built in a form of dendrogram, is generated using the subroutine *HierarchicalClustering* which automates the procedure proposed in the previous section. *HC* is added to *DEND*, the set of dendrograms.

In what follows, we illustrate the application of the above algorithm using an example process network.

4.4 Case study – energy integrated solid oxide fuel cell (SOFC) system

4.4.1 Process description

Let us consider the energy integrated SOFC system with an external reformer proposed in [80]. The process flow diagram of the system is shown in Figure 4.4. In this system, the feed stream of water is vaporized in HE1 and HE2, and it is mixed with the methane stream which is preheated in HE7. Then, the fuel mixture is heated in HE3 before

entering the steam reformer to facilitate the following reactions:

$$CH_4 + H_2O \rightleftharpoons CO + 3H_2$$
$$CO + H_2O \rightleftharpoons CO_2 + H_2$$

The outlet of the steam reformer is heated in HE5, and supplied into the anode of the fuel cell. The air feed stream is heated in HE4 and HE6, then fed into the cathode of the fuel cell. In the fuel cell, electrochemical potential is generated through the following redox reactions:

$$H_2 + O^{2-} \rightarrow H_2O + 2e^-$$
$$\frac{1}{2}O_2 + 2e^- \rightarrow O^{2-}$$

Both the outlet streams of the fuel cell are then fed into the catalytic burner where methane, carbon monoxide and hydrogen are completely combusted. The outlet stream of the catalytic burner, which is at high temperature, provides energy to the stream reformer and the feed streams as it passes through the jacket of the stream reformer and the heat exchangers.

Important control objectives for this system are:

- control the temperature of the fuel cell (y_1) to prevent material damages while sustaining the ionic conductivity of the electrolyte
- control the fuel utilization of the fuel cell (y_2) to ensure efficient usage of fuel while preventing fuel starvation
- control the fuel cell air inlet temperature (y_3) to enhance the operation of the fuel cell

- control the conversion of the steam reformer (y_4) to generate hydrogen-rich stream
- control the inlet temperature of the steam reformer (y_5) to facilitate the reactions

Also, we have the following potential manipulated inputs:

- the flow rate of air stream entering the fuel cell (u_1)
- the flow rate of fuel stream entering the fuel cell (u_2)
- the duty of HE6
- the jacket flow rate of the steam reformer (u_4)
- the duty of HE3

Since the duty of heat exchangers cannot be manipulated directly, the bypass ratio around the hot channel of HE6 (u_3) and the bypass ratio around the hot channel of HE3 (u_5) will be used as manipulated inputs.

4.4.2 Automated synthesis of control configurations

Let us now apply the developed graph-theoretic framework to this energy integrated SOFC system. The original equation graph consists of 224 nodes and 529 edges, and for brevity, only a part of the original equation graph, which contains the nodes for the inputs and the outputs mentioned previously, and the nodes for the state variables that are included in the IOPs, is shown in Figure 4.5. Table 4.2 shows the list of the nodes

in this graph. The RDM is obtained as:

$$M_r = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{array} \begin{array}{c} \begin{array}{ccccc} y_1 & y_2 & y_3 & y_4 & y_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 2 & 7 & 7 \\ 1 & 1 & 4 & 6 & 6 \\ 3 & 4 & 2 & 5 & 5 \\ 4 & 5 & 5 & 3 & 3 \\ 5 & 6 & 4 & 4 & 2 \end{bmatrix} \end{array}$$

Two optimal solutions are returned by the algorithm:

$$P_{opt,1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_{opt,2} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

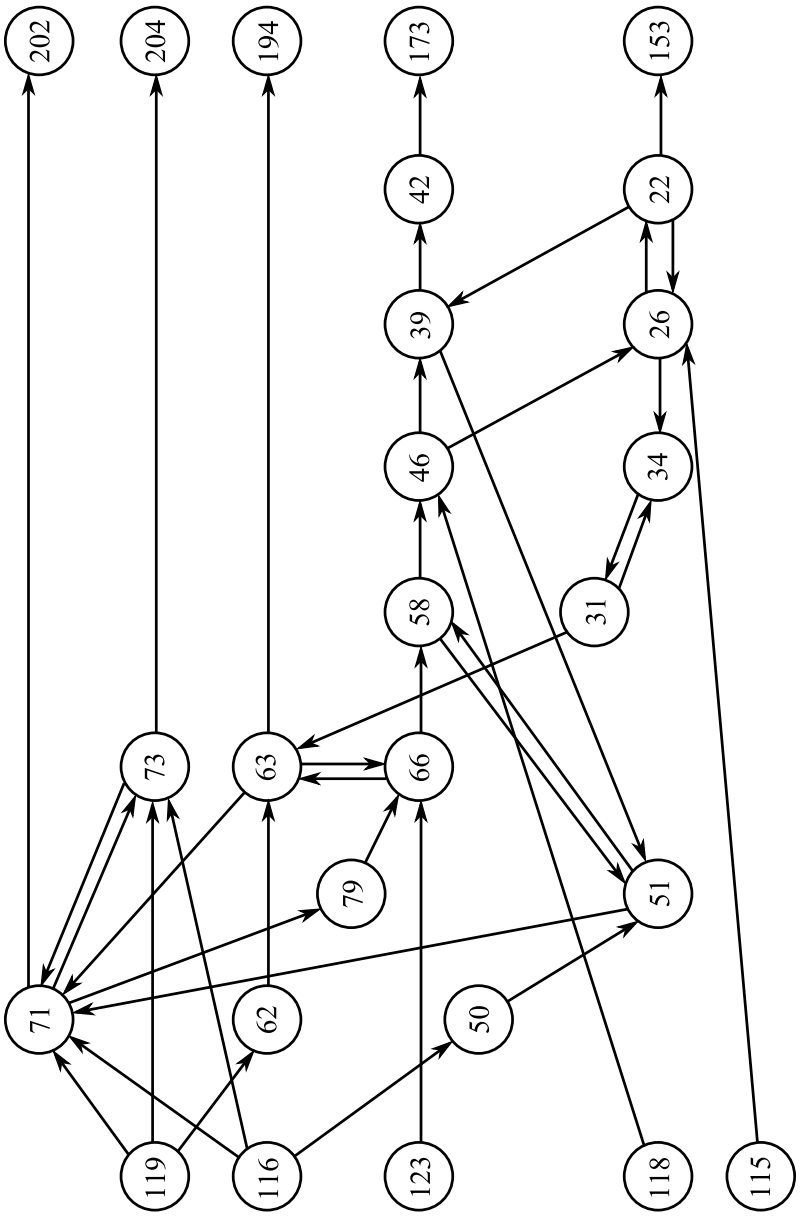


Figure 4.5: Part of the equation graph of the energy integrated SOFC system

which translate into the following optimal RDMs:

$$M_{r,opt,1} = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{array} \begin{array}{c} \begin{array}{ccccc} y_1 & y_2 & y_3 & y_4 & y_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 2 & 7 & 7 \\ 1 & 1 & 4 & 6 & 6 \\ 3 & 4 & 2 & 5 & 5 \\ 4 & 5 & 5 & 3 & 3 \\ 5 & 6 & 4 & 4 & 2 \end{bmatrix} \end{array}$$

$$M_{r,opt,2} = \begin{array}{c} \\ u_2 \\ u_1 \\ u_3 \\ u_4 \\ u_5 \end{array} \begin{array}{c} \begin{array}{ccccc} y_1 & y_2 & y_3 & y_4 & y_5 \end{array} \\ \begin{bmatrix} 1 & 1 & 4 & 6 & 6 \\ 1 & 1 & 2 & 7 & 7 \\ 3 & 4 & 2 & 5 & 5 \\ 4 & 5 & 5 & 3 & 3 \\ 5 & 6 & 4 & 4 & 2 \end{bmatrix} \end{array}$$

From both solutions, we note that, for each row, the diagonal element has the smallest value, implying that each input is paired with the output that it has the most direct effect on. The process flow diagram with the control loops determined by $P_{opt,1}$ is shown in Figure 4.6. In this configuration, the temperature and the fuel utilization of the fuel cell are controlled by manipulating the flow rates of the air and fuel inlet streams, respectively, and the air inlet temperature is controlled by manipulating the bypass ratio around the hot channel of HE6. The jacket flow rate of the steam reformer and the bypass ratio around the hot channel of HE3 are manipulated to control the conversion and the inlet temperature of the steam reformer, respectively. Note that this control configuration matches the control strategy proposed and evaluated via simulations in

Input nodes			
Index	Variable	Index	Variable
115	u_5	119	u_1
116	u_2	123	u_3
118	u_4		
State nodes			
Index	Variable	Index	Variable
22	$T_{HE3,cold}$	51	$T_{HE5,cold}$
26	$T_{HE3,hot}$	58	$T_{HE5,hot}$
31	$T_{HE4,cold}$	62	$M_{HE6,cold}$
34	$T_{HE4,hot}$	63	$T_{HE6,cold}$
39	T_{SR}	66	$T_{HE6,hot}$
42	$c_{CH_4,SR}$	71	T_{SOFC}
46	T_{SRJ}	73	$c_{H_2,SOFC}$
50	$M_{HE5,cold}$	79	T_{CB}
Output nodes			
Index	Variable	Index	Variable
153	y_5	202	y_1
173	y_4	204	y_2
194	y_3		

Table 4.2: Node list of the energy integrated SOFC system equation graph in Figure 4.5

[80]. By switching the manipulated inputs which are used to control the temperature and the fuel utilization of the fuel cell in the previous configuration, we get the control configuration determined by $P_{opt,2}$. This change is reasonable since u_1 and u_2 have the same structural effect on y_1 and y_2 (i.e. $M_r(1,1) = M_r(1,2) = M_r(2,1) = M_r(2,2) = 1$).

For the control configuration determined by $P_{opt,1}$, we have the following initial clusters:

$$C(\{1\}) = IO_1 = \{u_1, y_1\}$$

$$C(\{2\}) = IO_2 = \{u_2, y_2\}$$

$$C(\{3\}) = IO_3 = \{u_3, y_3\}$$

$$C(\{4\}) = IO_4 = \{u_4, y_4\}$$

$$C(\{5\}) = IO_5 = \{u_5, y_5\}$$

and the distance matrix D_d:

$$D_d = \begin{array}{c} \\ C(\{1\}) \\ C(\{2\}) \\ C(\{3\}) \\ C(\{4\}) \\ C(\{5\}) \end{array} \begin{array}{c} \begin{array}{ccccc} C(\{1\}) & C(\{2\}) & C(\{3\}) & C(\{4\}) & C(\{5\}) \end{array} \\ \left[\begin{array}{ccccc} 0 & 0 & 3 & 10 & 11 \\ 0 & 0 & 5 & 8 & 9 \\ 3 & 5 & 0 & 5 & 6 \\ 10 & 8 & 5 & 0 & 3 \\ 11 & 9 & 6 & 3 & 0 \end{array}\right] \end{array}$$

Note that there is only one pair of clusters with the shortest distance ($C(\{1\})$ and $C(\{2\})$), thus the additional criteria are not required at this level. $C(\{1\})$ and $C(\{2\})$ are the clusters to be merged, and D_d is updated accordingly:

$$D_d^{(1)} = \begin{array}{c} \\ C(\{1,2\}) \\ C(\{3\}) \\ C(\{4\}) \\ C(\{5\}) \end{array} \begin{array}{c} \begin{array}{cccc} C(\{1,2\}) & C(\{3\}) & C(\{4\}) & C(\{5\}) \end{array} \\ \left[\begin{array}{cccc} 0 & 5 & 10 & 11 \\ 5 & 0 & 5 & 6 \\ 10 & 5 & 0 & 3 \\ 11 & 6 & 3 & 0 \end{array}\right] \end{array}$$

Again, we have only one pair of clusters with the shortest distance, $C(\{4\})$ and

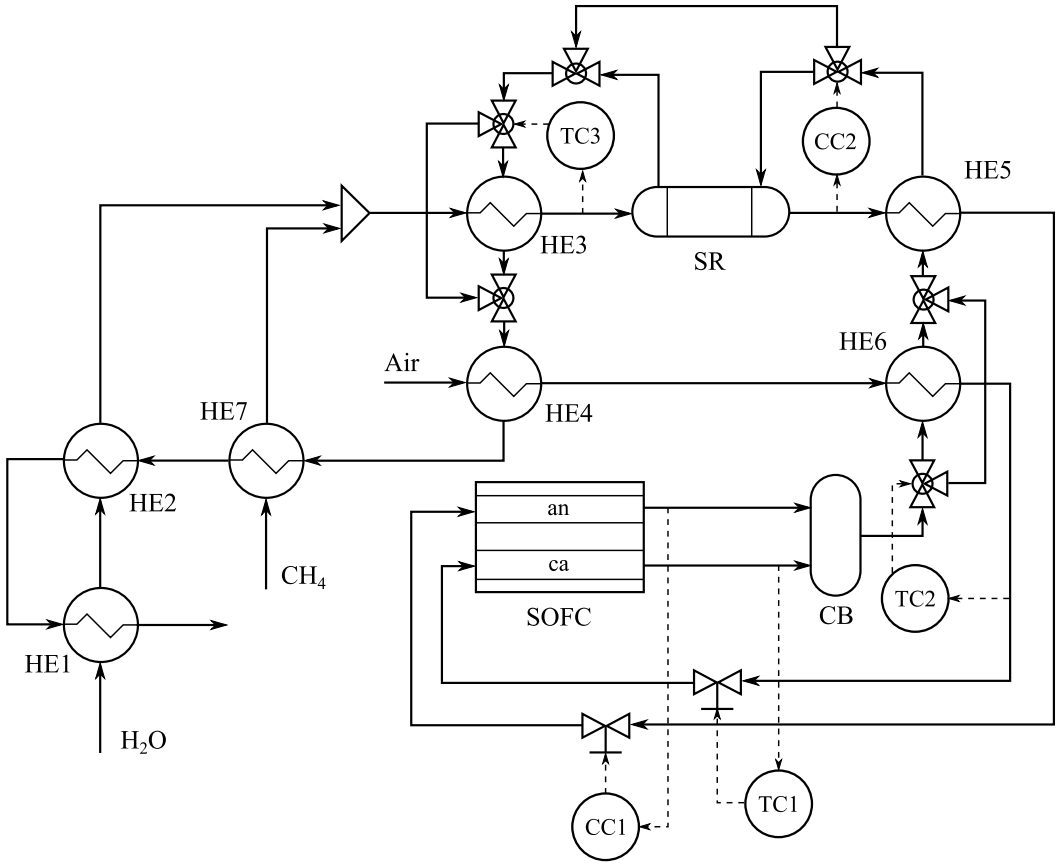


Figure 4.6: Control configuration for solution 1

$C(\{5\})$, which are merged yielding the following distance matrix:

$$D_d^{(2)} = \begin{array}{c} \\ C(\{1,2\}) \\ C(\{3\}) \\ C(\{4,5\}) \end{array} \begin{array}{c} \begin{array}{ccc} C(\{1,2\}) & C(\{3\}) & C(\{4,5\}) \end{array} \\ \begin{bmatrix} 0 & 5 & 11 \\ 5 & 0 & 6 \\ 11 & 6 & 0 \end{bmatrix} \end{array}$$

