Decentralized multi-model control for MIMO processes with applications in HVAC systems

He, Ming

2006

He, M. (2006). Decentralized multi-model control for MIMO processes with applications in HVAC systems. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/4348>

<https://doi.org/10.32657/10356/4348>

Nanyang Technological University

Downloaded on 26 Apr 2023 21:06:58 SGT

q407012

Decentralized Multi-model Control
for
MIMO Processes with Applications in HVAC Systems

HE MING

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2006

TK
7881.2
H432
2006

Acknowledgements

I would like to express my sincere gratitude and appreciation to my supervisor, Associate Professor Dr. Cai Wenjian, for his patient supervision, enthusiastic encouragement and support throughout the duration of this research. His rigorous scholarship and perceptive faculties have had a great influence on this work.

I would like to give a special thank to Dr. Li Shaoyuan (Shanghai Jiaotong University, China) for his constructive guidance, helpful suggestions and discussions. I would also like to thank the past and present colleagues and technicians at Instrumentation & System Engineering Laboratory and Process Instrumentation Laboratory for their hospitality, friendship and assistance.

I am indebted to my family and friends for their patience and friendship. I would like to thank my parents for their enthusiasm and support in my education. Special thanks go to my wife, Li Kun, for her love, companionship and continuous encouragement, understanding and steadfast support.

Finally, I would like to acknowledge the School of Electrical and Electronic Engineering, Nanyang Technological University for providing the award of a scholarship and the research facilities.

Summary

Recently, multi-model control methods for nonlinear processes have attracted a great deal of attentions, which are based on *divide-and-conquer* strategy to divide the complex problems into simpler ones. In this thesis, the author presents a systematic decentralized multi-model control strategy for complex nonlinear multivariable processes. The main work includes: local models identification, local controllers design, and global multi-model control construction.

In local models identification, a decentralized closed-loop identification method based on step tests is proposed, which relaxes the restrictions of most existing multivariable process identification methods. An optimal iterative searching method is developed to determine the critical frequency of each transfer function, which greatly reduces the computational burden. The proposed method requires no prior knowledge about the process, and the obtained local models with simple forms are suitable for the multi-model control construction.

In local controllers design, a new effective loop pairing criterion based on the information of the steady-state gain and the bandwidth of the process transfer function matrix elements is proposed, which provides a comprehensive yet simple description for loop interactions. Based on the proposed criterion index, a simple tuning formula is developed for decentralized controllers. To improve the controller performance, a two-

level PID controller tuning technique derived from multi-step receding horizon optimization is developed to regulate the original controller parameters.

The construction of global multi-model control by an incremental strategy is discussed. Additional local regimes are added based on the evaluation of controller performance and through a scheduled fuzzy distribution. The integration of local regimes is adjusted according to different operating points. The proposed strategy is straightforward, requires little prior knowledge of the process, and can be fully automated. The applications to HVAC process control demonstrate the effectiveness of proposed decentralized multi-model control method.

Contents

Acknowledgements	i
Summary	ii
Contents	iv
List of Figures	vii
List of Tables	x
1 Introduction	1
1.1 Overview of Control Strategies for Nonlinear Multivariable Processes	1
1.2 Motivations and Objectives.....	9
1.3 Major Contributions of the Thesis	14
1.4 Organization of the Thesis	18
2 Closed-loop Parameter Identification in Frequency Domain	19
2.1 Introduction	19
2.2 Step Tests for TITO System Identification	22
2.3 Decentralized Frequency Domain Identification	26
2.4 Transfer Function Modeling and Identification Procedure	33

2.5	Simulation Examples	40
3	Decentralized Control System Design Based on ERGA.....	48
3.1	Introduction	48
3.2	Preliminaries	51
3.3	ERGA-based Loop Pairing	54
3.4	ERGA-based Controller Tuning Method	59
3.5	Simulation Examples	66
4	Fine-tuning Decentralized PID Controllers by Receding Horizon	
	Optimization	70
4.1	Introduction	70
4.2	Model Transformation	74
4.3	Multi-step Prediction.....	77
4.4	General Formulation of PID Controllers with Multi-step Prediction ...	80
4.5	Tuning of Decentralized PID Parameters with Multi-step Prediction ..	84
4.6	Simulation Examples	90
5	Construction of Global Multi-model Control Structure	96
5.1	Introduction	96
5.2	General Multi-model Control Structure	100
5.3	Performance Assessment of Control Systems.....	102
5.4	Multi-model Division Strategy	106
5.5	Determination of Overall Controller Output	113
5.6	Overall Structure of Multi-model Control Strategy	116

6 Applications in HVAC Process Control Systems.....	120
6.1 Introduction	120
6.2 Single-zone HVAC Control System	123
6.3 Two-zone Air Temperature Control System	133
7 Conclusions and Recommendations	141
7.1 Conclusions	141
7.2 Recommendations	144
 Author's Publications	 146
 Bibliography	 147

List of Figures

2.1	Closed-loop TITO control system	22
2.2	Settling time for parameter identification	24
2.3	Decentralized identification process	27
2.4	Desired frequencies in the important frequency range	28
2.5	System step tests	41
2.6	Process Nyquist plots and estimated first-order model frequency responses for Example 2.1	43
2.7	Process Nyquist plots and estimated second-order model frequency responses for Example 2.1	44
2.8	Process Nyquist plots and estimated frequency responses for Example 2.2	45
2.9	Process Nyquist plots and estimated frequency responses for Example 2.3	47
3.1	Closed-loop multivariable control system	52
3.2	Response curve and effective energy of $g_y(j\omega)$	56
3.3	Detuning factor comparison for $\phi < 1.0$	65
3.4	PI and PID closed-loop responses for Example 3.2.....	67

3.5	PI and PID closed-loop responses for Example 3.3.....	69
4.1	Closed-loop response with loops open and close	87
4.2	Multi-step prediction fine-tuning diagram	88
4.3	PI closed-loop responses for Example 4.1	91
4.4	Fine-tuning controller performance for Example 4.1	92
4.5	PI closed-loop responses for Example 4.2.....	93
4.6	Fine-tuning controller performance for Example 4.2	95
5.1	Multi-model controller structure.....	100
5.2	Multiple local regimes structure	102
5.3	Integration of local models and local controllers.....	103
5.4	Initial distribution	108
5.5	Boundary distribution	110
5.6	Additional points inside enclosure of some existing operating points	111
5.7	Weight determination of involved local regimes.....	114
5.8	Global structure of multi-model control	118
6.1	Schematic of a typical building HVAC system	122
6.2	Structure of the single-zone HVAC system.....	124
6.3	Closed-loop responses of single-zone application.....	128
6.4	Manipulated variables of single-zone application	129
6.5	Operating regimes distribution of single-zone application	130
6.6	Multi-model approach performance of single-zone application.....	131

6.7 Multi-model manipulated variables of single-zone application 131

6.8 Fine-tuning performance for single-zone application..... 132

6.9 Performance for different cooling load conditions of single-zone application
..... 133

6.10 Schematic diagram of two-zone system 134

6.11 Pilot plant of centralized HVAC system..... 136

6.12 Operating regimes distribution of two-zone application 137

6.13 Closed-loop performance of Luyben’s method 138

6.14 Manipulated variables of Luyben’s method 139

6.15 Closed-loop performance of two-zone application..... 139

6.16 Manipulated variables of two-zone application..... 140

List of Tables

2.1	Identification results for Example 2.1	42
2.2	Identification results for Example 2.2	44
2.3	Identification results for Example 2.3	46
2.4	Computation cost comparison for Examples 2.1	47
3.1	Effects of interaction on Loop 1	65
4.1	PI controller settings for Example 4.1	91
4.2	Cost value comparison for Example 4.1	92
4.3	PI controller settings for Example 4.2	93
4.4	Cost value comparison for Example 4.2	94
6.1	System parameters for single-zone application	126

Chapter 1

Introduction

1.1 Overview of Control Strategies for Nonlinear Multivariable Processes

The development and evolution of automatic control theory may date back to the famous work of “Regeneration Theory” by Harry Nyquist in 1932 [1], which introduced the Nyquist criterion and laid the foundation for the frequency domain stability analysis of linear control systems. Since then, there have been remarkable developments in linear control theory and their applications in a wide range of industrial process control systems. In many situations, however, the processes encountered in practice are usually nonlinear, and often involve Multiple-input and Multiple-output (MIMO) with strong interactions among the process variables.

Indeed, most important processes are multivariable and nonlinear in nature. In this regard, the linear system model can be considered as a special case which is based on an approximate expression of a naturally nonlinear process in its validity range. The nonlinearity and the extent of interaction could vary with the working points of independent variables, and changes in the directionality of the gain and the dynamics in the wide range of operating conditions, pose a real challenge for modeling and control of the nonlinear MIMO processes. Consequently, a considerable amount of research has been carried out on strategies for controlling nonlinear processes. Those strategies with their own merits and limitations are briefly reviewed as follows.

Linearization methods

Since many industrial processes exhibit only mildly nonlinear dynamic behavior, these processes have been traditionally controlled by using linear systems analysis and design tools [2]. A main reason for the use of linear systems theory is that there is an analytical solution, hence generally it is possible to provide more rigorous stability and performance proofs. Furthermore, the computational demands for linear system simulation and implementation are usually much less compared with nonlinear controllers. In practice, it is always possible to obtain an approximate linear model for nonlinear systems [3], and it is not uncommon to find practical systems that utilize very sophisticated linear controllers, e.g. based on high order Kalman filters and linear quadratic regulator (LQR). This kind of control system design involves the model linearization and designing control system around the system operating point. A typical formula for linearization is based on Taylor series expansions at a given operating point. The classical approach for linearization and subsequent linear control systems design may therefore be used successful at the case of mildly nonlinear and small deviations from nominal steady state operating

conditions. However, the controller performance will deteriorate as the process moves further away from the operating point around which the model was linearized. The deterioration would be more serious for MIMO processes, in which the interactions between process variables usually display different characteristics with the change of operating points. For those nonlinear processes existing strong nonlinearities and strong interactions, the global behaviors of the systems may only be adequately described by nonlinear models over the wide operating range, rather than by linear ones that are only able to approximate the system around a given operating point. In such situations linear controller design techniques often prove inadequate and the more sophisticated alternatives must be considered.

Adaptive control methods

Adaptive control offers possible solutions to cope with nonlinearities by recognizing the presence of nonlinearities and consequently providing the controller a means for systematically adapting them [3]. In adaptive control methods, the controller parameters are continuously adjusted to keep up with the changes in the process dynamics and disturbances. Generally, there are two types of adaptive control schemes, i.e. direct and indirect methods [4]. In direct methods, controller parameters are adjusted directly from data obtained in closed-loop system operation. While in indirect methods, the parameters of a process model are updated online by recursive parameter estimation. There are a large number of schemes available for both direct and indirect methods, e.g. the model reference control and self-tuning regulator are typical examples of direct adaptive and indirect adaptive controllers, respectively. The well-known gain scheduling control method can also be classified as the direct adaptive control schemes [3, 5]. Although it is intuitively recognized that if properly designed, the adaptive control schemes will provide a significant performance improvement over the classical control schemes, there is still a lack of

sufficient practical applications in the process control industry. The major constraints for those approaches are that adaptive controllers often require huge computational effort, need a priori process information, and are difficult to guarantee the closed-loop stability over the wide working range.