We finally merge $C(\{1,2\})$ and $C(\{3\})$, and then $C(\{1,2,3\})$ and $C(\{4,5\})$ to complete the clustering. Note that, in this case, the additional criteria are not used as we

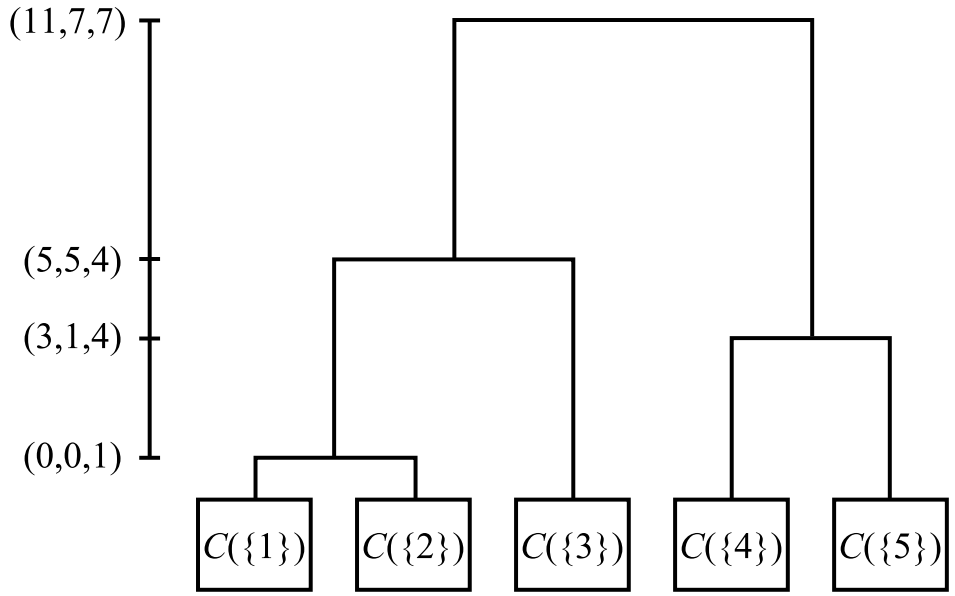


Figure 4.7: Hierarchical clustering result for the energy integrated SOFC system, solution 1

have only one pair of clusters with the shortest distance at each level of clustering. The hierarchical clustering results obtained from $M_{r,opt,1}$ and $M_{r,opt,2}$ are shown in Figures 4.7 and 4.8, respectively.

The dendrograms shown in both figures can be cut at different heights to generate the block decentralized control configurations summarized in Tables 4.3 and 4.4, respectively. We can compare the block decentralized configurations at the same level (e.g. solution 1-1 and solution 2-1) based on the distance triplet values shown in Figures 4.7 and 4.8. In solution 1-1 and solution 2-1, the distance triplets of the closest clusters are (3,1,4). Thus, the block decentralized control configurations in both solutions are equally good from a structural point of view. In solution 1-2 and solution 2-2, the distance triplets of the closest clusters are (5,5,4) and (5,3,4), respectively. Since solution 1-2 has a greater δ, it represents a more weakly coupled control configuration than solution 2-2. Similarly, the control configuration in solution 2-3 is more weakly coupled

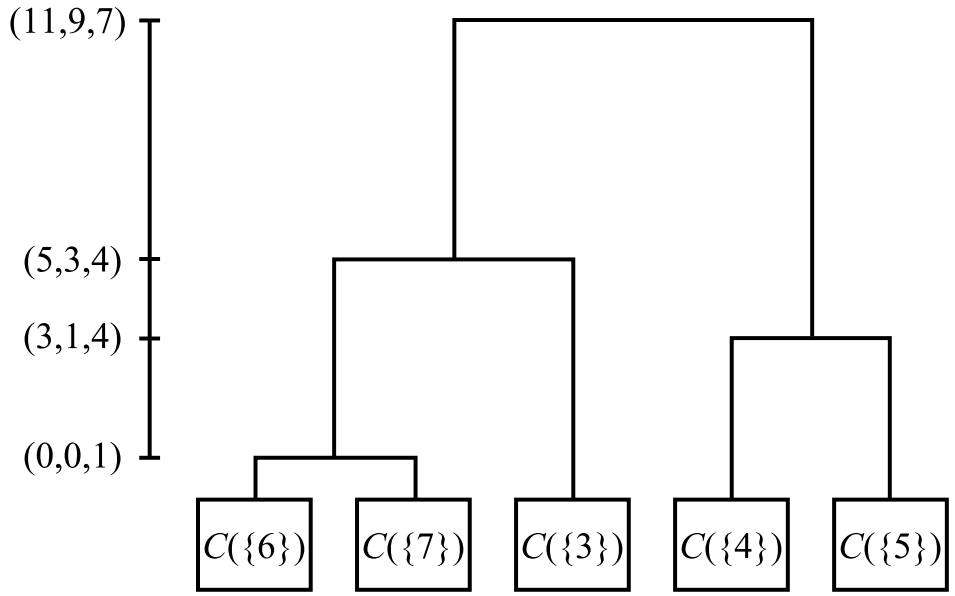


Figure 4.8: Hierarchical clustering result for the energy integrated SOFC system, solution 2 ($C(\{6\}) = \{u_2, y_1\}$, $C(\{7\}) = \{u_1, y_2\}$)

than the one in solution 1-3. Note that, although the additional criteria (i.e. δ and Δ) are not used to complete the clustering, they can be used to further analyze the results.

Noting that the manipulated inputs and the controlled outputs represent physical properties of process units and the flows connecting them, the hierarchical clustering results can also be used to identify subnetworks of closely coupled and decoupled process units. By identifying the process units associated with the inputs and outputs in each cluster, different subnetworks are detected as shown in Figure 4.9. For example, as shown in Figure 4.10, solution 1-3 suggests to address the control problem around the fuel cell (i.e. control of the temperature, the fuel utilization and the air inlet temperature of the fuel cell) using a MIMO controller with the flow rates of the air and fuel inlet streams, and the bypass ratio around the hot channel of HE6 as the manipulated inputs, while the control problem around the steam reformer (i.e. control of the temperature and the conversion of the steam reformer) is addressed by another MIMO controller

Solution	Number of Controller	Input/output clusters
1-1	4	$C(\{1,2\})$, $C(\{3\})$, $C(\{4\})$, $C(\{5\})$
1-2	3	$C(\{1,2\})$, $C(\{3\})$, $C(\{4,5\})$
1-3	2	$C(\{1,2,3\})$, $C(\{4,5\})$

Table 4.3: Block decentralized control configurations generated from the energy integrated SOFC system, solution 1

Solution	Number of Controller	Input/output clusters
2-1	4	$C(\{6,7\})$, $C(\{3\})$, $C(\{4\})$, $C(\{5\})$
2-2	3	$C(\{6,7\})$, $C(\{3\})$, $C(\{4,5\})$
2-3	2	$C(\{3,6,7\})$, $C(\{4,5\})$

Table 4.4: Block decentralized control configurations generated from the energy integrated SOFC system, solution 2

manipulating the jacket inlet flowrate and the bypass ratio around the hot channel of HE3. These choices look reasonable but of course will need to be evaluated using additional criteria as well, and ultimately via simulations.

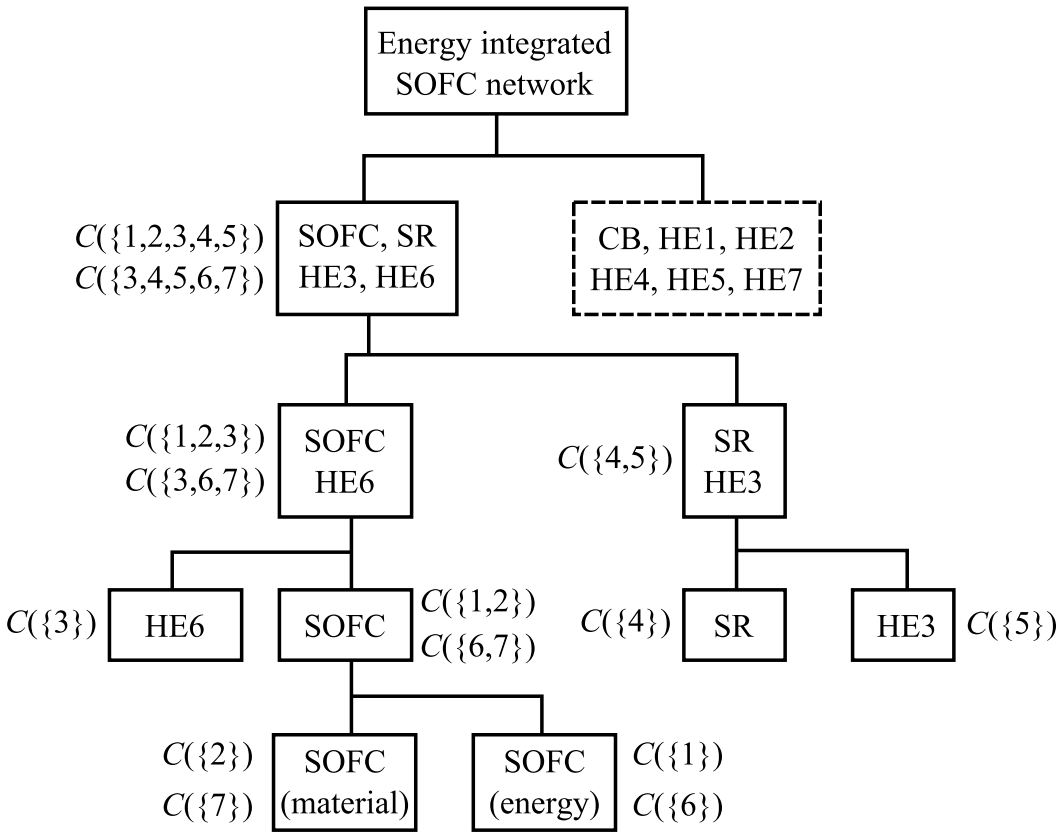


Figure 4.9: Subnetworks of the energy integrated SOFC system

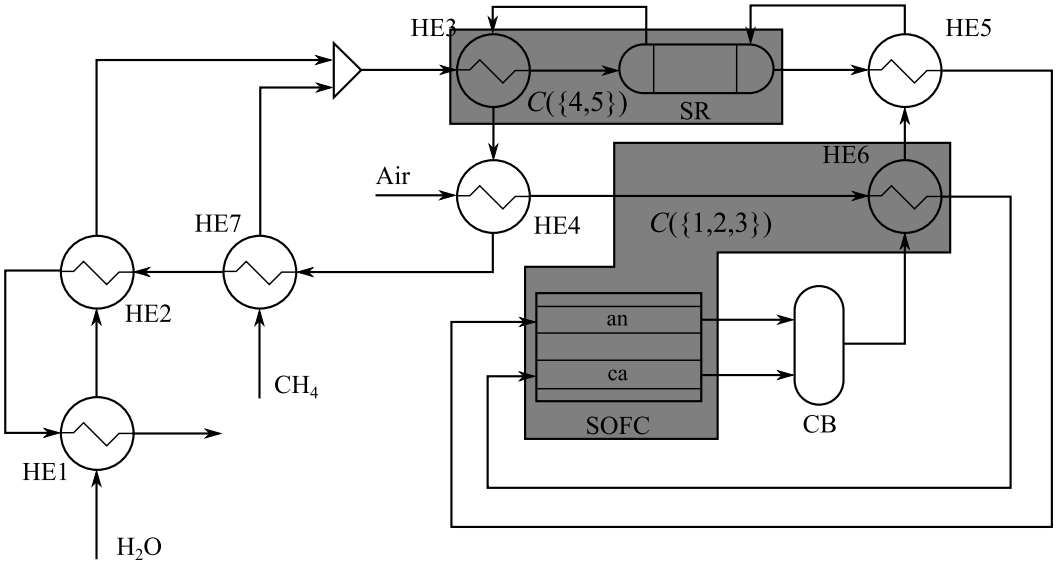


Figure 4.10: Subnetworks controlled by different controllers, solution 1-3

CHAPTER 5

Control-relevant decomposition of process networks based on structural coupling

Complex process networks are ubiquitous in chemical/energy plants, and typically cannot be controlled effectively via purely decentralized control approaches. To this end, in Chapter 4, an agglomerative hierarchical clustering method was developed for the identification of constituent subnetworks such that the components of each subnetwork are strongly connected whereas different subnetworks are weakly connected. In this chapter, a divisive hierarchical clustering method is developed to generate input/output clusters of varying modularity, using relative degree information to define appropriate notions of distance between such clusters and compactness within clusters.