Model based nonlinear control methods

Since 1980s', the general approaches of nonlinear model based control have attracted substantial research interests [6, 7]. Differential geometric approach [8] was proposed and considered as an effective tool for the analysis and design of nonlinear control systems. Similar to the way that Laplace transforms and linear algebra have been used for linear control system analysis and design, an understanding of the structural characteristics of nonlinear systems can be obtained using differential geometric concepts. Reference system synthesis (RSS) was another nonlinear control technique developed by using both process model and desired process response to determine a feedback control law. Lee and Sullivan [9] suggested a generic model control (GMC) method based on the RSS technique. Henson and Seborg [10] later pointed out that GMC and internal decoupling are implicitly based on differential geometric concepts. Several other nonlinear control techniques, such as, sliding-mode technique, a min-max optimization, feedback linearization, back stepping etc, have also appeared in the literature [11, 12]. Although model based nonlinear control techniques provide perspective solutions for the naturally nonlinear processes, they still lack general methodologies to design and evaluate the performance of the controlled systems. Furthermore, the stability can only be guaranteed if the processes can be exactly described by the nonlinear models. The availability of accurate nonlinear model and the heavy computational burden in solving nonlinear control problems are the main obstacles for the wide application of model based nonlinear control technologies. Effort to overcome these

weaknesses may have contributed recently to the significant developments of the nonlinear predictive control, which can be applied to more common nonlinear processes than those of other model based nonlinear control techniques [13, 14]. They are the extensions of linear model predictive techniques that have been successfully applied to a number of multivariable control problems in industry during the past decade. Constraints on manipulated (as well as state and output) variables and dead time are explicitly handled. These approaches are intuitive and easily understood by a process operator. The main restriction of those methods is that there is no convergence guarantee or rigorous robustness proof with respect to stability and performance.

Neural Networks control methods

Due to the rapid development of Neural Networks (NN) since 1980s', NN control has been regarded as one of the most promising methods in solving the complex nonlinear control problems [15]. The nonlinear functional mapping properties that based on the topology of multi-layered architecture and the learning mechanics are central to the use of NN in control systems. It stems from its theoretical ability to approximate arbitrary nonlinear mapping and hence to model nonlinear systems. Training a NN using input-output data from a nonlinear process can be considered as a nonlinear functional approximation problem. There are a large number of control structures proposed on NN architectures in the literature [16], some of the well developed schemes are: supervised control, where NN provides an alternative approach similar to the expert systems that used to provide the knowledge representation and control problem formalization; direct inverse control, where NN acts as the direct inverse control that utilizes an inverse system model; model reference control, where the error between plant output and the reference model output is used to train the NN controller to follow a desired closed-

loop system; Internal model control (IMC), where the NN models are placed for both the process model (system forward model) and the controller (the inverse model) to mimic the classic IMC method; predictive control, where a NN model is used to provide prediction of the future plant response over the specified horizon which are passed to a numerical optimization routine which attempts to minimize a specified performance criterion in the calculation of a suitable control signal. Although many control structures have been proposed, there is still a lack of systematic methods to choose the proper NN structure for different applications and to realize fast convergence of NN learning algorithms.

Fuzzy control methods

Since the invention of fuzzy set theory (Zadeh, 1965) [17], it has been rapidly developed and applied in wide variety of areas, such as database management, decision support, signal processing, data classifications, computer vision etc. Fuzzy systems that utilize human expertise by storing its essential components in the knowledge base, and perform fuzzy reasoning to infer the overall output value, are considered to be advantageous in the logical field. The first successful application in process control was reported in 1974 by Mamdani [18]. The early work in fuzzy control system design was motivated by a desire to directly express the control actions of an experienced human operator in the controllers, i.e. to mimic human behavior, and obtain smooth interpolation between discrete controller outputs. Subsequently, the applications of fuzzy control has widened substantially. Characteristic for fuzzy controllers is that they contain a static nonlinear mapping defined by a fuzzy inference system [19]. The rule-based formulation is intuitive and easy to be understood by engineers. Each rule represents local process knowledge on how the control signal should be selected for certain input signals. In most cases, a fuzzy controller is used for direct feedback control. However, it can

also be used on the supervisory level, e.g. a self-tuning device for a conventional PID controller. Since the fuzzy controller defines a nonlinear input/output mapping, it has a strong relationship with nonlinear control, they are sometimes designed to achieve feedback compensation or feedback linearization.

Neuro-fuzzy control methods

The combination of fuzzy logic and NN has a firm basis because they often complement each other in the design of intelligent decision making systems. NN provides an essentially low-level learning and computational power to process large amounts of data, while fuzzy logic which enables reasoning on a higher, linguistic or semantic level creates a structural framework that utilizes and interprets the low-level results coming from NN. This property makes the integrated systems to be more powerful than the pure fuzzy system or NN in the modeling and control of nonlinear dynamical systems. Research efforts that aim at synthesizing NN with fuzzy logic have appeared at end of the 1980s and experienced a big growth in 1990s with variety of structures and algorithms [20]. The aim of neuro-fuzzy controller is to find a structure that can effectively deal with two main types of data and information which usually contribute to the description of the decision making processes in control of complex systems, that is, qualitative and quantitative data. Fuzzy neural networks mix the NN with fuzzy logic in three ways [21]: cooperative, concurrent and fused. The most common architecture is the fused neuro-fuzzy approach that uses neural network to learn some internal parameters of a fixed structure [22]. The adaptive neuro-fuzzy inference system (ANFIS) belongs to this category [23] which is able to approach any linear or nonlinear functions. However, the neuro-fuzzy controller has not yet to generate big impact on the applications of complex process control industries. There are two possible reasons: 1) unacceptable efforts and difficulties in developing the complex models; 2) insignificant

achievable performance improvement compare with the conventional controllers.

Multi-model control methods

In solving complex engineering problems, a common approach is the *divide-and-conquer* strategy [24]. This strategy leads to the concept of multi-model techniques, which can be effective in dealing with the complex process with nonlinearity and time varying characteristics. Multi-model approach consists of different models and controllers under different operating conditions. This technique has been an appealing research field for complex nonlinear systems, and a variety of multi-model schemes have been developed to solve such problems [24-32]. The motivation behind this strategy is that global modeling is complicated because it will be necessary to describe the interactions among a large number of phenomena that appear globally. On the other hand, local models may be considerably simpler because locally there may be a smaller number of phenomena that are relevant, and their interactions are simpler [33]. Normally, it is necessary to know the definition of the full operating range when dividing the system's operating regimes. This information is then used to make a reasonable partition of a complex system over the full operating range to simplify the modeling and control problems. After the key variables to characterize the operating regimes are identified, simple local models and controllers are selected within each operating regime. The local models or controllers are usually determined according to the local knowledge under their corresponding operating conditions. This approach thus leads to several possibly overlapping operating regimes that local models or controllers are applied upon. Integrating the local models or controllers into the global structure can be obtained by a combination of the local model interpolations.

1.2 Motivations and Objectives

Although the increasing numbers of advanced control strategies have been proposed, many of them have been found difficult to be applied in solving real control problems for nonlinear complex MIMO processes. The major constraints are the effort and difficulty in developing a good nonlinear model that could be applied to a reasonably large number of nonlinear systems, and control decision usually involves a considerable amount of computations. Although nonlinear input-output models, such as NN and fuzzy logic, do not require much in terms of basic fundamental knowledge of the processes, they still need a great deal of process data training in order to describe the system behavior over a wide range of operating conditions [30]. Besides the complex nonlinearity, another challenge in the control system design is that the controlled processes are often MIMO systems involving a number of variables need to be controlled and a number of variables can be manipulated. Compared with Single-input Single-output (SISO) counterparts, MIMO systems are more difficult to control due to the interactions between input and output variables [34-36]. Although considerable efforts have been dedicated to solve this problem, MIMO control systems design and implementation are still of great challenge for control engineers due to the lack of systematic approach.

In most of MIMO control system, linear control strategies which provide powerful control design tools are used for complex processes under approximated linear models. However, the key limitation is that a single nominal model may not be able to represent a plant in the wide operating range when the plant dynamics exhibit significant nonlinearities. Unfortunately, most of the industry processes exhibit nonlinear and time varying characteristics, other factors such as load disturbance may also cause significant dynamic changes. Among the existing methods, the multi-model approach provides a promising way for wide industry

applications to solve such class of control problems. Therefore, the main objective of this thesis is to propose a systematic multi-model strategy to solve the nonlinear MIMO control problems. Since Two-input Two-output (TITO) processes account for majority of MIMO industry processes, we will concentrate our discussion in the TITO framework, and the results can be extended to other MIMO nonlinear processes with some modifications. To develop an effective multi-model control method, there are generally four basic problems need to be resolved:

- ♦ Construction of the local models.
- ♦ Design of the local controllers.
- ♦ Partition of operating regimes.
- ♦ Determination of overall control output.

These issues are briefly discussed as follows.

Identification of local models

Among the existing multivariable control techniques, many of them such as inverse Nyquist array or characteristic locus methods require a full model of the process in the form of a transfer-function matrix or a frequency-response matrix over the entire working frequency range [37-39]. In many cases, such a model is not available and physical modeling may require a prohibitive engineering effort. Therefore, practical and effective estimation of the process models by means of experimental tests performed online becomes appealing and has been an active research area of control engineering for a few decades. There are two ways to identify the parameters of a multivariable process for control application, i.e. open-loop and closed-loop identifications. In any case, an external excitation of the process is needed to extract useful information on process dynamics. For open-loop

transient response experiments, step or pulse excitation signals are commonly injected at the process inputs, and the responses are measured [40–42]. The open-loop identification, despite its apparent simplicity, has several drawbacks: 1) the process deviation from operating condition cannot be controlled; 2) the result may be inconsistent for the case of nonlinear process [43]. For closed-loop identification, a majority of existing techniques are in the frequency domain while the frequency range of interest for such applications is usually from zero up to the process critical frequency [44–48]. In closed-loop condition, a controller is active and thus allows limiting the amount of deviation from operating conditions.

Since close-loop identification is preferred to open-loop one in process control systems [49], it is of practical interest to identify the local models for certain operating range under the closed-loop control structure. So far, a practical method to effectively obtain the model is still lacking [50–55]. The major challenges for closed-loop model identification are 1) obtaining the frequency response for each input-output correlation; 2) finding the critical point for each transfer element; 3) determining the model parameters for each transfer function element. In this thesis, a practical closed-loop identification method is proposed which can obtain the accurate model parameters according to the critical point of individual process by a very simple procedure.

Design of local controllers

For control system design, there are mainly two multivariable control techniques used in the process industry. The first one is the decentralized control, which uses a set of SISO controllers to control the MIMO processes. This control technique is a straightforward extension of single loop control to multivariable systems. The second strategy is a centralized or coordinated control, which utilizes a single control algorithm combining all the variables simultaneously to calculate all

the required values of the manipulated variables such that the desired control performance can be achieved. Compared with the centralized control systems, the decentralized control system has certain advantages: 1) fewer parameters are required to be tuned; 2) much easier to be understood and implemented; 3) loop failure tolerance of the resulting control system can be assured; and 4) hardware simplicity, requires less hardware devices. Therefore, the decentralized control systems are more popular in real MIMO process control applications [56, 57].

After the local model has been obtained, the local controller design is the second important step on multi-model control scheme. Although decentralized PI/PID control systems with simpler control structures are more popular in industry applications, they are more difficult in term of design compared with centralized methods. To design the decentralized control systems, the first step is to properly pair the manipulated variables and controlled variables to achieve the minimum interactions among loops [36]. The Relative Gain Array (RGA) based techniques for control loop configuration have been widely used with the obvious advantage of less calculation [58]. However, this method only uses steady state gain, and may result in incorrect interaction measures and consequently loop pairing decisions. On the contrary, Dynamic RGA (DRGA) measure was proposed with the consideration of process dynamics [59]. Although the process dynamics is considered by DRGA through employing the transfer function model, its difficulty in calculation, being controller dependent and hard to be understood by practical control engineers make it hard to be applied in real application.

In this thesis, a practical loop pairing criterion in terms of Effective RGA (ERGA) is proposed, which combines the advantages of RGA and DRGA by using the steady state gain and the bandwidth of the process transfer function matrix. Based on the proposed ERGA, an effective engineering oriented controller tuning

formula is developed by using Ziegler-Nichols or other tuning methods for SISO systems. As classical decentralized PID control design techniques may not provide an optimal performance due to the structure limitations, it is of practical importance to tune the decentralized PID controller for better performance. In order to further improve the decentralized control system performance, the multi-step horizon optimization is applied for fine tuning of the controller parameters.

Partition of operating regimes

How to determine the right number of operating regimes and define the corresponding boundaries for each operating regime is another important issue to be resolved for multi-model control system design. Although the existing multi-model approaches have the advantage of efficiency and transparency for learning, it is difficult to automatically find the proper number of local models or controllers. Actually, it is hard to use a theoretical guideline to solve this problem. Normally, the operating regimes of the complex systems are partitioned based on the expert knowledge of some scheduled variables, such as control signal, process variable, or an external signal etc. Such division requires that the underlying nonlinearities of the practical system and some of its physical parameters are known in a priori. However, as many of the nonlinearities cannot be modeled accurately due to the system complexity and constraints on physical device for measuring, this kind of partition may not be appropriate. Thus, how to partition the complex system operating regime with less process knowledge is very important. Since our objective is to control MIMO processes, the controller performance will be used to help to determine how to divide the whole operating range. An incremental strategy is proposed in the thesis to construct the multiple local models and controllers. That is, an additional local model and controllers are added to the multi-model control system only if the unsatisfied performance is detected. In this method, the strategy

of constructing a global control system is guided by a supervisor, which decides the operating range division and local controllers combination.

Determination of overall control output

After the right number of operating regimes has been determined, and the corresponding local models and controllers have been obtained and designed respectively, the remaining task is to determine the overall controller output by means of combining local decentralized controller outputs. In real process operation, since the operating point may be at any location within the operating range, a proper combination of local controller outputs is needed to determine the global output. When the operating point is located between different operating regimes, the fuzzy interpolation provides a reasonable solution as process nonlinearities between different operating regimes usually change gradually, i.e. there exists a vague regime between different regimes. Consequently, the fuzzy combination of local operating regimes through proper membership function definition can achieve smooth behavior across all operating regions. In the thesis, a fuzzy integration method is proposed to combine the local controller output as the overall process input.

In summary, we aim at developing a multi-model based approach to deal systematically and effectively with MIMO processes with complex nonlinear characteristics.

1.3 Major Contributions of the Thesis

In this thesis, a systematic approach for decentralized multi-model control systems is proposed. It provides an effective way to cope with the situations where

single model controller cannot obtain a satisfactory performance for complex nonlinear MIMO processes over a wide working range. The main contributions of the thesis can be summarized as follows.

1. Local model identification

An engineering oriented local model identification techniques for MIMO processes is presented. The significance of the proposed method is that it relaxes most restrictions of existing multivariable process identification methods, and it is applicable to closed-loop identification for MIMO processes. It includes three major parts:

1. The closed-loop system is excited by two sequential step tests. The coupled closed-loop TITO is decoupled equivalently into four independent single open-loop processes with same input signal acting on the four transfer functions.
2. An optimal search method - Min-Max Critical Frequency Search (Min-Max CFS) method is proposed. Making use of estimation results of transfer functions with lower critical frequencies by the transfer functions with higher critical frequencies, the calculation burden has been greatly reduced.
3. Least square algorithm combined with the sequential step tests to obtain the parameters of transfer function elements is presented, which is simple and accurate.

2. Local controller design

For the widely applied decentralized control structure, a new loop pairing criterion and consequently, a practical controller design method are proposed. The

initial controller obtained can be further fine tuned by employing the concepts of multi-step horizon optimization and realizable references. It includes three major parts:

1. A new loop pairing criterion in terms of ERGA is proposed. The proposed ERGA combines the advantages of RGA and DRGA by applying the steady state gain and the bandwidth of the process transfer function matrix elements to provide a more comprehensive yet simple description for loop interaction measure and rule for loop pairing.
2. A practical controller tuning formula is developed based on the proposed ERGA, which makes use of Ziegler-Nichols tuning or other tuning formulae for the non-interactive SISO system plus detuning factor evaluated by ERGA.
3. A two-level PID controller tuning structure based on multi-step receding horizon optimization is presented. By formulating the decentralized PID control problem in forms of generalized predictive control form and introducing a realizable reference for the decentralized control structure, an effective algorithm is proposed for fine tuning of the controllers designed by ERGA.

3. Global multi-model construction

Multi-model constructing strategy based on controller performance monitoring is proposed. A supervisor is defined to coordinate the whole procedure. The proposed strategy is to divide the whole operating range according to the current controller performance, which is more suitable for control of complex nonlinear MIMO system. The main work includes:

1. The incremental way of adding local regimes is applied to partition the operating regime according to the controller performance. The fuzzy partition is defined according to the validity function of local regimes, and the membership function are varied as the additional local one is added to the global system for different cases.
2. A Model Predictive Control (MPC) like cost function is introduced to the multi-model control system to monitor the control system performance. It offers a criterion to evaluate the controller performance online to determine whether additional local regime is needed.
3. The method for combination of local controllers through the fuzzy integrations of the involved local regimes is proposed to realize a smooth behavior between different operating regimes. Reasonable fuzzy weights redistributions of local regimes are discussed for different situations.

4. HVAC application

Heating, Ventilation and Air-conditioning (HVAC) systems are used in almost every building such as residential, commercial, institutional, and industrial buildings. HVAC processes are very complicated, involving several functional blocks and each functional block can be formulated by a MIMO nonlinear dynamic model. Good dynamic control will contribute to the reduction of energy and operating costs which have gained considerable importance in the minds of building owners and plant operators in recent years. In this thesis, the proposed multi-model control method is applied to control two processes of HVAC systems:

1. Control of single-zone HVAC process: Based on a widely used single-zone HVAC nonlinear MIMO model, the proposed multi-model approach is applied to achieve a satisfactory controller performance of conditioned

space temperature and relative humidity for a wide operating range.

2. Control of air temperature of two zones: A two-zone air temperature control application is tested in a pilot central HVAC system. Based on the proper division of operating regime, a satisfactory controller performance is achieved.

1.4 Organization of the Thesis

The rest of thesis is organized as follows:

Chapter 2 presents a practical identification technique for closed-loop MIMO processes. It is used to construct local models for the multi-model control system.

Chapter 3 presents a new loop pairing criterion in terms of ERGA, and a simple decentralized PID controller tuning formula from single loop tuning methods based on the proposed ERGA.

Chapter 4 presents a two-level PID controller fine tuning technique based on multi-step receding horizon optimization.

Chapter 5 presents an applicable strategy to construct the global system, involving performance evaluation, operating regimes division and overall control integration.

Chapter 6 presents real MIMO process control applications for HVAC systems based on the proposed multi-model control method.

Chapter 7 concludes the thesis and provides some possible future research directions for decentralized MIMO control systems.

Chapter 2

Closed-loop Parameter Identification in Frequency Domain

2.1 Introduction

To apply multi-model control techniques for MIMO processes, a key issue is the determination of the local dynamic models. Since the step test has its advantage of requiring simple test procedure and little prior process knowledge, it is the dominant test method in process control applications. Therefore, the development of a simple, yet accurate dynamic model of unknown processes through the step responses is critical to the success of control systems. For single loop systems, many identification methods are proposed by using such a test to determine the

dynamics of unknown processes. Bi and Cai, *et al.* [60] proposed a simple but robust identification method to obtain a first-order plus dead-time model for a linear monotonic process from a step test of open-loop control systems. The identification method was later applied to design auto-tuning PID controllers for HVAC systems [61], experimental results have demonstrated the effectiveness of the technique. Wang and Cluett [62] presented an identification algorithm for processes operating in closed-loop process, which involves fitting two Laguerre models directly to the control signal and the process output signal generated by a step change in the set-point. Wang *et al.* [42, 63] proposed some robust identification methods for linear time-delay processes from step responses in both time domain and frequency domain.

For multivariable closed-loop identification, a straightforward extension is the relay feedback test to multivariable processes including sequential loop test [51] and simultaneous loop test [52, 53]. Unlike the single loop counterpart, however, these methods has certain drawbacks when applied to multivariable system: 1) the combined dynamics are overlooked in the sequential loop test; 2) the critical point is generally difficult to be found in the simultaneous loop test; 3) Both tests only partially identify the frequency-response matrix at one or two points, which are insufficient to model the process accurately. Some other interesting works for closed-loop identification are based on set-point changes during closed-loop P or PID control operation [45, 54]. Unfortunately, most of the work did not extend to multivariable systems. Melo and Friedly [55] proposed a method to determine the open-loop frequency response in multivariable system during closed-loop PID control operation which needs prior knowledge of the controller dynamics and without providing the transfer function matrix. Wang and Zhang [50] provided a general methodology using relay or step tests for multivariable process identification, however, it again may suffer some drawbacks in engineering

applications: 1) For relay-test on which the method mainly focused, the transients instead of stationary oscillations in the relay responses are adopted to identify the multiple frequency responses, which may results in the poor estimation of frequency response matrix for multivariable process despite the fact that it works for single loop process; 2) In order to obtain the transfer function, the process step response has to be reconstructed first from the estimated frequency response using IFFT and then the second-order parameter model are identified from the reconstructed step response in time domain, which makes the computation and procedure complicated and difficult for application of on-line identification and controllers auto-tuning. Recently, Wang and Cai [46] proposed an improved method to identify frequency-response matrix at multiple points over the important frequency range on the basis of the decentralized relay feedback, in which only the very first periods unsymmetrical two-side pulse signals of n oscillating inputs are implemented. The two-side pulse waveform contains continuous frequency spectrum over the important frequency range. The process frequency-response matrix can be estimated with *Fourier* analysis method. Since the method uses the transient oscillations, it again may result in the poor estimation of frequency response matrix.

In this chapter, we present an engineering oriented identification technique for multivariable process, especially, for a decentralized TITO closed-loop control system. By two set-point changes tests, the frequency response in the range of interest corresponding to π phase lag is estimated; and the parameters of transfer function matrix are identified by matching the model in the frequency domain. Furthermore, in order to reduce the computational burden, the coupled closed-loop TITO system is decoupled equivalently into four individual single open-loop processes with the same input signal acting on the four transfer functions. Consequently, an optimal search method - Min-Max CFS method is proposed. The proposed identification method only requires response data of process input and

output, and no prior knowledge of the process dynamics and of the controller dynamics is needed. Various typical multivariable processes have been employed to illustrate the effectiveness of the method. It is ready for use in retuning an existing multivariable control system and designing advanced controllers for multivariable processes.