5.1 Relative degree and structural coupling

Let us consider a general state space model of the following form:

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{n_u} g_i(x) u_i \\ y_j &= h_j(x),\ j = 1, \ldots, n_y \end{aligned} \tag{5.1}$$

where $x \in \mathbb{R}^{n_x}$, $u_i, y_j \in \mathbb{R}$ denote the state variables, the input variables and the output variables, respectively. f, g_i are vector fields on $\mathbb{R}^{n_x}$ and h_j are scalar fields on $\mathbb{R}^{n_x}$.

The relative degree of the output y_j with respect to u_i, r_{ij}, is defined as the smallest integer which satisfies [101]:

$$\mathcal{L}_{g_i} \mathcal{L}_f^{r_{ij}-1} h_j(x) \not\equiv 0 \tag{5.2}$$

where $\mathcal{L}$ represents the Lie derivative:

$$\mathcal{L}_f h(x) = \frac{\partial h(x)}{\partial x} f(x)$$

In [69], it was shown that the relative degree is a meaningful measure of how direct the effect of an input is on an output, as it represents the number of integrations required for an input to affect an output. It was also shown that the relative degree is a measure of physical closeness between an input and an output as captured by the sluggishness of the input/output response.

In what follows, we develop a divisive hierarchical clustering framework to decompose a large scale, interconnected system based on structural coupling. The goal is to generate a hierarchy of block decentralized control configurations with increasing number of controllers, whereby in each level we decompose a controller block with the least

favorable structural coupling characteristics in a sense that we will describe shortly.

5.2 Control configuration synthesis by network decomposition

5.2.1 Divisive hierarchical clustering of process networks based on relative degrees

Let us consider a square system with n inputs and n outputs, whose dynamics can be described by Eq.(5.1), and the corresponding relative degree matrix (RDM), M_r, whose i,j-th element is the relative degree between u_i and y_j:

$$M_r = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{bmatrix}$$

The problem of block decentralized control configuration design for such a system corresponds to the rearrangement of the above RDM in a partitioned form as follows:

$$M_r' = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1n_c} \\ R_{21} & R_{22} & \cdots & R_{2n_c} \\ \vdots & \vdots & \ddots & \vdots \\ R_{n_c1} & R_{n_c2} & \cdots & R_{n_cn_c} \end{bmatrix}$$

where n_c ($\leq n$) represents the number of controllers, and R_{ij} represent the relative degree matrices whose elements are the relative degrees between the inputs distributed to the controller i and the outputs distributed to the controller j.

One of the major challenges for such a problem is the absence of a priori knowledge on the structure and the number of subnetworks that are weakly interacting with one another, and thus can be effectively controlled by separate controllers. To this end, a hierarchical clustering method can be adopted to generate block decentralized control configuration candidates of different modularity.

Let us define an input/output cluster (corresponding to a controller), C, as a group of input variables and output variables. We assume that all input/output clusters are square, i.e. the same number of inputs and outputs are distributed to each cluster. The size of the cluster, $N(C)$, is defined as the number of inputs (or outputs) that are distributed to the cluster. We also define $M_r(C)$ as a submatrix of M_r, which contains the relative degrees between inputs and outputs in C.

A divisive clustering of C can then be formulated as a bipartition into clusters C_m and C_n, such that $C_m \cup C_n = C$, $C_m \cap C_n = \varnothing$. This leads to a rearrangement of $M_r(C)$ in the following partitioned form:

$$\begin{bmatrix} M_r(C_m) & R_{mn} \\ R_{nm} & M_r(C_n) \end{bmatrix} \tag{5.3}$$

The above bipartition (i.e. block decentralized control configuration) can be evaluated based on the *compactness* of each cluster (i.e. the physical closeness among the inputs and the outputs of each controller), captured by the relative degrees in the diagonal matrices, and the *closeness* between the clusters (i.e. the interaction between the controllers), captured by the relative degrees in the off-diagonal matrices.

Different notions of compactness can be defined as a function of the relative degrees in the cluster. For example, the compactness of the cluster C, $\gamma(C)$, can be captured

by the interaction of the most weakly coupled input/output pair in C, i.e.:

$$\gamma(C) = \frac{1}{\max\{r_{ij} | r_{ij} \in M_r(C)\}} \tag{5.4}$$

or, by the average interaction among the inputs and the outputs in C, i.e.:

$$\gamma(C) = \frac{N(C)^2}{\sum_{M_r(C)} r_{ij}} \tag{5.5}$$

We then define the compactness of the bipartition, $\gamma(C_m, C_n)$, as follows:

$$\gamma(C_m, C_n) = \min\{\gamma(C_m), \gamma(C_n)\} \tag{5.6}$$

i.e. the compactness of bipartition is characterized by the looser cluster.

Similarly, different notions of closeness can be defined. For example, the closeness between the clusters C_m and C_n, $\Gamma(C_m, C_n)$, can be captured by the interaction of the most strongly coupled input and output from different clusters, i.e.:

$$\Gamma(C_m, C_n) = \min\{r_{ij} | r_{ij} \in (R_{mn} \cup R_{nm})\} \tag{5.7}$$

or, by the average interaction, i.e.:

$$\Gamma(C_m, C_n) = \frac{\sum_{R_{mn}} r_{ij} + \sum_{R_{nm}} r_{ij}}{2N(C_m)N(C_n)} \tag{5.8}$$

Remark 5.1. *The distance between the clusters in Eq.(4.18) and the closeness between the clusters in Eq.(5.7) are consistent with the linkage criteria which are commonly used in the hierarchical clustering. They correspond to the complete linkage criterion and the single linkage criterion, respectively. The complete linkage criterion is used in Eq.(4.18) so that the distance defined therein can also be used as a measure of compactness of the*

new cluster to be formed through merging, while the single linkage criterion is used in Eq.(5.7) so that the new clusters to be formed through decomposition are separated as much as possible.

The problem of the optimal bipartition of C corresponds to an exhaustive search over all bipartitions of C such that the following *decentrality* measure is maximized:

$$\Gamma(C_m, C_n) \cdot \gamma(C_m, C_n) \tag{5.9}$$

Remark 5.2. *The above definitions of the compactness and the closeness are not exhaustive, making the proposed divisive clustering method quite flexible.* $\gamma(C_m, C_n)$ *and* $\Gamma(C_m, C_n)$ *have to be selected carefully to obtain block decentralized control configurations with desired structural coupling characteristics. For example, if the interaction between every possible pair in the cluster needs to be taken into account, the compactness measure defined in Eq.(5.5) can be used.*

In what follows, a generic mathematical optimization problem to identify the optimal bipartition is formulated.

5.2.2 Optimization problem formulation

We now proceed to formulate an optimization problem which can be used to maximize the decentrality measure in Eq.(5.9). Let us define two different sets of binary variables which will enable us to distinguish the relative degrees in different submatrices in Eq.(5.3) without rearrangement:

$$U(l,i) = \begin{cases} 1, & \text{if } u_i \text{ is distributed to the cluster } l \\ 0, & \text{otherwise} \end{cases} \tag{5.10}$$

$$Y(l,j) = \begin{cases} 1, & \text{if } y_j \text{ is distributed to the cluster } l \\ 0, & \text{otherwise} \end{cases} \tag{5.11}$$

Using these binary variables, the sets of the relative degrees $\mathbf{r_{mm}}$, $\mathbf{r_{mn}}$, $\mathbf{r_{nm}}$ and $\mathbf{r_{nn}}$ in $M_r(C_m)$, R_{mn}, R_{nm} and $M_r(C_n)$, respectively, can be defined as:

$$\mathbf{r_{mm}} = \{r_{ij} \in M_r(C) | U(m,i) \cdot Y(m,j) = 1\}$$
$$\mathbf{r_{mn}} = \{r_{ij} \in M_r(C) | U(m,i) \cdot Y(n,j) = 1\}$$
$$\mathbf{r_{nm}} = \{r_{ij} \in M_r(C) | U(n,i) \cdot Y(m,j) = 1\}$$
$$\mathbf{r_{nn}} = \{r_{ij} \in M_r(C) | U(n,i) \cdot Y(n,j) = 1\}$$

The optimization problem for identifying the bipartition of C can be stated as follows:

$$\text{maximize } \Gamma(C_m, C_n) \cdot \gamma(C_m, C_n) \tag{5.12}$$

subject to the following constraints:

- Any input or output can be distributed to one cluster at most:

$$\sum_l U(l,i) \leq 1, \ \forall i \tag{5.13}$$

$$\sum_l Y(l,j) \leq 1, \ \forall j \tag{5.14}$$

- The resulting clusters need to represent square controller blocks, i.e. the same

number of inputs and outputs needs to be distributed to each cluster:

$$\sum_i U(l,i) = \sum_j Y(l,j),\ l = m,n \tag{5.15}$$

- All outputs need to be distributed:

$$\sum_j \sum_l Y(l,j) = N(C) \tag{5.16}$$

The above constraint, with the constraint (5.15), also ensures that all inputs are distributed.

- Finally, the following constraint is introduced to make sure that decomposition occurs, i.e. all the outputs in C cannot be distributed to one cluster:

$$\sum_j Y(l,j) \leq N(C) - 1,\ l = m,n \tag{5.17}$$

Note that the above formulation (Eqs.(5.12)-(5.17)) is an integer nonlinear programming problem with $U(l,i)$ and $Y(l,j)$ as the decision variables. This optimization problem can be solved within the environment of GAMS using the global optimizer BARON, which is able to find alternative optimal solutions. The outputs of the optimization problem are the matrices U and Y, which can be used to identify the partitioned matrix in the form of Eq.(5.3).

The divisive hierarchical clustering procedure can be summarized as follows:

1. At each level of clustering, k, find the input/output cluster with the smallest compactness measure. Let this cluster be C_k.

2. Find the optimal bipartition of C_k by solving the optimization problem defined

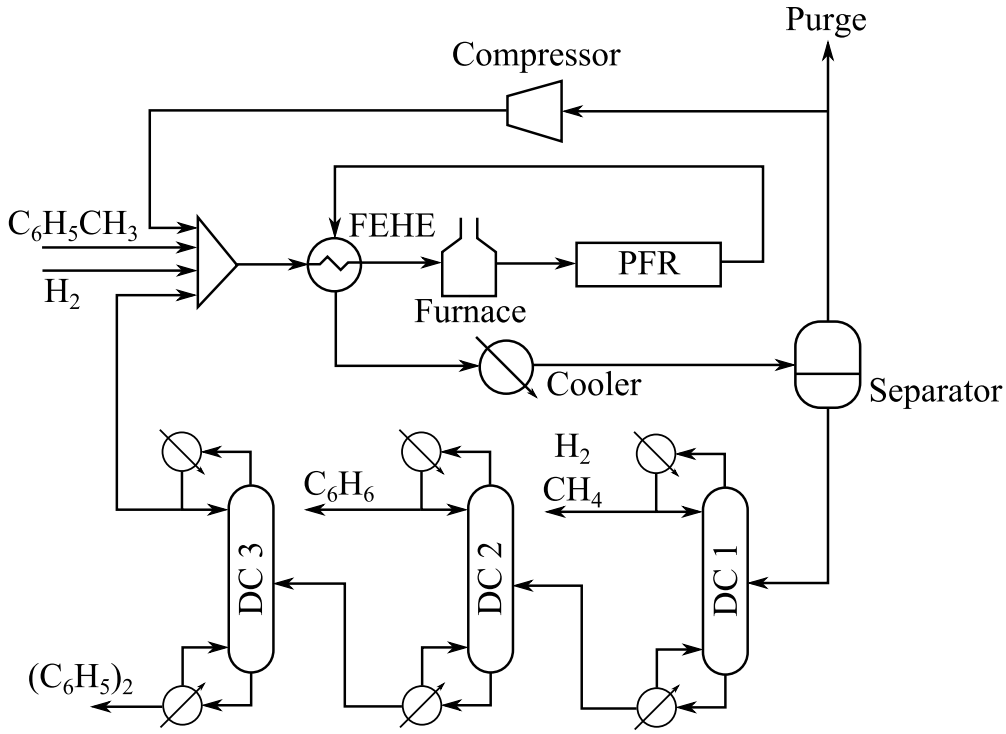


Figure 5.1: Toluene hydrodealkylation process

by Eq.(5.12)-Eq.(5.17).

3. Repeat steps 1 and 2. Terminate if all the clusters are of size 1 (i.e. fully decentralized control configuration).

In what follows, using an example chemical process network, we illustrate the application of the proposed procedure.