2.2 Step Tests for TITO System Identification

Consider a TITO process under decentralized control as shown in Fig. 2.1,

Fig. 2.1 Closed-loop TITO control system

where r_i , e_i , u_i , y_i , K_i , and G_{ij} , $i, j = 1, 2$, are set points, errors, controller outputs, process outputs, controllers, and process transfer functions, respectively. The fundamental relationship between controller output signals and transfer function outputs for the process are expressed as, $y(s) = G(s)u(s)$, where

$$y(s) = \begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix}, u(s) = \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix}, G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix}.$$

Assume that the process initially rest at a steady state with initial set-point, $r_1^0, r_2^0, y_1^0, y_2^0, u_1^0$ and u_2^0 . To identify the processes, the existing decentralized PID

controllers are only needed to guarantee the closed-loop system to be stable, the step tests involve the following two steps:

1. Make a step change to r_1 from r_1^0 to r_1^1 , with r_2 kept unchanged, record the controller outputs and process outputs signals for the two loops, until the new steady state is reached at $t \triangleq t_1 = T > t_s$ (see Fig. 2.2), t_s is the maximum settling time of all transfer functions, which can be observed from system responses. The incremental equation from initial state to the new state becomes

$$\Delta y_1^1(s) = G_{11}(s)\Delta u_1^1(s) + G_{12}(s)\Delta u_2^1(s) \quad (2.1)$$

$$\Delta y_2^1(s) = G_{21}(s)\Delta u_1^1(s) + G_{22}(s)\Delta u_2^1(s) . \quad (2.2)$$

where $\Delta u_1^1(s) = u_1^1(s) - u_1^0(s)$, $\Delta u_2^1(s) = u_2^1(s) - u_2^0(s)$.

2. Make a step change to r_2 from r_2^0 to r_2^1 , while keeping $r_1 = r_1^1$ as before, record controller outputs and process outputs for the two loops, until the new steady state is reached at $t \triangleq t_2 = 2T > 2t_s$. Again, the incremental equation from state (1) to the new state can be written as

$$\Delta y_1^2(s) = G_{11}(s)\Delta u_1^2(s) + G_{12}(s)\Delta u_2^2(s) \quad (2.3)$$

$$\Delta y_2^2(s) = G_{21}(s)\Delta u_1^2(s) + G_{22}(s)\Delta u_2^2(s) \quad (2.4)$$

where, $\Delta u_1^2(s) = u_1^2(s) - u_1^1(\infty)$, $\Delta u_2^2(s) = u_2^2(s) - u_2^1(\infty)$, $u_1^1(\infty)$, $u_2^1(\infty)$ are the frequency values of controller outputs when the steady state is achieved during the first step and are the initial values of the second step test.

Remark 2.1: During the testing, step change for each set point is performed sequentially as shown in Fig. 2.2, where t_s is defined as the maximum time required

for all outputs to settle at steady state under the two testing.

Remark 2.2: By using incremental signals, zero initial conditions for the equivalent decentralized identification system are guaranteed, that is $u = y_{ij} = 0$, $t \leq 0$, for $i, j = 1, 2$, regardless the value of initial steady state errors.

Fig. 2.2 Settling time for parameter identification

Since a signal $f(t)$ varying from one steady state to another steady state can be decomposed into transient part $\Delta f(t)$ and stationary part $f_s(t)$

$$\begin{aligned}\Delta f(t) &= f(t) - f(\infty) \\ f_s(t) &= f(\infty) \cdot 1(t)\end{aligned}$$

where $1(t)$ is unit step function. Consequently, the function $f(t)$ can be expressed as

$$f(t) = \Delta f(t) + f_s(t).$$

As transient part $\Delta f(t)$ will decay to zero after certain time, the *Laplace*-transform $F(s)$ of $f(t)$ can then be expressed as

$$F(s) = \Delta F(s) + F_s(s) = \int_0^{\infty} \Delta f(t) e^{-st} dt + \frac{f_s(\infty)}{s} \quad (2.5)$$

where $\Delta F(s)$ and $F_s(s)$ are the *Laplace* transforms of the transient part $\Delta f(t)$ and the stationary part $f_s(t)$ respectively. Suppose that at $t = T_f$, $f(t)$ have reached the steady state value and $\Delta f(t)$ is approximately zero after $t = T_f$, then (2.5) becomes

$$F(s) \approx \int_0^{T_f} \Delta f(t) e^{-st} dt + \frac{f_s(\infty)}{s} \quad (2.6)$$

where (2.6) can be computed by the standard *Fast Fourier Transform* (FFT) technique,

$$F(j\omega) \approx FFT(\Delta f(t)) + \frac{f_s(\infty)}{j\omega}. \quad (2.7)$$

Now, write Equations (2.1-2.4) into a matrix form

$$Y(s) = G(s)U(s) \quad (2.8)$$

where

$$Y(s) \triangleq \begin{bmatrix} \Delta y_1^1(s) & \Delta y_1^2(s) \\ \Delta y_2^1(s) & \Delta y_2^2(s) \end{bmatrix}, G \triangleq \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \text{ and } U \triangleq \begin{bmatrix} \Delta u_1^1(s) & \Delta u_1^2(s) \\ \Delta u_2^1(s) & \Delta u_2^2(s) \end{bmatrix}$$

As signals $\Delta y_1^1(t)$, $\Delta y_2^1(t)$, $\Delta y_1^2(t)$, $\Delta y_2^2(t)$, $u_1^1(t)$, $u_2^1(t)$, $u_1^2(t)$ and $u_2^2(t)$ are obtained from the two tests, applying the result of Equation (2.7) to Equation (2.8), G_{ij} , $i, j = 1, 2$ are given as

$$G(s) = Y(s)U(s)^{-1} \Rightarrow \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} = \begin{bmatrix} \Delta y_1^1(s) & \Delta y_1^2(s) \\ \Delta y_2^1(s) & \Delta y_2^2(s) \end{bmatrix} \begin{bmatrix} \Delta u_1^1(s) & \Delta u_1^2(s) \\ \Delta u_2^1(s) & \Delta u_2^2(s) \end{bmatrix}^{-1} \quad (2.9)$$

Suppose that the number of the frequency response to be identified in the frequency range $(0, \omega_u)$ is M , the process frequency response $G(j\omega_m)$ at the discrete

frequencies $0, \Delta\omega, 2\Delta\omega, \dots, (M-1)\Delta\omega$, where $\Delta\omega$ is the frequency increment and the transfer function of the TITO system can be identified by Equation (2.9). However, there are four transfer functions to be identified with the frequency range of interest for each transfer function is from zero up to the process critical frequency $\omega_c^{ij}, i, j = 1, 2$ for control system design and analysis, the use of Equation (2.9) with predetermined M and $\Delta\omega$ will face some difficulties:

- 1) Since there are four transfer functions in the TITO processes, and the frequency range of interest for each transfer function is from zero up to the process critical frequency, the cut off frequency ω_u is very difficult to determine. Too big a ω_u will result in an unnecessary computational burden, while a small ω_u may result in insufficient useful frequency information for the fast response transfer function elements.
- 2) Even if ω_u can be properly determined, the selection of M , consequently, $\Delta\omega$ is still a difficult task. Again, a large M will result in an unnecessary computational burden, while a small M may result in insufficient useful frequency information for the slow response transfer function elements.

Therefore, an effective identification algorithm is necessary to make the method useful, and it will be developed in the next section.

2.3 Decentralized Frequency Domain Identification

From Equation (2.8), each individual transfer function can be explicitly obtained as

$$\begin{cases} G_{11}(s) = \frac{\Delta y_1^1 \Delta u_2^2 - \Delta y_1^2 \Delta u_2^1}{\Delta u_1^1 \Delta u_2^2 - \Delta u_1^2 \Delta u_2^1} \triangleq \frac{Y_{11}(s)}{U(s)} \\ G_{12}(s) = \frac{\Delta y_1^2 \Delta u_1^1 - \Delta y_1^1 \Delta u_1^2}{\Delta u_1^1 \Delta u_2^2 - \Delta u_1^2 \Delta u_2^1} \triangleq \frac{Y_{12}(s)}{U(s)} \\ G_{21}(s) = \frac{\Delta y_2^1 \Delta u_2^2 - \Delta y_2^2 \Delta u_2^1}{\Delta u_1^1 \Delta u_2^2 - \Delta u_1^2 \Delta u_2^1} \triangleq \frac{Y_{21}(s)}{U(s)} \\ G_{22}(s) = \frac{\Delta y_2^2 \Delta u_1^1 - \Delta y_2^1 \Delta u_1^2}{\Delta u_1^1 \Delta u_2^2 - \Delta u_1^2 \Delta u_2^1} \triangleq \frac{Y_{22}(s)}{U(s)} \end{cases} \quad (2.10)$$

It is easy to show that the necessary and sufficient conditions for the existence of U inverse are:

1. The decentralized controllers satisfy $G_{c1}(s) \neq 0$, and $G_{c2}(s) \neq 0$ for the frequency of interest; and
2. The system is bounded input bounded output (BIBO) stable, i.e.

$$1 + G_{c1}(s)G_{11}(s) + G_{c2}(s)G_{22}(s) + G_{c1}(s)G_{c2}(s)[G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)] < \infty$$

In terms of parameter identification, the coupled closed-loop TITO system is explicitly decoupled into four individual single open-loop systems with same input signal acting on the four transfer functions as shown in Fig. 2.3. Consequently, the identification of coupled closed-loop TITO system is decoupled to the identification problem of multi-single open-loop systems.

Fig. 2.3 Decentralized identification process

Now consider an arbitrary single loop transfer function $G(j\omega)$, the gradient search method can be used to iteratively calculate the frequency response of $G(j\omega)$ from zero up to the process critical frequency ω_c corresponding to the process phase lag π for control system analysis and design. The formulae for the gradient search is given as

$$\omega_{m+1} = \omega_m - (\phi^* + \phi_m) \frac{\omega_m - \omega_{m-1}}{\phi_m - \phi_{m-1}} \quad (2.11)$$

where $\phi^* = \pi$ is the desired phase lag and

$$\phi_m = \arg[G(j\omega_m)] \quad (2.12)$$

Due to the quadratic convergence rate of Equation (2.11), ω_m is very close to the desired critical frequency ω_c corresponding to phase lag $\phi^* = \pi$ only after a couple of steps for the given initial values as the iterative formula (2.11) has a quadratic convergence rate near the solution. This has been shown by our extensive simulation studies. In order to obtain more accurate result over the entire frequency range of interest, we let the number of the frequency response to be identified in the frequency range $(0, \omega_c)$ is M , and divide the desired phase lag π into M subintervals with difference of $\frac{\pi}{M-1}$ as shown in Fig. 2.4.

Fig. 2.4 Desired frequencies in the important frequency range

Then the following algorithm can be used to calculate $G(j\omega_m)$ near the desired discrete frequencies:

Algorithm 2.1:

Step 1: For initial values ω_0 and ϕ_0 are zeros, ω_1 is chosen to be a value smaller than the desired frequency ω_2^* , say $\omega_1 = 10^{-3}$, and ϕ_1 is calculated by (2.12).

Step 2: Obtain the frequency ω_2 by

$$\omega_2 = \omega_1 - \left(\frac{\pi}{M-1} + \phi_1 \right) \frac{\omega_1 - \omega_0}{\phi_1 - \phi_0}$$

which is very close to the desired frequency ω_2^* corresponding to phase lag $\frac{\pi}{M-1}$,

then calculate $G(j\omega_2)$ and corresponding ϕ_2 .