5.3 Case study – Toluene hydrodealkylation process

Let us consider the toluene hydrodealkylation process shown in Figure 5.1. The detailed description of the process can be found in Chapter 2. Table 5.1 tabulates the control objectives and the manipulated inputs available in this process, adopted from

the previous studies [5, 105–107]. The RDM can be obtained as:

$$M_r = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \\ u_{11} \\ u_{12} \\ u_{13} \end{array} \begin{array}{c} \begin{array}{ccccccccccccc} y_1 & y_2 & y_3 & y_4 & y_5 & y_6 & y_7 & y_8 & y_9 & y_{10} & y_{11} & y_{12} & y_{13} \end{array} \\ \left[\begin{array}{ccccccccccccc} 5 & 3 & 2 & 2 & 4 & 2 & 4 & 5 & 5 & 7 & 7 & 9 & 9 \\ 4 & 6 & 3 & 5 & 3 & 5 & 1 & 2 & 2 & 4 & 4 & 6 & 6 \\ 4 & 6 & 3 & 5 & 1 & 5 & 2 & 4 & 4 & 6 & 6 & 8 & 8 \\ 4 & 4 & 1 & 3 & 3 & 3 & 1 & 4 & 4 & 6 & 6 & 8 & 8 \\ 1 & 3 & 5 & 4 & 6 & 2 & 6 & 7 & 7 & 9 & 9 & 11 & 11 \\ 5 & 6 & 3 & 1 & 3 & 5 & 2 & 4 & 4 & 6 & 6 & 8 & 8 \\ 8 & 10 & 12 & 11 & 13 & 9 & 13 & 1 & 3 & 5 & 5 & 7 & 7 \\ 6 & 8 & 10 & 9 & 11 & 7 & 11 & 12 & 12 & 1 & 3 & 5 & 5 \\ 2 & 4 & 6 & 5 & 7 & 3 & 7 & 8 & 8 & 10 & 10 & 1 & 3 \\ 9 & 1 & 3 & 5 & 7 & 5 & 7 & 9 & 9 & 11 & 11 & 13 & 13 \\ 6 & 8 & 10 & 9 & 11 & 7 & 11 & 3 & 1 & 3 & 3 & 5 & 5 \\ 4 & 6 & 8 & 7 & 9 & 5 & 9 & 10 & 10 & 3 & 1 & 3 & 3 \\ 4 & 6 & 8 & 7 & 9 & 5 & 9 & 10 & 10 & 12 & 12 & 3 & 1 \end{array}\right] \end{array}$$

We first apply the agglomerative hierarchical clustering method developed in Chapter 4. Then, the divisive clustering method developed in the previous section is applied, and the results from both methods are compared.

	Manpulated inputs
u_1	Bypass ratio around the hot channel of the FEHE
u_2	Separator liquid outlet flowrate
u_3	Purge flowrate
u_4	Separator quencher outlet flowrate
u_5	Toluene feed flowrate
u_6	Cooler duty
u_7	S-Condenser duty
u_8	P-Condenser duty
u_9	R-Condenser duty
u_{10}	Furnace duty
u_{11}	S-Reboiler duty
u_{12}	P-Reboiler duty
u_{13}	R-Reboiler duty
	Controlled outputs
y_1	H_2 to aromatic compounds ratio at the mixer outlet
y_2	Furnace outlet temperature
y_3	Quencher outlet temperature
y_4	Cooler outlet temperature
y_5	Recycle gas methane composition
y_6	Furnace inlet temperature
y_7	Separator liquid phase holdup
y_8	Stabilizer distillate benzene composition
y_9	Stabilizer bottom methane composition
y_{10}	Product column distillate benzene composition
y_{11}	Product column bottom benzene composition
y_{12}	Recycle column distillate diphenyl composition
y_{13}	Recycle column bottom toluene composition

Table 5.1: Manipulated inputs and controlled ouputs for the HDA process

5.3.1 Agglomerative clustering

The optimization problem defined by Eq.(4.7)-Eq.(4.11) is solved for this process to obtain the following optimal solution:

$$P_{opt} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

which translates into the following optimal RDM:

$$M_{r,opt} = \begin{array}{c} \\ u_5 \\ u_{10} \\ u_4 \\ u_6 \\ u_3 \\ u_1 \\ u_2 \\ u_7 \\ u_{11} \\ u_8 \\ u_{12} \\ u_9 \\ u_{13} \end{array} \begin{array}{c} \begin{array}{cccccccccccccc} y_1 & y_2 & y_3 & y_4 & y_5 & y_6 & y_7 & y_8 & y_9 & y_{10} & y_{11} & y_{12} & y_{13} \end{array} \\ \left[\begin{array}{cccccccccccccc} 1 & 3 & 5 & 4 & 6 & 2 & 6 & 7 & 7 & 9 & 9 & 11 & 11 \\ 9 & 1 & 3 & 5 & 7 & 5 & 7 & 9 & 9 & 11 & 11 & 13 & 13 \\ 4 & 4 & 1 & 3 & 3 & 3 & 1 & 4 & 4 & 6 & 6 & 8 & 8 \\ 5 & 6 & 3 & 1 & 3 & 5 & 2 & 4 & 4 & 6 & 6 & 8 & 8 \\ 4 & 6 & 3 & 5 & 1 & 5 & 2 & 4 & 4 & 6 & 6 & 8 & 8 \\ 5 & 3 & 2 & 2 & 4 & 2 & 4 & 5 & 5 & 7 & 7 & 9 & 9 \\ 4 & 6 & 3 & 5 & 3 & 5 & 1 & 2 & 2 & 4 & 4 & 6 & 6 \\ 8 & 10 & 12 & 11 & 13 & 9 & 13 & 1 & 3 & 5 & 5 & 7 & 7 \\ 6 & 8 & 10 & 9 & 11 & 7 & 11 & 3 & 1 & 3 & 3 & 5 & 5 \\ 6 & 8 & 10 & 9 & 11 & 7 & 11 & 12 & 12 & 1 & 3 & 5 & 5 \\ 4 & 6 & 8 & 7 & 9 & 5 & 9 & 10 & 10 & 3 & 1 & 3 & 3 \\ 2 & 4 & 6 & 5 & 7 & 3 & 7 & 8 & 8 & 10 & 10 & 1 & 3 \\ 4 & 6 & 8 & 7 & 9 & 5 & 9 & 10 & 10 & 12 & 12 & 3 & 1 \end{array}\right] \end{array}$$

The initial clusters are:

$$C_1 = \{u_5, y_1\},\ C_2 = \{u_{10}, y_2\},\ C_3 = \{u_4, y_3\},\ C_4 = \{u_6, y_4\},\ C_5 = \{u_3, y_5\}$$

$$C_6 = \{u_1, y_6\},\ C_7 = \{u_2, y_7\},\ C_8 = \{u_7, y_8\},\ C_9 = \{u_{11}, y_9\}$$

$$C_{10} = \{u_8, y_{10}\},\ C_{11} = \{u_{12}, y_{11}\},\ C_{12} = \{u_9, y_{12}\},\ C_{13} = \{u_{13}, y_{13}\}$$

Applying the agglomerative hierarchical clustering method results in the unique dendrogram shown in Figure 5.2. The detailed results of the clustering are not shown here for brevity. Different clusters, identified during the clustering, and their members

Cluster	Members	Cluster	Members
C_{14}	C_3+C_6	C_{20}	C_3+C_4+C_5+C_6+C_7
C_{15}	C_5+C_7	C_{21}	C_1+C_3+C_4+C_5+C_6+C_7
C_{16}	C_8+C_9	C_{22}	C_1+C_2+C_3+C_4+C_5+C_6+C_7
C_{17}	C_{10}+C_{11}	C_{23}	C_8+C_9+C_{12}+C_{13}
C_{18}	C_{12}+C_{13}	C_{24}	C_1+C_2+C_3+C_4+C_5+C_6+C_7+C_{10}+C_{11}
C_{19}	C_3+C_4+C_6		

Table 5.2: Cluster list for the HDA process: agglomerative clustering

are summarized in Table 5.2. Block decentralized control configurations can be obtained by drawing horizontal lines between the adjacent levels of the dendrogram shown in Figure 5.2, and are summarized in Table 5.3. For example, in configuration 3, we have four MIMO controllers which address the control objectives associated with the reaction-recycle network, the stabilizer, the product column and the recycle column separately.

Note that, in configuration 1, the product column, which is in the middle of the distillation column network, is coupled with the reaction-recycle network, while the other two columns, which are separated by the product column, are coupled with each other. Such a subnetwork structure is predicted through the clustering, since different subnetworks are merged on the basis of the *weakest* interaction between them. The product column is not coupled with the recycle column since, in order for the control action in the recycle column to affect the control of the product column, it has to propagate through the reaction-recycle network and the stabilizer column, implying that the distance between the two columns (captured by the distance metric defined in 4.18) is large. Similar argument can be made for the product column and the stabilizer column. Motivated by this, in the following divisive clustering analysis, we define the compactness measure on the basis of the *overall* interactions between different clusters.

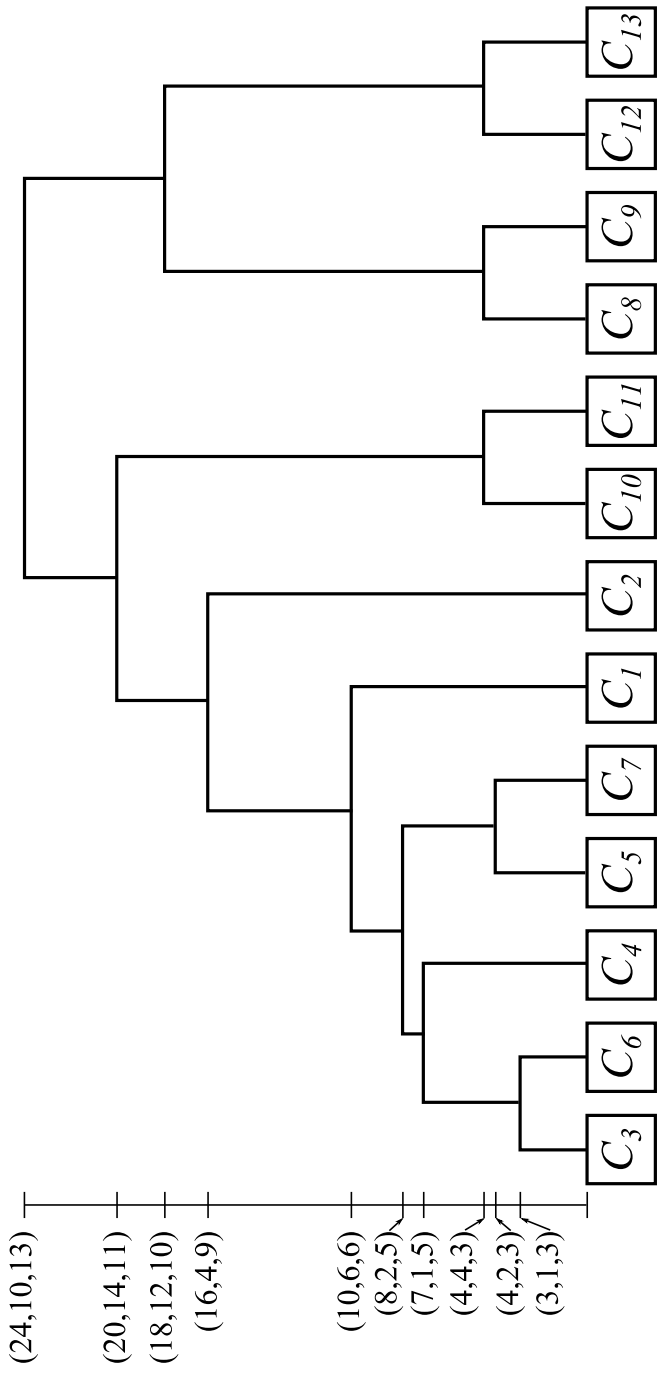


Figure 5.2: Hierarchical clustering result for the HDA process: agglomerative clustering

Configuration	Conroller blocks
1	C_{23}, C_{24}
2	C_{17}, C_{22}, C_{23}
3	C_{16}, C_{17}, C_{18}, C_{22}
4	C_2, C_{16}, C_{17}, C_{18}, C_{21}
5	C_1, C_2, C_{16}, C_{17}, C_{18}, C_{20}
6	C_1, C_2, C_{15}, C_{16}, C_{17}, C_{18}, C_{19}
7	C_1, C_2, C_4, C_{14}, C_{15}, C_{16}, C_{17}, C_{18}
8	C_1, C_2, C_4, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}, C_{15}
9	C_1, C_2, C_4, C_5, C_7, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}

Table 5.3: Block decentralized control configurations for the HDA process: agglomerative clustering

5.3.2 Divisive clustering

In this analysis, we use the compactness measure and the closeness measure defined in Eq.(5.5) and Eq.(5.7), respectively.

The optimization problem defined by Eq.(5.12)-Eq.(5.17) can be solved iteratively to decompose C_0 (the cluster with all the inputs and the outputs).

Clustering level 1: $C_0 \rightarrow C_1'$ ($\gamma = 0.19$), C_2' ($\gamma = 0.22$)

$$M_r(C_1') = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_9 \\ u_{10} \\ u_{11} \end{array} \begin{array}{c} \begin{array}{cccccccccc} y_1 & y_2 & y_3 & y_4 & y_5 & y_6 & y_7 & y_8 & y_9 & y_{12} \end{array} \\ \left[\begin{array}{cccccccccc} 5 & 3 & 2 & 2 & 4 & 2 & 4 & 5 & 5 & 9 \\ 4 & 6 & 3 & 5 & 3 & 5 & 1 & 2 & 2 & 6 \\ 4 & 6 & 3 & 5 & 1 & 5 & 2 & 4 & 4 & 8 \\ 4 & 4 & 1 & 3 & 3 & 3 & 1 & 4 & 4 & 8 \\ 1 & 3 & 5 & 4 & 6 & 2 & 6 & 7 & 7 & 11 \\ 5 & 6 & 3 & 1 & 3 & 5 & 2 & 4 & 4 & 8 \\ 8 & 10 & 12 & 11 & 13 & 9 & 13 & 1 & 3 & 7 \\ 2 & 4 & 6 & 5 & 7 & 3 & 7 & 8 & 8 & 1 \\ 9 & 1 & 3 & 5 & 7 & 5 & 7 & 9 & 9 & 13 \\ 6 & 8 & 10 & 9 & 11 & 7 & 11 & 3 & 1 & 5 \end{array}\right] \end{array}$$

$$M_r(C_2') = \begin{array}{c} \\ u_8 \\ u_{12} \\ u_{13} \end{array} \begin{array}{c} \begin{array}{ccc} y_{10} & y_{11} & y_{13} \end{array} \\ \left[\begin{array}{ccc} 1 & 3 & 5 \\ 3 & 1 & 3 \\ 12 & 12 & 1 \end{array}\right] \end{array}$$

Clustering level 2: $C_1' \rightarrow C_3'$ ($\gamma = 0.199$), C_4' ($\gamma = 1$)

$$M_r(C_3') = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_9 \\ u_{11} \end{array} \begin{array}{c} \begin{array}{ccccccccc} y_1 & y_3 & y_4 & y_5 & y_6 & y_7 & y_8 & y_9 & y_{12} \end{array} \\ \begin{bmatrix} 5 & 2 & 2 & 4 & 2 & 4 & 5 & 5 & 9 \\ 4 & 3 & 5 & 3 & 5 & 1 & 2 & 2 & 6 \\ 4 & 3 & 5 & 1 & 5 & 2 & 4 & 4 & 8 \\ 4 & 1 & 3 & 3 & 3 & 1 & 4 & 4 & 8 \\ 1 & 5 & 4 & 6 & 2 & 6 & 7 & 7 & 11 \\ 5 & 3 & 1 & 3 & 5 & 2 & 4 & 4 & 8 \\ 8 & 12 & 11 & 13 & 9 & 13 & 1 & 3 & 7 \\ 2 & 6 & 5 & 7 & 3 & 7 & 8 & 8 & 1 \\ 6 & 10 & 9 & 11 & 7 & 11 & 3 & 1 & 5 \end{bmatrix} \end{array}$$

$$M_r(C_4') = \begin{array}{c} \\ u_{10} \end{array} \begin{array}{c} \begin{array}{c} y_2 \end{array} \\ \begin{bmatrix} 1 \end{bmatrix} \end{array}$$

Clustering level 3: $C_3' \rightarrow C_5'$ ($\gamma = 0.305$), C_6' ($\gamma = 0.243$)

$$M_r(C_5') = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{array} \begin{array}{c} \begin{array}{cccccc} y_1 & y_3 & y_4 & y_5 & y_6 & y_7 \end{array} \\ \begin{bmatrix} 5 & 2 & 2 & 4 & 2 & 4 \\ 4 & 3 & 5 & 3 & 5 & 1 \\ 4 & 3 & 5 & 1 & 5 & 2 \\ 4 & 1 & 3 & 3 & 3 & 1 \\ 1 & 5 & 4 & 6 & 2 & 6 \\ 5 & 3 & 1 & 3 & 5 & 2 \end{bmatrix} \end{array}, \quad M_r(C_6') = \begin{array}{c} \\ u_7 \\ u_9 \\ u_{11} \end{array} \begin{array}{c} \begin{array}{ccc} y_8 & y_9 & y_{12} \end{array} \\ \begin{bmatrix} 1 & 3 & 7 \\ 8 & 8 & 1 \\ 3 & 1 & 5 \end{bmatrix} \end{array}$$

Clustering level 4: $C'_2 \rightarrow C'_7$ ($\gamma = 0.5$), C'_8 ($\gamma = 1$)

$$M_r(C'_7) = \begin{array}{c} \\ u_8 \\ u_{12} \end{array} \begin{array}{c} \begin{array}{cc} y_{10} & y_{11} \end{array} \\ \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix} \end{array}, \; M_r(C'_8) = \begin{array}{c} \\ u_{13} \end{array} \begin{array}{c} y_{13} \\ \begin{bmatrix} 1 \end{bmatrix} \end{array}$$

Clustering level 5: $C'_6 \rightarrow C'_9$ ($\gamma = 0.5$), C'_{10} ($\gamma = 1$)

$$M_r(C'_9) = \begin{array}{c} \\ u_7 \\ u_{11} \end{array} \begin{array}{c} \begin{array}{cc} y_8 & y_9 \end{array} \\ \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix} \end{array}, \; M_r(C'_{10}) = \begin{array}{c} \\ u_9 \end{array} \begin{array}{c} y_{12} \\ \begin{bmatrix} 1 \end{bmatrix} \end{array}$$

Clustering level 6: $C'_5 \rightarrow C'_{11}$ ($\gamma = 0.333$), C'_{12} ($\gamma = 0.5$)

$$M_r(C'_{11}) = \begin{array}{c} \\ u_1 \\ u_5 \\ u_6 \end{array} \begin{array}{c} \begin{array}{ccc} y_1 & y_4 & y_6 \end{array} \\ \begin{bmatrix} 5 & 2 & 2 \\ 1 & 4 & 2 \\ 5 & 1 & 5 \end{bmatrix} \end{array}, \; M_r(C'_{12}) = \begin{array}{c} \\ u_2 \\ u_3 \\ u_4 \end{array} \begin{array}{c} \begin{array}{ccc} y_3 & y_5 & y_7 \end{array} \\ \begin{bmatrix} 3 & 3 & 1 \\ 3 & 1 & 2 \\ 1 & 3 & 1 \end{bmatrix} \end{array}$$

Clustering level 7: $C'_{11} \rightarrow C'_{13}$ ($\gamma = 0.4$), C'_{14} ($\gamma = 1$)

$$M_r(C'_{13}) = \begin{array}{c} \\ u_1 \\ u_6 \end{array} \begin{array}{c} \begin{array}{cc} y_4 & y_6 \end{array} \\ \begin{bmatrix} 2 & 2 \\ 1 & 5 \end{bmatrix} \end{array}, \; M_r(C'_{14}) = \begin{array}{c} \\ u_5 \end{array} \begin{array}{c} y_1 \\ \begin{bmatrix} 1 \end{bmatrix} \end{array}$$

Clustering level 8: $C'_{13} \rightarrow C'_{15}$ $(\gamma = 1)$, C'_{16} $(\gamma = 1)$

$$M_r(C'_{15}) = \begin{matrix} & y_4 \\ u_6 & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}, \; M_r(C'_{16}) = \begin{matrix} & y_6 \\ u_1 & \begin{bmatrix} 2 \end{bmatrix} \end{matrix}$$

Clustering level 9: $C'_{7} \rightarrow C'_{17}$ $(\gamma = 1)$, C'_{18} $(\gamma = 1)$

$$M_r(C'_{17}) = \begin{matrix} & y_{10} \\ u_8 & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}, \; M_r(C'_{18}) = \begin{matrix} & y_{11} \\ u_{12} & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}$$

Clustering level 10: $C'_{9} \rightarrow C'_{19}$ $(\gamma = 1)$, C'_{20} $(\gamma = 1)$

$$M_r(C'_{19}) = \begin{matrix} & y_8 \\ u_7 & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}, \; M_r(C'_{20}) = \begin{matrix} & y_9 \\ u_{11} & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}$$

Clustering level 11: $C'_{12} \rightarrow C'_{21}$ $(\gamma = 0.667)$, C'_{22} $(\gamma = 1)$

$$M_r(C'_{21}) = \begin{matrix} & y_3 & y_7 \\ u_2 & 3 & 1 \\ u_4 & 1 & 1 \end{matrix}, \; M_r(C'_{22}) = \begin{matrix} & y_5 \\ u_3 & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}$$

Clustering level 12: $C'_{21} \rightarrow C'_{23}$ $(\gamma = 1)$, C'_{24} $(\gamma = 1)$

$$M_r(C'_{23}) = \begin{matrix} & y_3 \\ u_4 & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}, \; M_r(C'_{24}) = \begin{matrix} & y_7 \\ u_2 & \begin{bmatrix} 1 \end{bmatrix} \end{matrix}$$

The optimal clustering result provided above is summarized in Figure 5.3. Also, one

more optimal clustering result is obtained, and shown in Figure 5.4. The subnetworks of closely coupled and decoupled units are also summarized in Figure 5.5 and Figure 5.6. In these figures, the HDA process is decomposed into smaller subnetworks at each circular node. Note that, in both solutions, the product column is decoupled from the reaction-recycle network. Note also that the fully decentralized control configuration in both solutions matches the one obtained in the previous subsection using the agglomerative clustering analysis.

The major difference between the solutions obtained from both hierarchical clustering methods (other than the coupling of the reaction-recycle network and the distillation column network) lies in the subnetwork structure of the reaction-recycle network. These structures can be analyzed further using additional process information to select the block decentralized control configuration to be implemented.

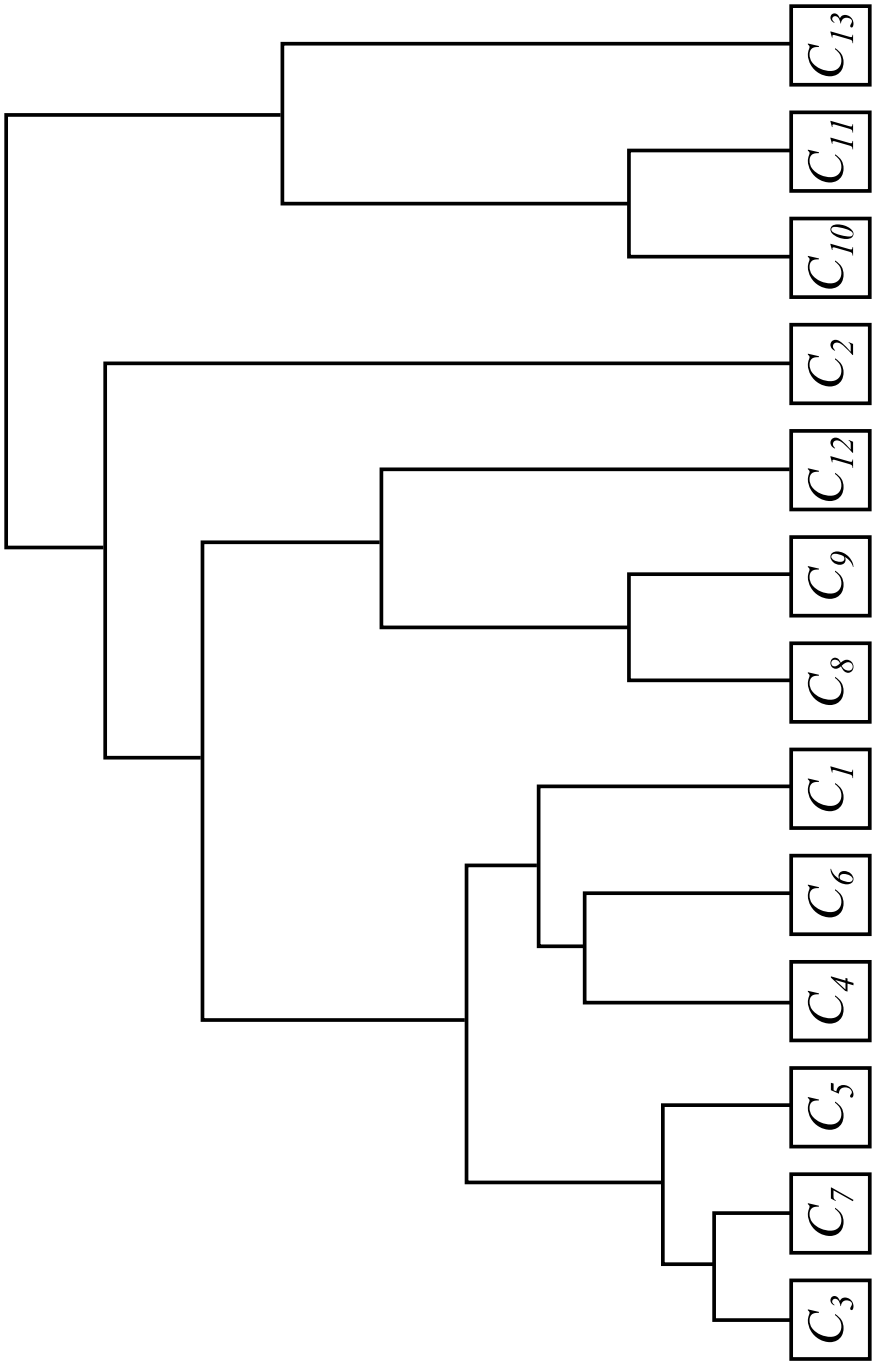


Figure 5.3: Hierarchical clustering result for the HDA process: divisive clustering, solution 1