Step 3: For $m = 2, 3, \dots, M-1$, determine ω_{m+1} , $G(j\omega_{m+1})$ and ϕ_{m+1} by

$$\omega_{m+1} = \omega_m - \left(\frac{m\pi}{M-1} + \phi_m \right) \frac{\omega_m - \omega_{m-1}}{\phi_m - \phi_{m-1}} \quad (2.13)$$

Step 4: Stop

Remark 2.3 Since the objective of Equations (2.12) and (2.13) is to make the discrete frequencies more or less evenly distributed with respect to phase lag over the important frequency range in order to obtain an accurate process transfer function model. The closeness of the calculated discrete frequencies to their respective desired frequencies is not an important issue here.

Applying Algorithm 2.1 to the decoupled TITO process of Equation (2.10), the frequency response of each transfer function $G_{ij}(j\omega)$, $i, j = 1, 2$ can be determined by

$$\omega_{m+1}^{ij} = \omega_m^{ij} - \left(\frac{m\pi}{M-1} + \phi_m^{ij} \right) \frac{\omega_m^{ij} - \omega_{m-1}^{ij}}{\phi_m^{ij} - \phi_{m-1}^{ij}} \quad (2.14)$$

and

$$\phi_m^{ij} = \arg[G(j\omega_m^{ij})] \quad (2.15)$$

for $i, j = 1, 2$ and $m = 1, 2, \dots, M$.

Since each transfer function element may have a different critical frequency ω_c^{ij} , individual determination of frequency response of each transfer function $G_{ij}(j\omega)$, ($i, j = 1, 2$) over the frequency range $(0, \omega_c^{ij})$, according to Equation (2.10) and Algorithm 2.1, may still require extensive computation since $\Delta y_1^1(t)$, $\Delta y_2^1(t)$, $\Delta y_1^2(t)$, $\Delta y_2^2(t)$, $u_1^1(t)$, $u_2^1(t)$, $u_1^2(t)$, $u_2^2(t)$ have to be repeatedly calculated for each transfer function element and at different frequency points. To reduce the computational burden, we here propose a Min-Max CFS method. Let the minimal number of points and critical frequencies for TITO process be M and ω_c^{11} , ω_c^{12} , ω_c^{21} , ω_c^{22} , respectively, the working principle of Min-Max CFS method is explained as follows.

1. Minimal critical frequency search: in order to guarantee the transfer function with minimal critical frequency has M number of frequency points. The min-pace is used to set the search pace at every iteration step. Consider the m^{th} step, we have the 4 phase angles from Equation (2.15), say, ϕ_m^{11} , ϕ_m^{12} , ϕ_m^{21} , ϕ_m^{22} , and the minimum one is used to compute the step size for the next iteration. This min-pace search iterated until the transfer function $G_{ij}(j\omega)$ with minimal critical frequency $\omega_{c,\min}$ reaches phase lag π .

2. Maximum critical frequency search: in order to minimize the iterating steps after the first M points have already been calculated for all transfer functions, max-pace

step size is applied for the remaining search. In this search, the iteration points is revised based on the maximum phase angle of the remaining elements at the frequency $\omega_{c,\min}$, say

$$M = \text{round} \left(M - \frac{\min |\phi_{\omega_{c,\min}}^{ij}|}{\pi} (M - 1) \right)$$

$\text{round}(arg)$ denotes a mathematical operator rounding the argument arg to the nearest integer. The maximum phase angles is used to compute the next step in every iteration until the element $G_{ij}(j\omega)$ with maximal critical frequency range reaches the phase lag π .

3. Stop Criterion: As already mentioned in Remark 2.3, the frequency response identification is used for modeling and control system design, the closeness of the phase lag to π is not critical. At any iteration step, once the phase of a transfer function reaches, say, $\left((\pi + \phi_{\omega_m}^{ij}) < 0.1\pi \right)$, the calculation of the particular transfer function frequency response is terminated.

The calculation procedure can be summarized as Algorithm 2.2.

Algorithm 2.2:

Step 1: Minimal critical frequency search:

1. Specify the initial number M as the frequency response points to be identified,
2. Apply Algorithm 2.1 with Equations (2.10) and (2.15) to determine $G_{ij}(j\omega_m)$, ϕ_m^{ij} , and $\phi_m = \min(\phi_m^{ij})$ for the transfer function with minimal critical frequency $\omega_{c,\min}$.

3. Check any other transfer functions $G_{ij}(j\omega_m)$ satisfy $\left((\pi + \phi_{\omega_m}^{ij}) < 0.1\pi\right)$, if yes, stop it (them) from the iteration.

Step 2: Maximum critical frequency search:

1. Calculate the iteration number of remaining transfer functions by

$$M = \text{round} \left(M - \frac{\min |\phi_{\omega_{c,\min}}^{ij}|}{\pi} (M - 1) \right).$$

2. For initial values ω_0 and ϕ_0 are zeros, let $m = 1, 2 \dots M$, $\omega_1 = \omega_m$ and

$$\phi_1 = \phi_{m,\min}^{ij} \text{ and define } \phi_m = \max(\phi_m^{ij}).$$

3. Apply Algorithm 2.1, Equations (2.10) and (2.15) to determine $G_{ij}(j\omega_m)$,

$$\phi_m^{ij} \text{ for the maximal critical point } \omega_{c,\max}.$$

4. If any transfer functions $G_{ij}(j\omega_m)$ satisfy $\left((\pi + \phi_{\omega_m}^{ij}) < 0.1\pi\right)$ or phase lag larger than π , stop it from the iteration.

Step 3: Stop.

Remark 2.4: The purpose of the proposed algorithm is to make use of the calculation result of the transfer functions with minimal and maximum critical frequencies by the transfer functions with critical frequencies in between. Since the closeness of the calculated discrete frequencies to their respective desired frequencies is not an important issue here, the revised Algorithm 2.2 can significantly reduce the computational efforts compared with the direct application of Algorithm 2.1.

2.4 Transfer Function Modeling and Identification Procedure

As every transfer function from the process frequency response $G_{ij}(j\omega_m)$, $i, j = 1, 2$, $m = 1, 2, \dots, M_{ij}$, has a different frequency range, we can obtain each transfer function of the process independently. As first or second-order plus time delay models can approximate monotonic, oscillatory and non-minimum phase processes which cover almost all industrial applications, the following development will concentrate on these two types of models.

A. First-order model

The first-order plus dead-time model is adopted for the transfer function:

$$G'_{ij}(s) = \frac{b_{ij0}}{a_{ij1}s + 1} e^{-L_{ij}s} \quad (2.16)$$

Its parameters may be determined by matching $G'_{ij}(j\omega)$ to $G_{ij}(j\omega)$ at multiple points ω_m , $m = 1, 2, \dots, M_{ij}$, that means

$$G_{ij}(j\omega_m) = \frac{b_{ij0}}{j\omega_m a_{ij1} + 1} e^{-j\omega_m L_{ij}}, i, j = 1, 2, m = 1, 2, \dots, M_{ij} \quad (2.17)$$

The parameters a_{ij1}, b_{ij0} and L_{ij} in (2.16) are determined by both magnitude and phase conditions of (2.17).

For the frequency ω_m , the magnitude and phase values of (2.16) are given as

$$-\omega_m^2 |G_{ij}(j\omega_m)|^2 a_{ij1}^2 + b_{ij0}^2 = |G_{ij}(j\omega_m)|^2 \quad (2.18)$$

and

$$-\arg[G_{ij}(j\omega_m)] - \tan^{-1}(a_{ij1}\omega_m) = \omega_m L_{ij} \quad (2.19)$$

for $m=1,2,\dots,M_{ij}$, the magnitude equation of (2.18) can be arranged in a matrix form

$$\Psi_{ij} \Theta_{ij} = \Gamma_{ij} \quad (2.20)$$

where

$$\Psi_{ij} = \begin{bmatrix} -\omega_1^2 |G_{ij}(j\omega_1)|^2 & 1 \\ -\omega_2^2 |G_{ij}(j\omega_2)|^2 & 1 \\ \vdots & \vdots \\ -\omega_{M_{ij}}^2 |G_{ij}(j\omega_{M_{ij}})|^2 & 1 \end{bmatrix}, \quad \Gamma_{ij} = \begin{bmatrix} |G_{ij}(j\omega_1)|^2 \\ |G_{ij}(j\omega_2)|^2 \\ \vdots \\ |G_{ij}(j\omega_{M_{ij}})|^2 \end{bmatrix} \quad (2.21)$$

$$\Theta_{ij} = \begin{bmatrix} \theta_{ij1} \\ \theta_{ij2} \end{bmatrix} = \begin{bmatrix} a_{ij1}^2 \\ b_{ij0}^2 \end{bmatrix}. \quad (2.22)$$

B. Second-order model

The second-order plus dead-time model is adopted for the transfer function

$$G'_{ij}(s) = \frac{b_{ij0}}{a_{ij2}s^2 + a_{ij1}s + 1} e^{-L_{ij}s} \quad (2.23)$$

which can represent more complicated processes, such as monotonic, oscillatory and non-minimum phase processes. Its parameters may be determined by matching $G'_{ij}(j\omega)$ to $G_{ij}(j\omega)$ at multiple points ω_m , $m=1,2,\dots,M_{ij}$, that means

$$G_{ij}(j\omega_m) = \frac{b_{ij0}}{(j\omega_m)^2 a_{ij2} + j\omega_m a_{ij1} + 1} e^{-j\omega_m L_{ij}} \quad (2.24)$$

$i, j = 1, 2, m = 1, 2, \dots, M_{ij}$

The parameters $a_{ij1}, a_{ij2}, b_{ij0}$ and L_{ij} in (2.23) are determined by both magnitude and phase conditions of (2.24).

For the frequency ω_m , the magnitude and phase values of (2.24) are given as

$$-\omega_m^4 |G_{ij}(j\omega_m)|^2 a_{ij2}^2 - \omega_m^2 |G_{ij}(j\omega_m)|^2 (a_{ij1}^2 - 2a_{ij2}) + b_{ij0}^2 = |G_{ij}(j\omega_m)|^2 \quad (2.25)$$

and

$$-\arg[G_{ij}(j\omega_m)] - \tan^{-1}\left(\frac{a_{ij1}\omega_m}{1 - a_{ij2}\omega_m^2}\right) = \omega_m L_{ij} \quad (2.26)$$

for $m=1,2,\dots,M$, the magnitude equation of (2.25) can be arranged in a matrix form as (2.20) with

$$\Psi_{ij} = \begin{bmatrix} -\omega_1^4 |G_{ij}(j\omega_1)|^2 & -\omega_1^2 |G_{ij}(j\omega_1)|^2 & 1 \\ -\omega_2^4 |G_{ij}(j\omega_2)|^2 & -\omega_2^2 |G_{ij}(j\omega_2)|^2 & 1 \\ \vdots & \vdots & \vdots \\ -\omega_M^4 |G_{ij}(j\omega_M)|^2 & -\omega_M^2 |G_{ij}(j\omega_M)|^2 & 1 \end{bmatrix}, \Gamma_{ij} = \begin{bmatrix} |G_{ij}(j\omega_1)|^2 \\ |G_{ij}(j\omega_2)|^2 \\ \vdots \\ |G_{ij}(j\omega_M)|^2 \end{bmatrix} \quad (2.27)$$

and

$$\Theta_{ij} = \begin{bmatrix} \theta_{ij1} \\ \theta_{ij2} \\ \theta_{ij3} \end{bmatrix} = \begin{bmatrix} a_{ij2}^2 \\ a_{ij1}^2 - 2a_{ij2} \\ b_{ij0}^2 \end{bmatrix}. \quad (2.28)$$