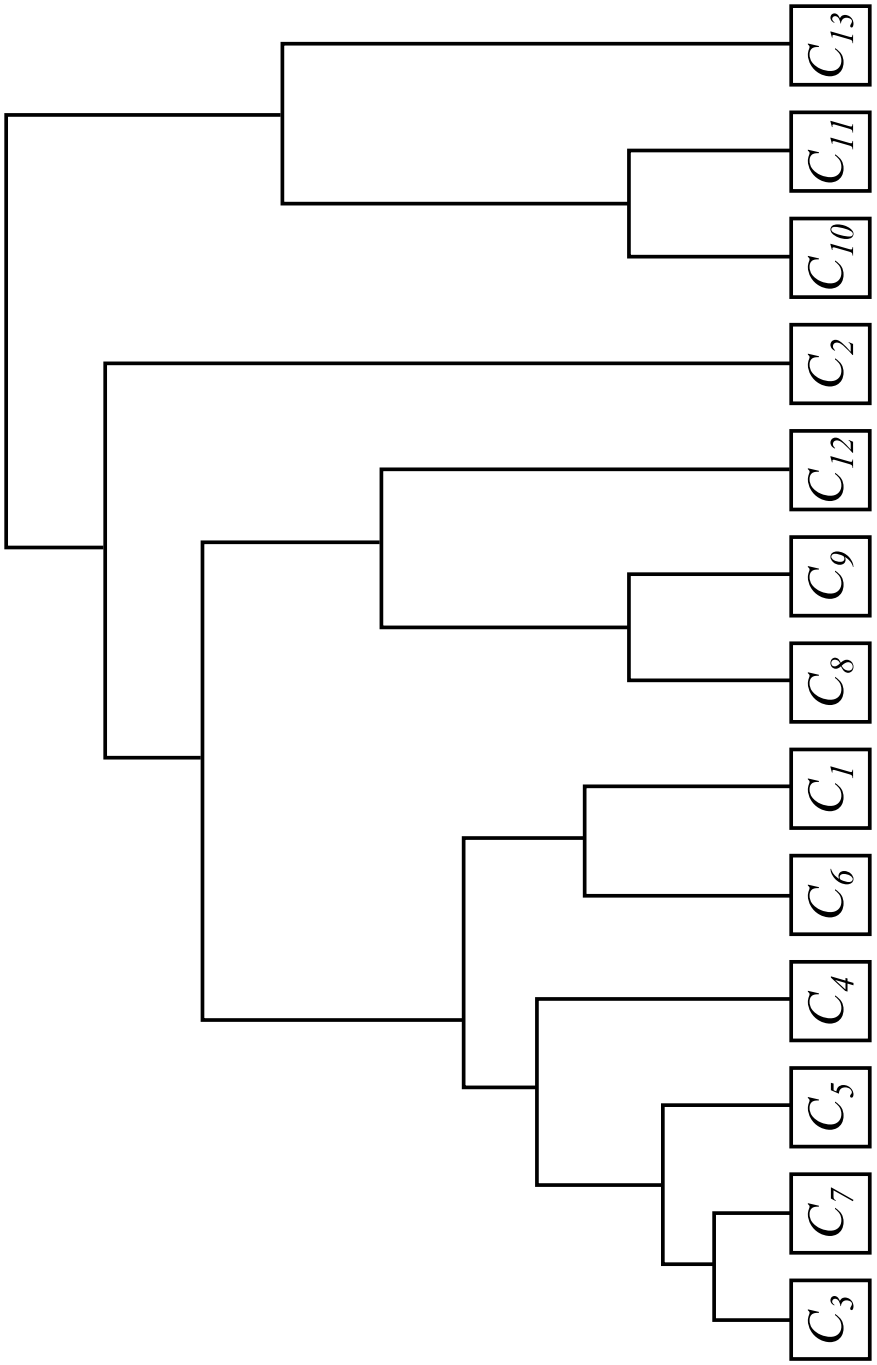


Figure 5.4: Hierarchical clustering result for the HDA process: divisive clustering, solution 2

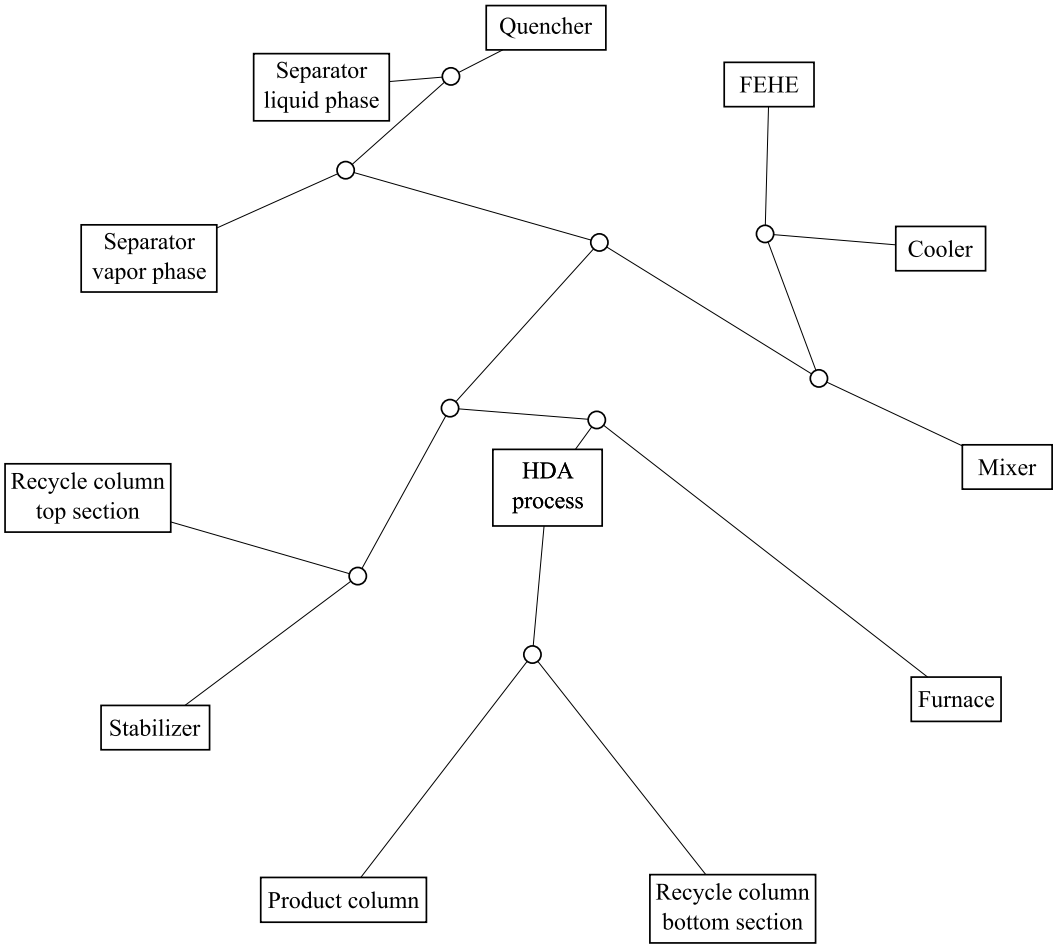


Figure 5.5: Subnetworks of the HDA process, solution 1

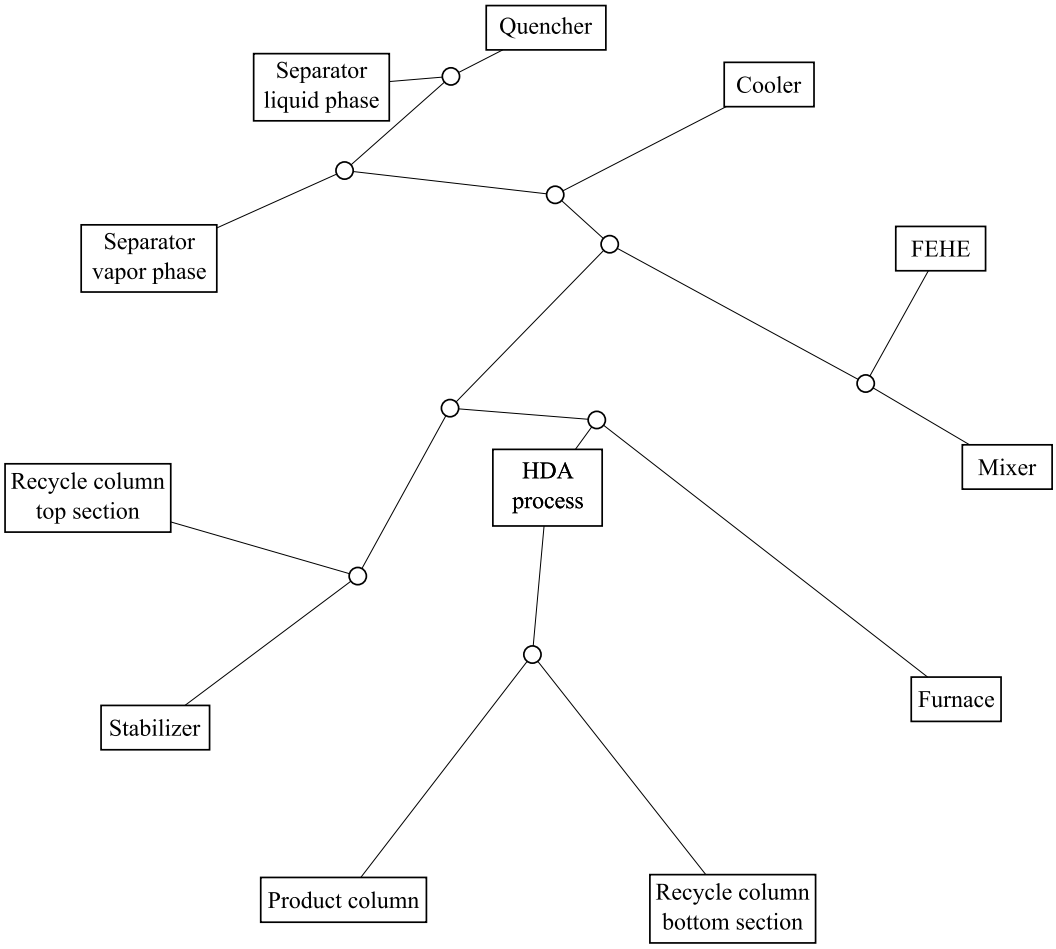


Figure 5.6: Subnetworks of the HDA process, solution 2

CHAPTER 6

Summary and Future Research

6.1 Thesis summary

This thesis addressed the control structure design for complex integrated process networks. Process integration, while motivated by expensive energy resources and environmental regulations, results in significant interactions among different process units, making the control structure design challenging. While extensive research activity has been being pursued to develop control approaches for such networks, a rigorous control-oriented network decomposition framework for identifying sub-systems, which can be effectively controlled and coordinated, is currently lacking. Motivated by this, a generic framework was developed, which decomposes process networks into subnetworks based on two different criteria: time scales in which process variables evolve, and topological closeness between input variables and output variables.

In the first part of the thesis, complex integrated process networks, featuring different energy/material flows spanning multiple orders of magnitude, were analyzed. A graph theory-based model reduction framework for complex energy integrated process networks was described, and a similar framework for complex material integrated process networks was then developed. These essentially mimic the main steps of the singular perturbations-based analysis; they rely on information on the connectivity of process networks and the order of magnitude of different energy/material flows to automatically generate information on the time scales in which different process variables evolve, and hierarchical control structures by classifying potential manipulated inputs and controlled outputs in each time scale.

Such frameworks provide several advantages over the singular perturbations-based analysis. First, they are scalable as they adopt efficient graph theory algorithms which can be easily automated. Also, they are generic and have the potential to be applied to different types of complex networks, where the degree of interaction among different members of networks span several orders of magnitude. Moreover, they require minimal process knowledge, namely the connectivity and the orders of magnitude of the energy/material flows.

The second part of the thesis focused on the control structure design for more general complex integrated process networks. First, a method for optimal control configuration synthesis based on structural coupling was developed. Specifically, an integer optimization problem was proposed to synthesize optimal fully decentralized control configurations using the relative degree as a measure of structural coupling. An agglomerative hierarchical clustering procedure was then developed to identify groups of inputs and outputs such that the variables in the same group interact strongly with each other, while different groups are loosely coupled. This was based on a distance metric

defined between input/output clusters, and additional distance measures to differentiate among clusters with the same distance. A compactness measure for input/output clusters was also defined to propose a divisive hierarchical clustering procedure. It was shown that the proposed hierarchical clustering procedures can be used to identify not only the groups of inputs and outputs that are strongly connected, but also the groups of process units that are interacting strongly each other.

The results presented in the two parts of the thesis can be integrated as summarized in Figure 6.1, to systematically decompose complex integrated process networks. The missing links in the overall framework are highlighted using the dashed arrows, which need to be filled in the future research. In what follows, potential research directions are briefly discussed.

6.2 Future research

6.2.1 Complex networks combining energy and material integration

A combined model reduction framework is needed to establish correlations between the time scales where energy balance variables evolve, and the time scales where material balance variables evolve. The framework developed in Chapter 3 can be extended to this end.

In this case, the process flow graph can be modified by assigning an additional weight, which represents the order of magnitude of temperature of different flows, to each edge. Noting that material flows act as energy carriers, the order of magnitude of energy flow can be determined by the product of the orders of magnitude of material flow and temperature. Pure energy flows (e.g. heat transfer in a heat exchanger) can be represented using edges with the first n_s (the number of chemical species involved in the network) weights being zero.