Parameters Θ_{ij} in Equations (2.22) or (2.28) can be solved using the linear least squares method

$$\tilde{\Theta}_{ij} = (\Psi_{ij}^T \Psi_{ij})^{-1} \Psi_{ij}^T \Gamma_{ij} \quad (2.29)$$

and the original model parameters a_{ij1} , a_{ij2} , and b_{ij0} can then be recovered from θ_{ij} for first-order model

$$\begin{bmatrix} a_{ij1} \\ b_{ij0} \end{bmatrix} = \begin{bmatrix} \sqrt{\theta_{ij1}} \\ \pm \sqrt{\theta_{ij2}} \end{bmatrix} \quad (2.30)$$

and for second-order model

$$\begin{bmatrix} a_{ij2} \\ a_{ij1} \\ b_{ij0} \end{bmatrix} = \begin{bmatrix} \sqrt{\theta_{ij1}} \\ \sqrt{\theta_{ij2} + 2\sqrt{\theta_{ij1}}} \\ \pm\sqrt{\theta_{ij3}} \end{bmatrix} \quad (2.31)$$

After model parameters a_{ij1} and b_{ij0} for first-order model are determined, the delay L_{ij} can be estimated again using the phase relations of Equations (2.19) and (2.26) and the least-squares method

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_M \end{bmatrix} L_{ij} = \begin{bmatrix} -\arg[G_{ij}(j\omega_1)] - \tan^{-1}(a_{ij1}\omega_1) \\ -\arg[G_{ij}(j\omega_2)] - \tan^{-1}(a_{ij1}\omega_2) \\ \vdots \\ -\arg[G_{ij}(j\omega_M)] - \tan^{-1}(a_{ij1}\omega_M) \end{bmatrix} \quad (2.32)$$

for the first order model and similarly

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_M \end{bmatrix} L_{ij} = \begin{bmatrix} -\arg[G_{ij}(j\omega_1)] - \tan^{-1}\left(\frac{a_{ij1}\omega_1}{1-a_{ij2}\omega_1^2}\right) \\ -\arg[G_{ij}(j\omega_2)] - \tan^{-1}\left(\frac{a_{ij1}\omega_2}{1-a_{ij2}\omega_2^2}\right) \\ \vdots \\ -\arg[G_{ij}(j\omega_M)] - \tan^{-1}\left(\frac{a_{ij1}\omega_M}{1-a_{ij2}\omega_M^2}\right) \end{bmatrix} \quad (2.33)$$

for the second-order model, respectively.

Similarly, if a model described by second order system with a zero is needed, such as

$$G'_{ij}(s) = \frac{b_{ij1}s + b_{ij0}}{a_{ij2}s^2 + a_{ij1}s + 1} e^{-L_{ij}s}$$

the same procedure can be applied where

$$\Psi_{ij} = \begin{bmatrix} -\omega_1^4 |G_{ij}(j\omega_1)|^2 & -\omega_1^2 |G_{ij}(j\omega_1)|^2 & \omega_1^2 & 1 \\ -\omega_2^4 |G_{ij}(j\omega_2)|^2 & -\omega_2^2 |G_{ij}(j\omega_2)|^2 & \omega_2^2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -\omega_M^4 |G_{ij}(j\omega_M)|^2 & -\omega_M^2 |G_{ij}(j\omega_M)|^2 & \omega_M^2 & 1 \end{bmatrix}, \quad \Gamma_{ij} = \begin{bmatrix} |G_{ij}(j\omega_1)|^2 \\ |G_{ij}(j\omega_2)|^2 \\ \vdots \\ |G_{ij}(j\omega_M)|^2 \end{bmatrix}$$

$$\Theta_{ij} = \begin{bmatrix} \theta_{ij1} \\ \theta_{ij2} \\ \theta_{ij3} \\ \theta_{ij4} \end{bmatrix} = \begin{bmatrix} a_{ij2}^2 \\ a_{ij1}^2 - 2a_{ij2} \\ b_{ij1}^2 \\ b_{ij0}^2 \end{bmatrix}$$

and the parameters can be obtained as

$$\begin{bmatrix} a_{ij2} \\ a_{ij1} \\ b_{ij1} \\ b_{ij0} \end{bmatrix} = \begin{bmatrix} \sqrt{\theta_{ij1}} \\ \sqrt{\theta_{ij2} + 2\sqrt{\theta_{ij1}}} \\ \pm\sqrt{\theta_{ij3}} \\ \pm\sqrt{\theta_{ij4}} \end{bmatrix}$$

It should be noted that for the second-order model, if $G(0) > 0$, then $b_{ij0} = \sqrt{\theta_{ij4}} > 0$. A negative b_{ij1} corresponds to an inverse response, *i.e.* the output step response will first move in a direction opposite to its final value, and a positive b_{ij1} corresponds to a minimum phase process. Thus, by observing the output time response, one is able to determine b_{ij1} by

$$b_{ij1} = \begin{cases} -\sqrt{\theta_{ij3}}, & \text{if an inverse response is detected;} \\ \sqrt{\theta_{ij3}}, & \text{otherwise.} \end{cases}$$

If $G_{ij}(0) < 0$, reverse the signs of b_{ij0} and b_{ij1} accordingly.

The overall closed-loop TITO frequency domain identification method with its implementation issues are summarized as follows:

Algorithm 2.3:

Initialization: Bring the process to a steady state, record the set points, process steady state outputs and controller outputs, $r_1^0, r_2^0, y_1^0, y_2^0, u_1^0$, and u_2^0 , respectively.

Step 1. Make a step change to r_1 from r_1^0 to r_1^1 , with r_2 kept unchanged, record the controller outputs and process outputs signals for the two loops, until the new steady state is reached at $t \triangleq t_1 = T > t_s$, where the moment of set point change is taken as time zero. Calculate $\Delta y_1^1, \Delta y_2^1, \Delta u_1^1$ and Δu_2^1 .

Step 2. Make a step change to r_2 from r_2^0 to r_2^1 , while keeping $r_1 = r_1^1$ as before, record the error signals for the two loops, until the new steady state is reached at $t \triangleq t_2 = 2T > 2t_s$. Again, the moment of set-point change is taken as time zero. Calculate $\Delta y_1^2, \Delta y_2^2, \Delta u_1^2$ and Δu_2^2 .

Step 3. Estimate frequency responses from (2.10) (2.15) over the frequency range $(0, \omega_c)$ for each sub-system of the process by **Algorithm 2.2**.

Step 4. Form Γ_{ij} and Ψ_{ij} by Equations (2.21) for $n=1$ and (2.27) for $n=2$, respectively. Using least squares formula (2.29) to calculate Θ_{ij} $i, j=1,2$.

Step 5. Estimate model parameters a_{y1}, b_{y1} and L_y by Equations (2.30) and (2.32) for $n=1$, or a_{y1}, a_{y2}, b_{y0} , and L_y by Equations (2.31) and (2.33) for $n=2$, $i, j=1,2$, respectively.

Step 6. Stop.

Remark 2.5: In the noise-free testing environment, the least square method yields very good results. However, in the presence of large measurement noise, the results by least squares method may deviate from the true value. A possible improvement is to use the instrumental variable least-squares method [4, 42], the instrumental

matrix Z is selected such that:

- (a) the inverse of $\lim_{n \rightarrow \infty} (1/n) Z^T \Psi$ exist, and
- (b) $\lim_{n \rightarrow \infty} (1/n) Z^T \Delta = R_{z\Delta} = 0$, i.e. Z and Δ are uncorrelated.

with the instrumental matrix, the best estimation $\tilde{\Theta}$ of Θ is obtained from:

$$\tilde{\Theta} = (Z^T \Psi)^{-1} Z^T \Gamma$$

However, there are no general guidelines in selecting an appropriate Z , it should be chosen by a trade-off between results accuracy and computational burden.

Remark 2.6: It is noted that even though the method is derived for TITO systems, the extension to general MIMO system is straightforward. Consider a n -input n -output multivariable process under decentralized control, which can be presented as, $y(s) = G(s)u(s)$, with

$$y(s) = \begin{bmatrix} y_1(s) \\ y_2(s) \\ \vdots \\ y_n(s) \end{bmatrix}, \quad u(s) = \begin{bmatrix} u_1(s) \\ u_2(s) \\ \vdots \\ u_n(s) \end{bmatrix}, \quad G(s) = \begin{bmatrix} G_{11} & G_{12} & \cdots & G_{1n} \\ G_{21} & G_{22} & \cdots & G_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ G_{n1} & G_{n2} & \cdots & G_{nn} \end{bmatrix}$$

For sequential step tests, repeat the steps similarly as Sector 2.2, where the test steps are extended from 2 to n . Starting from initial state, the testing involves n step sequential set-point changes in order to identify the parameters. Consider the i th step, we have

$$\begin{cases} \Delta y_1^i = G_{11} \Delta u_1^i + G_{12} \Delta u_2^i + \cdots + G_{1n} \Delta u_n^i \\ \Delta y_2^i = G_{21} \Delta u_1^i + G_{22} \Delta u_2^i + \cdots + G_{2n} \Delta u_n^i \\ \vdots \\ \Delta y_n^i = G_{n1} \Delta u_1^i + G_{n2} \Delta u_2^i + \cdots + G_{nn} \Delta u_n^i \end{cases}$$

Based on the n test steps, the matrix form can be formulized as Equation (2.8) and $G(s)$ can be solved from Equation (2.9) where

$$Y(s) \triangleq \begin{bmatrix} \Delta y_1^1(s) & \Delta y_1^2(s) & \cdots & \Delta y_1^n(s) \\ \Delta y_2^1(s) & \Delta y_2^2(s) & \cdots & \Delta y_2^n(s) \\ \cdots & \cdots & \ddots & \vdots \\ \Delta y_n^1(s) & \Delta y_n^2(s) & \cdots & \Delta y_n^n(s) \end{bmatrix} G(s) \triangleq \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1n}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2n}(s) \\ \vdots & \vdots & \ddots & \vdots \\ G_{n1}(s) & G_{n2}(s) & \cdots & G_{nn}(s) \end{bmatrix}$$

$$U(s) \triangleq \begin{bmatrix} \Delta u_1^1(s) & \Delta u_1^2(s) & \cdots & \Delta u_1^n(s) \\ \Delta u_2^1(s) & \Delta u_2^2(s) & \cdots & \Delta u_2^n(s) \\ \cdots & \cdots & \ddots & \vdots \\ \Delta u_n^1(s) & \Delta u_n^2(s) & \cdots & \Delta u_n^n(s) \end{bmatrix}$$

2.5 Simulation Examples

In practice, the real measurements of the process output under closed-loop test are inevitably corrupted by measurement noise, which leads to the corruption of the constructed test signals. In order to show the effectiveness of the proposed approach under measurement noise conditions, we define the noise-to-signal ratio as:

$$NSR = \frac{\text{mean}(\text{abs}(\text{noise}))}{\text{mean}(\text{abs}(\text{signal}))}$$

and the relative error in frequency domain up to crossover frequency ω_c defined by

$$E = \max_{\omega \in [0, \omega_c]} \left\{ \left| \frac{\hat{G}(j\omega) - G(j\omega)}{G(j\omega)} \right| \times 100\% \right\}$$

where $G(j\omega)$ and $\hat{G}(j\omega)$ are the actual and estimated process frequency response, respectively, and $\angle G(j\omega_c) = -\pi$.

Example 2.1 Consider the well-known Wood and Berry [46] binary distillation column plant:

$$G(s) = \begin{bmatrix} \frac{12.8e^{-s}}{1+16.7s} & \frac{-18.9e^{-3s}}{1+21s} \\ \frac{6.6e^{-7s}}{1+10.9s} & \frac{-19.4e^{-3s}}{1+14.4s} \end{bmatrix}.$$

It is a typical MIMO process with strong interaction and significant time delays. According to the proposed testing method, the initial PID controller parameter are selected as $K_{P1} = 0.5271$, $K_{I1} = 0.0763$, $K_{D1} = 0.45$, and $K_{P2} = -0.1064$, $K_{I2} = -0.018$, $K_{D2} = 0.02$, the step test signals are shown in Fig. 2.5.

Fig. 2.5 System step tests

The obtained identification models under difference noise level are given in Table 2.1. Under the 10% and 20% NSR, the Nyquist curve of the estimated results based on first-order models and second-order models are shown in Fig. 2.6 and Fig. 2.7, respectively. (Solid line: Actual process, +: Estimated frequency responses at 10% NSR, *: Estimated frequency response at 20% NSR).

Table 2.1 Identification results for Example 2.1

Noise Level	$\hat{G}(s)$		E (%)	
	First order	Second order	First order	Second order
0	$G_{11}(s) = \frac{12.7862e^{-1.0290s}}{16.5258s + 1}$	$G_{11}(s) = \frac{12.8354e^{-0.5650s}}{8.3353s^2 + 17.2432s + 1}$	0.9485	2.0726
	$G_{12}(s) = \frac{-18.8654e^{-3.0082s}}{20.9531s + 1}$	$G_{12}(s) = \frac{-18.9369e^{-2.4883s}}{14.2452s^2 + 21.3728s + 1}$	0.8985	2.3425
	$G_{21}(s) = \frac{6.6000e^{-7.0106s}}{10.8979s + 1}$	$G_{21}(s) = \frac{6.6036e^{-6.3980s}}{8.7006s^2 + 11.5022s + 1}$	0.1302	2.2539
	$G_{22}(s) = \frac{-19.4047e^{-3.0118s}}{14.4843s + 1}$	$G_{22}(s) = \frac{-19.6049e^{-2.5807s}}{5.3445s^2 + 14.9546s + 1}$	0.6187	2.1192
10%	$G_{11}(s) = \frac{12.5782e^{-1.2848s}}{16.4079s + 1}$	$G_{11}(s) = \frac{12.6407e^{-0.6428s}}{6.2878s^2 + 17.2161s + 1}$	4.2731	3.1530
	$G_{12}(s) = \frac{-18.7041e^{-2.8636s}}{20.6871s + 1}$	$G_{12}(s) = \frac{-18.7724e^{-2.4289s}}{14.7401s^2 + 21.9973s + 1}$	2.4054	3.1173
	$G_{21}(s) = \frac{6.4809e^{-7.1670s}}{10.7352s + 1}$	$G_{21}(s) = \frac{6.5877e^{-5.9717s}}{14.2220s^2 + 11.7385s + 1}$	2.6477	3.2859
	$G_{22}(s) = \frac{-19.2314e^{-2.8083s}}{14.2769s + 1}$	$G_{22}(s) = \frac{-19.7227e^{-2.1487s}}{12.0381s^2 + 15.5626s + 1}$	2.5753	3.1802
20%	$G_{11}(s) = \frac{12.3077e^{-1.4046s}}{16.0221s + 1}$	$G_{11}(s) = \frac{12.2679e^{-0.6721s}}{9.1029s^2 + 16.2473s + 1}$	9.4847	10.2519
	$G_{12}(s) = \frac{-18.2195e^{-2.7695s}}{19.9567s + 1}$	$G_{12}(s) = \frac{-18.4418e^{-2.3984s}}{14.7766s^2 + 20.8207s + 1}$	8.1978	7.9267

$G_{21}(s) = \frac{6.2992e^{-7.4528s}}{10.4185s + 1}$	$G_{21}(s) = \frac{6.5394e^{-5.9250s}}{15.2658s^2 + 11.7047s + 1}$	6.6869	8.0164
$G_{22}(s) = \frac{-19.0150e^{-2.5450s}}{13.7227s + 1}$	$G_{22}(s) = \frac{-20.0389e^{-2.0861s}}{13.0672s^2 + 15.7646s + 1}$	7.6368	6.2963

Fig.2.6 Process Nyquist plots and estimated first-order model frequency responses for Example 2.1

Fig.2.7 Process Nyquist plots and estimated second-order model frequency responses for Example 2.1

Example 2.2 This example is adopted from Palmor *et al.* [52] as

$$G(s)=\begin{bmatrix}\frac{0.5}{(0.1s+1)^2(0.2s+1)^2}&\frac{-1}{(0.1s+1)(0.2s+1)^2}\\\frac{1}{(0.1s+1)(0.2s+1)^2}&\frac{2.4}{(0.1s+1)(0.2s+1)^2(0.5s+1)}\end{bmatrix}$$

There exist strong interactions in this fourth-order process. In this example, the initial PID controller parameters are chosen as $K_{p1}=1.4771$, $K_{i1}=3.7177$, $K_{D1}=0.1485$, and $K_{p2}=0.3077$, $K_{i2}=0.4661$, $K_{D2}=0.0514$, the obtained identification models under 20% NSR are as in Table 2.2. And the Nyquist curve of the estimated results are shown in Fig. 2.8 (Solid line: Actual process, * line: Estimated model under 20% NSR).

Table 2.2 Identification results for Example 2.2

Noise Level	$\hat{G}(s)$	$E\%$
20%	$G_{11}(s)=\frac{0.4941e^{-0.1694s}}{0.0358s^2+0.4442s+1}$	13.3947
	$G_{12}(s)=\frac{-0.9412e^{-0.0421s}}{0.0671s^2+0.4214s+1}$	10.6542

Fig. 2.8 Process Nyquist plots and estimated frequency responses for Example 2.2

Example 2.3 The modified plant is adopted from Loh *et al.* [48], which represents a weakly coupled process

$$G(s) = \begin{bmatrix} \frac{12.8e^{-s}}{1+16.7s} & \frac{0.2e^{-3s}}{1+21s} \\ \frac{e^{-7s}}{1+10.9s} & \frac{19.4e^{-3s}}{1+14.4s} \end{bmatrix}$$

In this example, the initial PID controller parameters are chosen as $K_{p1} = 0.076$, $K_{i1} = 0.0065$, $K_{d1} = 0.2246$, and $K_{p2} = 0.0501$, $K_{i2} = 0.0044$, $K_{d2} = 0.1456$, the

obtained identification models under 20% NSR are as in Table 2.3. The Nyquist curve of the estimated results are shown in Fig. 2.9 (Solid line: Actual process, *: Estimated model under 20% NSR). And for this example, the comparison of computing cost is same as Example 2.1.

Table 2.3 Identification results for Example 2.3

Noise Level	$\hat{G}(s)$	$E(\%)$
20%	$G_{11}(s) = \frac{12.5377}{16.0619s + 1} e^{-0.6049s}$	7.5392
	$G_{12}(s) = \frac{0.1928}{19.4336s + 1} e^{-2.3672s}$	9.4618
	$G_{21}(s) = \frac{0.9820}{10.3899s + 1} e^{-6.3351s}$	11.0717
	$G_{22}(s) = \frac{19.0244}{13.7140s + 1} e^{-2.4024s}$	9.1977

Fig. 2.9 Process Nyquist plots and estimated frequency responses for Example 2.3

It is worth noting that if Equation (2.10) is used and 15 frequency points need to be calculated for $G_y(j\omega)$ with minimal critical frequency, the $G_y(j\omega)$ with maximal critical frequency will need to calculate more than 100 points for the predetermined $\Delta\omega$ of Example 2.1. On the other hand, if the initial frequency number used by Algorithm 2.1 and Algorithm 2.2 is also chosen as 15, the computation burden is reduced substantially. The number of *FFT* computation of Example 2.1 for each method is shown in Table 2.4. It is clearly shown that Algorithm 2.2 uses almost half of *FFT* computation required by Algorithm 2.1, and less than 20% of *FFT* computation compared with the direct computation of Equation (2.10).

Table 2.4 Computation cost comparison for Example 2.1

Algorithm	Trial and error	Algorithm 2.1	Algorithm 2.2
Number of computation	≥ 1000	360	176

Chapter 3

Decentralized Control System Design Based on ERGA

3.1 Introduction

Compared with complex centralized control systems, the decentralized control scheme enjoys certain advantages, in which fewer parameters are required, much easier to be understood and implemented, and loop failure tolerance can be assured at the design stage. Therefore, the decentralized control systems are more often used in real MIMO process control applications. To design decentralized control systems, the first step is to determine loop configuration, i.e. pair the manipulated variables and controlled variables to achieve the minimum interactions among loops so that the resulting MIMO control system mostly resembles its SISO counterparts and the

subsequent controller tuning is largely facilitated by SISO design techniques [36, 64].

Since the pioneering work of Bristol [58], the RGA based techniques for control loop configuration have been widely used in industry, applications including blending, energy conservation, and distillation columns, etc [65-67]. The most important advantage of RGA based techniques is simple calculation since only process steady state gains are involved and scaling independent due to its ratio nature [68]. However, using steady state gains alone may result in incorrect interaction measures and consequently improper loop pairing decisions since no dynamic information of the process is taken into consideration.

To overcome the limitations of RGA based loop pairing criterion, several pairing methods have later been proposed by using DRGA measures to consider the effects of process dynamics, which employ the transfer function model instead of the steady state gain matrix to calculate RGA [69-71]. In DRGA, the denominator involved is achieving perfect control at all frequencies, while the numerator is simply the open loop transfer function. Recently, Mc Avoy *et al.* [59] proposed a significant DRGA approach. By using the available dynamic process model, a proportional output optimal controller is designed based on the state space approach, and the resulting controller gain matrix is used to define a DRGA. Several examples in which the normal RGA provided the inaccurate interaction measure and improper pairing selection were studied, while the significant DRGA method provided more accurate interaction assessment and the best pairings in all cases. However, DRGA is often controller dependent, which makes it more difficult to calculate and be understood by practical control engineers.

For decentralized control system design, the existing methods can be roughly divided into three categories: 1) Sequential loop closing methods [72-74]: A controller is designed for a selected input-output pair and this loop is closed. Then a

second controller is designed by taking the closed-loop dynamic into account for the second pair while the first controller has been closed, etc. 2) Independent design methods [75-77]: Find each controller based on the paired transfer function while satisfying some constraints due to process interactions. In this approach, the issue of interaction between loops is examined first, and sufficient conditions are derived for the individual loops that guarantee stability and robust performance. Subsequently, the SISO controllers are designed such that each loop satisfies these sufficient conditions. 3) Detuning methods [78-80]: Each controller is firstly tuned on the basis of diagonal elements of the system transfer function matrix, ignoring the process interactions from other loops. Then, detune the controller gain parameters based on closed-loop stability conditions by taking the loop interactions into consideration. The typical example is using Ziegler-Nichols tuning formula for each diagonal element plus detuning factor evaluated by RGA.