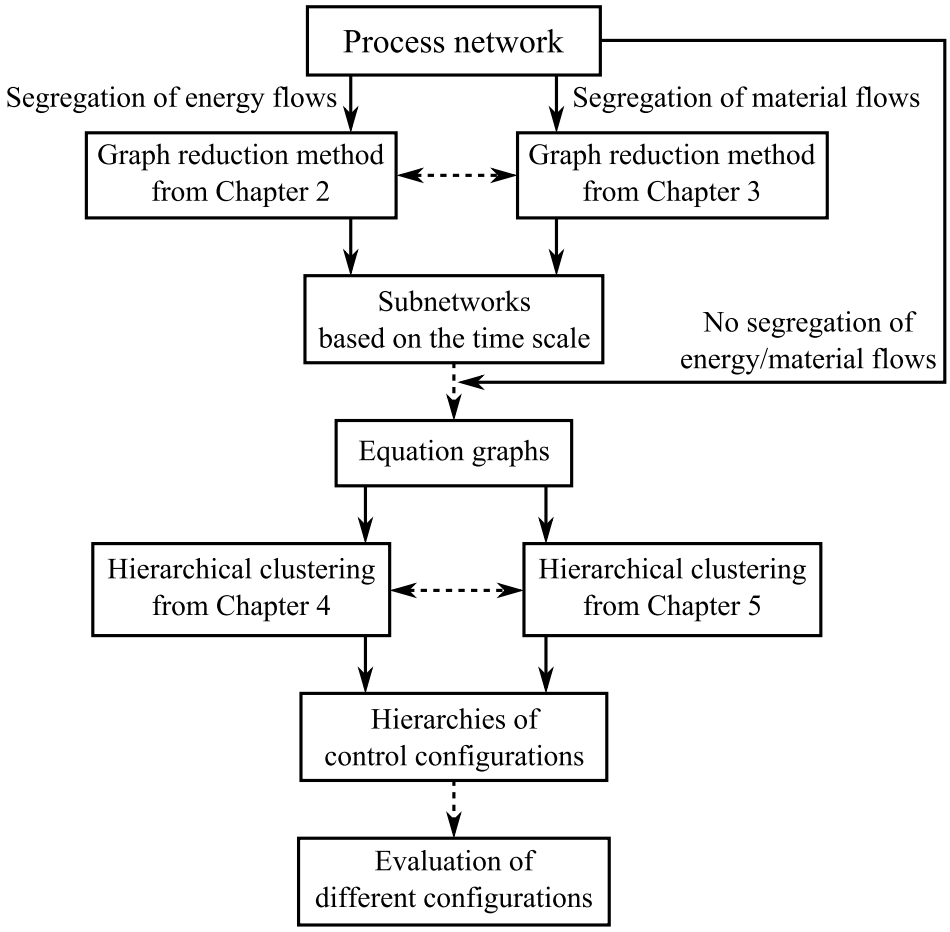


Figure 6.1: Flowchart of the control structure design framework proposed in the thesis

In Chapter 2, we discussed that process networks with large energy throughput exhibit two-time scale dynamics, in which energy balance variables evolve in the fast time scale, while material balance variables evolve in the slow time scale. Thus, this building block will play an important role in the graph theoretic algorithm to be developed. Recall that the subroutine PostProcess of the algorithm developed in Chapter 3 modifies the weights of the edges connected to the constituents of partial recycles. Similarly, once large energy throughputs are identified, the last weight of the edges connected to the constituents of large energy throughputs can be set to zero so that the energy dynamics are not affected by smaller energy flows.

6.2.2 Analysis of input-to-output paths

In the methods developed in this thesis, only the length of the shortest input-to-output paths (IOPs) is considered to design block decentralized control configurations, while it is also important to closely look at the actual paths in which the inputs affect the outputs (i.e. sets of state variables that are affected by the control actions). If a certain clutering result contains many state variables that appear in multiple shortest IOPs, the block decentralized control configuration given by that clustering result will be highly interactive. Thus, different clustering results with the same decentrality measure can be further evaluated based on the number of state variables that appear in multiple IOPs.

A graph theoretic algorithm on the basis of eqation graphs needs to be developed to carry out such analysis in an automatic manner. For each cluster, the algorithm to be developed will first identify the inputs and the outputs in the cluster, and then a subgraph of the original equation graph will be formed which contains the corresponding input nodes and output nodes, and the state nodes that are included in the shortest IOPs. After obtaining subgraphs for all the clusters, the algorithm will identify the state nodes that appear in multiple subgraphs.

6.2.3 Optimal clustering

The hierarchical clustering approaches, which are developed in Chapter 4 and Chapter 5, generate the entire hierarchy of clusters, without provisions to end the procedure before the final level of clustering is executed. Note that, as the size of network increases, completing the hierarchical clustering becomes more computationally expensive. Stopping rules (or clustering validation criteria) are often used to specify the conditions which, when met, end the procedure. They can be incorporated as a form of either external criteria (i.e. criteria specified by the user) or internal criteria (i.e. criteria inherent to the structure of clustering); however, internal criteria are preferred as external criteria heavily rely on the intuition/experience of the user.

In the clustering analysis literature, several internal cluster validation criteria (e.g. Davies-Bouldin index, Dunn's index, Calinski Harabasz index, silhouette index) have been proposed [108–115]. Although the exact formulations may differ, their common underlying form is:

$$\text{Cluster validity index} = \text{Overall closeness measure} \times \text{Overall compactness measure}$$

In our case, a possible approach is to use the *average* closeness/compactness measures as a measure of the overall closeness/compactness, which will yield the following index to be evaluated at each level of clustering:

$$\text{Cluster validity index} = \left[\frac{\sum_{i=1}^{n_C}\sum_{j=1}^{n_C}\Gamma(C_i, C_j)}{2n_C(n_C-1)}\right] \cdot \left[\frac{\sum_{i=1}^{n_C}\gamma(C_i)}{n_C}\right]$$

where Γ and γ are the closeness and compactness measures, respectively, defined in Chapter 5, and n_C is the number of clusters. A clustering result with compact and well separated clusters will give us a high value of such index, and thus the optimal

clustering will be the clustering with the largest value of such index.

Once a proper cluster validation index is identified, an optimization problem proposed in Chapter 5 can be modified to identify the optimal clusterings. In this case, the objective function to be maximized will be the index itself, and the constraints will be modified accordingly to allow large clusters to be split into multiple clusters instead of just two, and the dimensions of the two sets of binary variables (i.e. $U(l,i)$ and $Y(l,j)$) will be changed from $2 \times n$ to $n_C \times n$, where n_C is now a free variable.

6.2.4 Clustering using aggregate topological closeness measures

The proposed hierarchical clustering methods are indeed generally applicable, i.e. the closeness and compactness measures can be redefined by replacing (or augmenting) the relative degrees with different measures (e.g. to account for the significance of the interactions between inputs and outputs). This approach can be pursued, starting with a reformulation of the equation graph used in the generic calculation of relative degrees, by assigning the weight to each edge. The weights could include the magnitude of the interaction (e.g. values of Jacobian elements at the nominal steady state) as well as time constants obtained from identification experiments. Such information then can be used to define an *aggregate topological closeness* measure between the input u_i and the output y_j, $\bar{r}_{ij}$, which can be computed as the summation of the weights of the edges that are contained in the shortest path between them, i.e.:

$$\bar{r}_{ij} = \sum_{e_i \in IOP(i,j)} w(e_i)$$

where $IOP(i,j)$ is the set of edges contained in the shortest path between u_i and y_j, and $w(e_i)$ is the weight of the edge e_i.

The robustness of the resulting optimal clustering results with respect to different

operating conditions can be further evaluated using sensitivity analyses. Connections with existing measures for developing block decentralized control structures based on static information (e.g. using the block relative gain array [14, 116, 117]) will also need to be explored through alternative reformulations of the clustering criteria that place more weight to the static interactions among the input/output variables.

Bibliography

[1] M. van de Wal and B. de Jager. A review of methods for input/output selection. *Automatica*, 37(4):487–510, 2001.

[2] G.P. Rangaiah and V. Kariwala. *Plantwide control: Recent developments and applications*. John Wiley & Sons, 2012.

[3] R.M. Price and C. Georgakis. Plantwide regulatory control design procedure using a tiered framework. *Ind. Eng. Chem. Res.*, 32(11):2693–2705, 1993.

[4] C.S. Ng and G. Stephanopoulos. Synthesis of control systems for chemical plants. *Comput. Chem. Eng.*, 20:S999–S1004, 1996.

[5] M.L. Luyben, B.D. Tyreus, and W.L. Luyben. Plantwide control design procedure. *AIChE J.*, 43(12):3161–3174, 1997.

[6] S. Skogestad. Plantwide control: The search for the self-optimizing control structure. *J. Process Control*, 10(5):487–507, 2000.

[7] S. Skogestad. Self-optimizing control: The missing link between steady-state optimization and control. *Comput. Chem. Eng.*, 24(2):569–575, 2000.

[8] M. Baldea, A. Araujo, S. Skogestad, and P. Daoutidis. Dynamic considerations in the synthesis of self-optimizing control structures. *AIChE J.*, 54(7):1830–1841, 2008.

[9] J.W. Chang and C.C. Yu. The relative gain for non-square multivariable systems. *Chem. Eng. Sci.*, 45(5):1309–1323, 1990.

[10] D. Chen and D.E. Seborg. Relative gain array analysis for uncertain process models. *AIChE J.*, 48(2):302–310, 2002.

[11] V. Kariwala, S. Skogestad, and J.F. Forbes. Relative gain array for norm-bounded uncertain systems. *Ind. Eng. Chem. Res.*, 45(5):1751–1757, 2006.

[12] V. Manousiouthakis, R. Savage, and Y. Arkun. Synthesis of decentralized process control structures using the concept of block relative gain. *AIChE J.*, 32(6):991–1003, 1986.

[13] Y. Arkun. Dynamic block relative gain and its connection with the performance and stability of decentralized control structures. *Int. J. Control*, 46(4):1187–1193, 1987.

[14] V. Kariwala, J.F. Forbes, and E.S. Meadows. Block relative gain: Properties and pairing rules. *Ind. Eng. Chem. Res.*, 42(20):4564–4574, 2003.

[15] J.A. Heath, I.K. Kookos, and J.D. Perkins. Process control structure selection based on economics. *AIChE J.*, 46(10):1998–2016, 2000.

[16] A. Psaltis, I.K. Kookos, and C. Kravaris. Plant-wide control structure selection methodology based on economics. *Comput. Chem. Eng.*, 52:240–248, 2013.

[17] M. Ellis and P.D. Christofides. Selection of control configurations for economic model predictive control systems. *AIChE J.*, 60(9):3230–3242, 2014.

[18] A. Psaltis, I.K. Kookos, and C. Kravaris. Plantwide control structure selection methodology for the benchmark vinyl acetate monomer plant. *Comput. Chem. Eng.*, 62:108–116, 2014.

[19] J. Morud and S. Skogestad. Effects of recycle on dynamics and control of chemical processing plants. *Comput. Chem. Eng.*, 18:S529–S534, 1994.

[20] P. Mizsey and I. Kalmar. Effects of recycle on control of chemical processes. *Comput. Chem. Eng.*, 20:S883–S888, 1996.

[21] C.S. Bildea, A.C. Dimian, and P.D. Iedema. Nonlinear behavior of reactor-separator-recycle systems. *Comput. Chem. Eng.*, 24(2):209–215, 2000.

[22] F. Reyes and W.L. Luyben. Steady-state and dynamic effects of design alternatives in heat-exchanger/furnace/reactor processes. *Ind. Eng. Chem. Res.*, 39(9):3335–3346, 2000.

[23] S. Pushpavanam and A. Kienle. Nonlinear behavior of an ideal reactor separator network with mass recycle. *Chem. Eng. Sci.*, 56(8):2837–2849, 2001.

[24] Y.H. Chen and C.C. Yu. Design and control of heat-integrated reactors. *Ind. Eng. Chem. Res.*, 42(12):2791–2808, 2003.

[25] T. Larsson, M.S. Govatsmark, S. Skogestad, and C.C. Yu. Control structure selection for reactor, separator, and recycle processes. *Ind. Eng. Chem. Res.*, 42(6):1225–1234, 2003.

[26] A.A. Kiss, C.S. Bildea, A.C. Dimian, and P.D. Iedema. Design of recycle systems with parallel and consecutive reactions by nonlinear analysis. *Ind. Eng. Chem. Res.*, 44(3):576–587, 2005.

[27] Y. Zhu and X. Liu. Dynamics and control of high purity heat integrated distillation columns. *Ind. Eng. Chem. Res.*, 44(23):8806–8814, 2005.

[28] E.W. Jacobsen and M. Berezowski. Chaotic dynamics in homogeneous tubular reactors with recycle. *Chem. Eng. Sci.*, 53(23):4023–4029, 1998.

[29] E.W. Jacobsen. On the dynamics of integrated plants–non-minimum phase behavior. *J. Process Control*, 9(5):439–451, 1999.

[30] A. Kumar and P. Daoutidis. Nonlinear dynamics and control of process systems with recycle. *J. Process Control*, 12(4):475–484, 2002.

[31] S.S. Jogwar, M. Baldea, and P. Daoutidis. Dynamics and control of process networks with large energy recycle. *Ind. Eng. Chem. Res.*, 48(13):6087–6097, 2009.

[32] S.S. Jogwar, M. Baldea, and P. Daoutidis. Tight energy integration: Dynamic impact and control advantages. *Comput. Chem. Eng.*, 34(9):1457–1466, 2010.

[33] B.D. Tyreus and W.L. Luyben. Controlling heat integrated distillation columns. *Chem. Eng. Prog.*, 72(9):59–66, 1976.

[34] M.L. Luyben and C.A. Floudas. Analyzing the interaction of design and control–2. reactor-separator-recycle system. *Comput. Chem. Eng.*, 18(10):971–993, 1994.

[35] P. Mizsey, N.T. Hau, N. Benko, I. Kalmar, and Z. Fonyo. Process control for energy integrated distillation schemes. *Comput. Chem. Eng.*, 22:S427–S434, 1998.

[36] A. Malcolm, J. Polan, L. Zhang, B.A. Ogunnaike, and A.A. Linninger. Integrating systems design and control using dynamic flexibility analysis. *AIChE J.*, 53(8):2048–2061, 2007.

[37] H. Cui and E.W. Jacobsen. Performance limitations in decentralized control. *J. Process Control*, 12(4):485–494, 2002.

[38] V. Kariwala. Fundamental limitation on achievable decentralized performance. *Automatica*, 43(10):1849–1854, 2007.

[39] M. Baldea and P. Daoutidis. *Dynamics and nonlinear control of integrated process systems*. Cambridge University Press, 2012.

[40] M. Baldea, P. Daoutidis, and A. Kumar. Dynamics and control of integrated networks with purge streams. *AIChE J.*, 52(4):1460–1472, 2006.

[41] M. Baldea and P. Daoutidis. Control of integrated process networks–a multi-time scale perspective. *Comput. Chem. Eng.*, 31(5):426–444, 2007.

[42] S.S. Jogwar, A.I. Torres, and P. Daoutidis. Networks with large solvent recycle: Dynamics, hierarchical control, and a biorefinery application. *AIChE J.*, 58(6):1764–1777, 2012.

[43] S.S. Jogwar and P. Daoutidis. Energy flow patterns and control implications for integrated distillation networks. *Ind. Eng. Chem. Res.*, 49(17):8048–8061, 2010.

[44] N.P. Vora, M.-N. Contou-Carrere, and P. Daoutidis. Model reduction of multiple time scale processes in non-standard singularly perturbed form. In A.N. Gorban, N.K. Kazantzis, I.G. Kevrekidis, H.C. Öttinger, and C. Theodoropoulos, editors, *Model reduction and coarse-graining approaches for multiscale phenomena*, pages 99–113. Springer, Berlin Heidelberg, 2006.

[45] B. Bollobás. *Modern graph theory*, volume 184. Springer-Verlag, New York, 1998.

[46] S. Redner. How popular is your paper? An empirical study of the citation distribution. *Eur. Phys. J. B*, 4(2):131–134, 1998.

[47] M.E.J. Newman. The structure of scientific collaboration networks. *P. Natl. Acad. Sci.*, 98(2):404–409, 2001.

[48] M.E.J. Newman. Scientific collaboration networks. i. network construction and fundamental results. *Phys. Rev. E*, 64(1):016131, 2001.

[49] M.E.J. Newman. Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. *Phys. Rev. E*, 64(1):016132, 2001.

[50] H. Jeong, B. Tombor, R. Albert, Z.N. Oltvai, and A.-L. Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000.

[51] E. Ravasz, A.L. Somera, D.A. Mongru, Z.N. Oltvai, and A.-L. Barabási. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586):1551–1555, 2002.

[52] J. Camacho, R. Guimerà, and L.A.N. Amaral. Robust patterns in food web structure. *Physical Review Letters*, 88(22):228102, 2002.

[53] J.A. Dunne, R.J. Williams, and N.D. Martinez. Food-web structure and network theory: the role of connectance and size. *P. Natl. Acad. Sci.*, 99(20):12917–12922, 2002.

[54] S.S. Jogwar, S. Rangarajan, and P. Daoutidis. Multi-time scale dynamics in energy-integrated networks: A graph theoretic analysis. In *IFAC World Congress*, volume 18, pages 6085–6090, 2011.

[55] B. Wellman. Computer networks as social networks. *Science*, 293(5537):2031–2034, 2001.

[56] M. Girvan and M.E.J. Newman. Community structure in social and biological networks. *P. Natl. Acad. Sci.*, 99(12):7821–7826, 2002.

[57] S. Allesina, A. Bodini, and C. Bondavalli. Ecological subsystems via graph theory: the role of strongly connected components. *Oikos*, 110(1):164–176, 2005.

[58] M. Pascual and J.A. Dunne. *Ecological networks: linking structure to dynamics in food webs*. Santa Fe Institute Studies on the Sciences of Complexity. Oxford University Press, 2006.

[59] S.R. Borrett, B.D. Fath, and B.C. Patten. Functional integration of ecological networks through pathway proliferation. *J. Theor. Biol.*, 245(1):98–111, 2007.

[60] M. Farina and R. Scattolini. Distributed predictive control: a non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. *Automatica*, 48(6):1088–1096, 2012.

[61] N. Hudon and J. Bao. Dissipativity-based decentralized control of interconnected nonlinear chemical processes. *Comput. Chem. Eng.*, 45:84–101, 2012.

[62] Y. Sun and N.H. El-Farra. Resource aware quasi-decentralized control of networked process systems over wireless sensor networks. *Chem. Eng. Sci.*, 69(1):93–106, 2012.

[63] M. Baldea, N.H. El-Farra, and B.E. Ydstie. Dynamics and control of chemical process networks: Integrating physics, communication and computation. *Comput. Chem. Eng.*, 51:42–54, 2013.

[64] P.D. Christofides, R. Scattolini, D.M. de la Peña, and J. Liu. Distributed model predictive control: A tutorial review and future research directions. *Comput. Chem. Eng.*, 51:21–41, 2013.

[65] Y. Sun and N.H. El-Farra. Robust quasi-decentralized control of uncertain process networks. *Ind. Eng. Chem. Res.*, 53(18):7421–7433, 2013.

[66] M.J. Tippett and J. Bao. Distributed model predictive control based on dissipativity. *AIChE J.*, 59(3):787–804, 2013.

[67] M.A. Porter, J.-P. Onnela, and P.J. Mucha. Communities in networks. *Not. Am. Math. Soc.*, 56(9):1082–1097, 2009.

[68] S. Fortunato. Community detection in graphs. *Phys. Rep.*, 486(3):75–174, 2010.

[69] P. Daoutidis and C. Kravaris. Structural evaluation of control configurations for multivariable nonlinear processes. *Chem. Eng. Sci.*, 47(5):1091–1107, 1992.

[70] R.M. Cormack. A review of classification. *J. Roy. Stat. Soc. A Sta.*, pages 321–367, 1971.

[71] A.D. Gordon. A review of hierarchical classification. *J. Roy. Stat. Soc. A Sta.*, 150(2):119–137, 1987.

[72] S.C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.

[73] S. Heo, S. Rangarajan, P. Daoutidis, and S.S. Jogwar. Graph reduction of complex energy-integrated networks: Process systems applications. *AIChE J.*, 60(3):995–1012, 2014.

[74] T.F. Yee, I.E. Grossmann, and Z. Kravanja. Simultaneous optimization models for heat integration–I. Area and energy targeting and modeling of multi-stream exchangers. *Comput. Chem. Eng.*, 14(10):1151–1164, 1990.

[75] A.W. Westerberg. A retrospective on design and process synthesis. *Comput. Chem. Eng.*, 28(4):447–458, 2004.

[76] M. Baldea and P. Daoutidis. Modeling, dynamics and control of process networks with high energy throughput. *Comput. Chem. Eng.*, 32(9):1964–1983, 2008.

[77] S. Heo, S.S. Jogwar, S. Rangarajan, and P. Daoutidis. Graph reduction for hierarchical control of energy integrated process networks. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 6388–6393. IEEE, 2012.

[78] D.L. Terrill and J.M. Douglas. Heat-exchanger network analysis. 1. Optimization. *Ind. Eng. Chem. Res.*, 26(4):685–691, 1987.

[79] C.C. Zimmerman and R. York. Thermal demethylation of toluene. *Ind. Eng. Chem. Proc. Des. Dev*, 3(3):254–258, 1964.

[80] D. Georgis, S.S. Jogwar, A.S. Almansoori, and P. Daoutidis. Design and control of energy integrated SOFC systems for *in situ* hydrogen production and power generation. *Comput. Chem. Eng.*, 35(9):1691–1704, 2011.

[81] O. Annakou and P. Mizsey. Rigorous comparative study of energy-integrated distillation schemes. *Ind. Eng. Chem. Res.*, 35(6):1877–1885, 1996.

[82] S.S. Jogwar and P. Daoutidis. Dynamics and control of vapor recompression distillation. *J. Process Control*, 19(10):1737–1750, 2009.

[83] M.J. Andrecovich and A.W. Westerberg. An MILP formulation for heat-integrated distillation sequence synthesis. *AIChE J.*, 31(9):1461–1474, 1985.

[84] R. Tarjan. Enumeration of the elementary circuits of a directed graph. *SIAM J. Comput.*, 2(3):211–216, 1973.

[85] A.I. Torres, P. Daoutidis, and M. Tsapatsis. Continuous production of 5-hydroxymethylfurfural from fructose: a design case study. *Energy Environ. Sci.*, 3(10):1560–1572, 2010.

[86] M.L. Luyben and B.D. Tyréus. An industrial design/control study for the vinyl acetate monomer process. *Comput. Chem. Eng.*, 22(7):867–877, 1998.

[87] R. Chen, K. Dave, T.J. McAvoy, and M. Luyben. A nonlinear dynamic model of a vinyl acetate process. *Ind. Eng. Chem. Res.*, 42(20):4478–4487, 2003.

[88] S. Heo, W.A. Marvin, and P. Daoutidis. Automated synthesis of control configurations for process networks based on structural coupling. *Chem. Eng. Sci.*, 2015.

[89] P.D. Christofides, J.F. Davis, N.H. El-Farra, D. Clark, K.R.D. Harris, and J.N. Gipson. Smart plant operations: Vision, progress and challenges. *AIChE J.*, 53(11):2734–2741, 2007.

[90] B.E. Ydstie. Passivity based control via the second law. *Comput. Chem. Eng.*, 26(7):1037–1048, 2002.

[91] J.B. Rawlings and B.T. Stewart. Coordinating multiple optimization-based controllers: New opportunities and challenges. *J. Process Control*, 18(9):839–845, 2008.

[92] J. Liu, D. Muñoz de la Peña, and P.D. Christofides. Distributed model predictive control of nonlinear process systems. *AIChE J.*, 55(5):1171–1184, 2009.

[93] R. Scattolini. Architectures for distributed and hierarchical model predictive control–a review. *J. Process Control*, 19(5):723–731, 2009.

[94] P. Mhaskar, A. Gani, C. McFall, P.D. Christofides, and J.F. Davis. Fault-tolerant control of nonlinear process systems subject to sensor faults. *AIChE J.*, 53(3):654–668, 2007.

[95] Y. Sun and N.H. El-Farra. Quasi-decentralized model-based networked control of process systems. *Comput. Chem. Eng.*, 32(9):2016–2029, 2008.

[96] Z.Q. Jiang, W.X. Zhou, B. Xu, and W.K. Yuan. Process flow diagram of an ammonia plant as a complex network. *AIChE J.*, 53(2):423–428, 2007.

[97] M.E.J. Newman. Detecting community structure in networks. *Eur. Phys. J. B*, 38(2):321–330, 2004.

[98] M.E.J. Newman. Modularity and community structure in networks. *P. Natl. Acad. Sci.*, 103(23):8577–8582, 2006.

[99] S. Varigonda, T. Kalmár-Nágy, B. LaBarre, and I. Mezic. Graph decomposition methods for uncertainty propagation in complex, nonlinear interconnected dynamical systems. In *Decision and Control (CDC), 2004 43rd IEEE Conference on*, volume 2, pages 1794–1798. IEEE, 2004.

[100] T. Schné and K.M. Hangos. Decentralised controller structure design and retrofit of process systems based on graph theory. *Int. J. Syst. Sci.*, 42(6):1023–1033, 2011.

[101] H.K. Khalil. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, 2002.

[102] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to algorithms*. The MIT press, 2001.

[103] IBM. *IBM ILOG CPLEX Optimization Studio 12.2*, 2011.

[104] K. Kunen and J. Vaughan. *Handbook of set-theoretic topology*. North Holland, Amsterdam, 1984.

[105] I.K. Kookos and J.D. Perkins. Heuristic-based mathematical programming framework for control structure selection. *Ind. Eng. Chem. Res.*, 40(9):2079–2088, 2001.

[106] A.C.B. de Araújo, M. Govatsmark, and S. Skogestad. Application of plantwide control to the hda process. i–steady-state optimization and self-optimizing control. *Control Eng. Pract.*, 15(10):1222–1237, 2007.

[107] A.C.B. de Araújo, E.S. Hori, and S. Skogestad. Application of plantwide control to the hda process. ii–regulatory control. *Ind. Eng. Chem. Res.*, 46(15):5159–5174, 2007.

[108] D.L. Davies and D.W. Bouldin. A cluster separation measure. *IEEE T. Pattern Anal.*, PAMI-1(2):224–227, 1979.

[109] J.C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *J. Cybern.*, 3:35–57, 1973.

[110] J.C. Dunn. Well-separated clusters and optimal fuzzy partitions. *J. Cybern.*, 4(1):95–104, 1974.

[111] T. Caliński and J. Harabasz. A dendrite method for cluster analysis. *Commun. Stat. A-Theor.*, 3(1):1–27, 1974.

[112] L. Kaufman and P.J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons, 2009.

[113] G.W. Milligan and M.C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[114] N.R. Pal and J. Biswas. Cluster validation using graph theoretic concepts. *Pattern Recogn.*, 30(6):847–857, 1997.

[115] J.C. Bezdek and N.R. Pal. Some new indexes of cluster validity. *IEEE T. Syst. Man Cy. B*, 28(3):301–315, 1998.

[116] V. Manousiouthakis and M. Nikolaou. Analysis of decentralized control structures for nonlinear systems. *AIChE J.*, 35(4):549–558, 1989.

[117] J. Chen, J.S. Freudenberg, and C.N. Nett. The role of the condition number and the relative gain array in robustness analysis. *Automatica*, 30(6):1029–1035, 1994.