In this chapter, we will focus on the decentralized control system design issues for the local models. The main works are: 1) by combining the advantages of RGA and DRGA, both the steady state gain and the bandwidth of the process transfer function matrix elements are employed to provide a more comprehensive yet simple description for loop interactions; 2) a new loop pairing criterion based on the new interaction measure which results in minimum loop interactions are proposed in terms of ERGA of the process; and 3) based on the proposed ERGA, an effective engineering oriented controller tuning formula is developed by using Ziegler-Nichols tuning rule or other tuning methods for SISO system plus detuning factor evaluated by ERGA.

The loop pairing and control system design technique provides a convenient way for control engineers to tune the MIMO control systems. Since ERGA can provide simple and more accurate information of the interactions, the detuning factor based

controller design method can be considered as a benchmark for the control system. The main advantages of the proposed method are: 1) compared with DRGA method, it can also provide a comprehensive description of dynamic interaction among individual loops without the requirement of specified controller type; 2) the effectiveness of the new loop pairing criterion is comparable to DRGA but requires much less computation; 3) it results less conservative controller design when the detuning factor method is used; and 4) it is very simple for field engineers to understand and work out loop pairing together with controller design in applications.

3.2 Preliminaries

Consider a MIMO system with n -inputs and n -outputs as shown in Fig. 3.1, where $r_i, i = 1, 2, \dots, n$ are the reference inputs; $u_i, i = 1, 2, \dots, n$ are the manipulated variables; $y_i, i = 1, 2, \dots, n$ are the system outputs, $G(s)$ and $G_c(s)$ are process matrix and decentralized controller matrix, respectively, $G(s)$ and $G_c(s)$ are with compatible dimensions and expressed by

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) & \dots & g_{1n}(s) \\ g_{21}(s) & g_{22}(s) & \dots & g_{2n}(s) \\ \dots & \dots & \dots & \dots \\ g_{n1}(s) & g_{n2}(s) & \dots & g_{nn}(s) \end{bmatrix}$$

and

$$G_c(s) = \begin{bmatrix} g_{c1}(s) & 0 & \dots & 0 \\ 0 & g_{c2}(s) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & g_{cn}(s) \end{bmatrix}$$

Fig. 3.1 Closed-loop MIMO control system

Assume that the system is open loop stable and the process matrix $G(s)$ is nonsingular at steady state. The loop pairing problem defines the control system structure, i.e. the available plant inputs are to be used to control each of the plant outputs. The most popular loop pairing method is the RGA and NI based pairing rules as follows [58, 68, 81].

Define relative gain from the process steady state gains

$$\lambda_{ij} = \frac{(\partial y_i / \partial u_j)_{\text{all loops open}}}{(\partial y_i / \partial u_j)_{\text{all other loops close except for loop } y_i - u_j}} \quad (3.1)$$

and RGA as

$$\Phi_{RGA} = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1n} \\ \lambda_{21} & \lambda_{22} & \dots & \lambda_{2n} \\ \dots & \dots & \dots & \dots \\ \lambda_{n1} & \lambda_{n2} & \dots & \lambda_{nn} \end{bmatrix} \quad (3.2)$$

which can be calculated by [68]

$$\Phi_{RGA} = G \otimes G^{-T} \quad (3.3)$$

where the operator $\otimes$ is the hadamard product and G^{-T} representing the transpose of inverse of G . The RGA based pairing criterion is:

- The paired RGA elements are closest to 1.0;
- All the paired RGA elements are positive;
- Large RGA elements should be avoided.

Furthermore, if all n loops are closed, the multi-loop system will be unstable for all possible values of controller parameters, i.e. it will be “structurally monotonic unstable” if the NI

$$NI = \frac{|G(0)|}{\prod_{i=1}^n g_{ii}(0)} < 0 \quad (3.4)$$

where $|G(0)|$ denotes the determinant of matrix $G(0)$. A positive sign of NI, i.e. $NI > 0$, provides a necessary stability condition and consequently, constitutes a complementary tool to the RGA in variable pairing selection.

One of the main advantages of the RGA based loop pairing method is that the interaction measure depends on only the steady state gains. This information is easily obtained from simple identification experiments or steady state design models. A potential weakness of these methods, however, is the same fact that they only use the steady state gains, which is based on the assumption of perfect loop control, to determine loop pairing. We use the following example to illustrate this point.

Example 3.1 Consider a slightly revised process given by [3],

$$G(s) = \begin{bmatrix} \frac{2e^{-s}}{10s+1} & \frac{2e^{-s}}{s+1} \\ \frac{e^{-s}}{s+1} & \frac{-4e^{-s}}{10s+1} \end{bmatrix}$$

The RGA is

$$\Phi_{RGA}(G(0)) = \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}$$

This resulted RGA implies the diagonal pairing 1-12-2 is a good choice. Hence, according to the RGA based loop pairing criterion, the pairing of 1-12-2 should be preferred for the smaller interaction to any one loop from other closed loop. However, it has been noted that the diagonal pairing configuration is not advisable, and the RGA detuning method can not be used which will result unstable performance. On the other hand, the off-diagonal pairing 1-22-1 takes advantage of the fast $g_{21}(s)$ and $g_{12}(s)$ transfer functions can achieve better outputs response. The main reason for the better performance of the off-diagonal pairing is the dynamic properties of the transfer functions. It can be easily seen that the effective time constants (the sum of time delay and time constant) of the off-diagonal elements are almost 5 times smaller than diagonal ones. In such a case, although the steady state gains of diagonal elements are obviously larger than the off-diagonal elements, pairing the faster loops still take advantage of the time scale decoupling such that seriousness of the interactions from the slower loop would be reduced.

3.3 ERGA-based Loop Pairing

In designing decentralized controllers for MIMO processes, it is desired that the interaction measures and loop pairing will address the following issues:

- 1) The interaction measure should consider the finite bandwidth control, since the assumption of perfect control is only valid for very low frequency range.
- 2) The loop pairing decision should be controller independent such that any controller type could be designed after loop pairing.
- 3) The pairing should result in minimal interaction within the interested frequency range not only statically but also dynamically.

- 4) It should be simple and easy to use for practicing engineers.

In decentralized control system design, since the individual loop is tuned around the critical frequency region of the transfer function which is the region around the control system bandwidth, this is the frequency region that should be focused upon when considering the effect of interactions. Therefore, two factors in the open loop transfer functions will affect the loop pairing decision:

- 1) Steady state gain: the steady state gain $g_{ij}(0)$ of the transfer function reflects the effect of the manipulated variable u_j to the controlled variable y_i at steady state.
- 2) Response speed: response speed is accountable for the sensitivity of the controlled variable y_i to manipulated variable u_j and, consequently, the ability to reject the interactions from other loops.

As the response speed is proportional to the bandwidth in frequency domain, we can use the bandwidth to reflect both interactions from finite bandwidth control and pairing loops to results in a fast response. Let

$$g_{ij}(j\omega) = g_{ij}(0)g_{ij}^0(j\omega) \quad (3.5)$$

where $g_{ij}(0)$ and $g_{ij}^0(j\omega)$ are steady state gain and normalized transfer function of $g_{ij}(j\omega)$, i.e. $g_{ij}^0(0) = 1$, respectively.

In order to use both steady state gain and response speed information for interaction measure and loop pairing, we now define the effective gain e_{ij} for a particular transfer function as

$$e_{ij} = g_{ij}(0) \int_0^{\omega_{B,ij}} |g_{ij}^0(j\omega)| d\omega \quad (3.6)$$

where $\omega_{B,ij}$ for $i, j = 1, 2, \dots, n$ are the bandwidths of transfer function $g_{ij}(j\omega)$ and

$|\bullet|$ is the absolute value of $\bullet$.

Fig. 3.2 Response curve and effective energy of $g_{ij}(j\omega)$

For the frequency response of $g_{ij}(j\omega)$ as shown in Fig. 3.2, e_{ij} is the area covered by $g_{ij}(j\omega)$ up to $\omega_{B,ij}$, where $\omega_{B,ij}$ is determined by the frequency where the magnitude of frequency response reduced to $0.707 g_{ij}(0)$, i.e.

$$g_{ij}(j\omega_{B,ij}) = 0.707 g_{ij}(0) \quad (3.7)$$

Then, the effective gain matrix can be expressed as

$$E = \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1n} \\ e_{21} & e_{22} & \dots & e_{2n} \\ \dots & \dots & \dots & \dots \\ e_{n1} & e_{n2} & \dots & e_{nn} \end{bmatrix}$$

To simplify the calculation, we approximate the integration of e_{ij} by a rectangle area, i.e.

$$e_{ij} \approx g_{ij}(0) \omega_{B,ij} \quad i, j = 1, 2, \dots, n \quad (3.8)$$

Then, the effective gain matrix is given as

$$E = G(0) \otimes \Omega \quad (3.9)$$

where

$$G(0) = \begin{bmatrix} g_{11}(0) & g_{12}(0) & \dots & g_{1n}(0) \\ g_{21}(0) & g_{22}(0) & \dots & g_{2n}(0) \\ \dots & \dots & \dots & \dots \\ g_{n1}(0) & g_{n2}(0) & \dots & g_{nn}(0) \end{bmatrix}$$

and

$$\Omega = \begin{bmatrix} \omega_{B,11} & \omega_{B,12} & \dots & \omega_{B,1n} \\ \omega_{B,21} & \omega_{B,22} & \dots & \omega_{B,2n} \\ \dots & \dots & \dots & \dots \\ \omega_{B,n1} & \omega_{B,n2} & \dots & \omega_{B,nn} \end{bmatrix}$$

are the steady state gain matrix and the bandwidth matrix, respectively. Since the e_{ij} is an indication of interaction energy to other loops when loop $y_i - u_j$ is closed, the bigger the e_{ij} value is, the more dominant of the loop will be.

Replacing the steady state gain matrix $G(0)$ of Equation (3.3) by the effective gain matrix E of Equation (3.9), one can use RGA and NI based rules for loop pairing. The ERGA pairing procedure based on effective gain is given as follows.

Define ϕ_{ij} the effective relative gain between output variable y_i and input variable u_j , as the ratio of two effective gains

$$\phi_{ij} = \frac{e_{ij}}{\hat{e}_{ij}} \quad (3.10)$$

where $\hat{e}_{ij}$ is the effective gain between output variable y_i and input variable u_j when all other loops are closed. When the effective relative gains are calculated for all the input/output combinations of a multivariable process, it results in an array of the form

$$\Phi_{ERGA} = \begin{bmatrix} \phi_{11} & \phi_{12} & \dots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \dots & \phi_{2n} \\ \dots & \dots & \dots & \dots \\ \phi_{n1} & \phi_{n2} & \dots & \phi_{nn} \end{bmatrix} \quad (3.11)$$

which can be calculated by

$$\Phi_{ERGA} = E \otimes E^{-T} \quad (3.12)$$

Since both RGA and ERGA using relative gains, the properties of RGA can be direct extended to ERGA:

- 1) The value of ϕ_{ij} is a measure of the effective interaction expected in the i^{th} loop if its output y_i is paired with u_j .
- 2) The elements of the ERGA across any row, or down any column, sum up to 1, that is

$$\sum_{j=1}^n \phi_{ij} = \sum_{j=1}^n \phi_{ij} = 1.$$

- 3) Let $\hat{e}_{ij}$ represent the loop i effective gain when all the other loops are closed, whereas ϕ_{ij} represents the normal, open-loop effective gain, then

$$\hat{e}_{ij} = \frac{1}{\phi_{ij}} e_{ij}$$

- 4) When ϕ_{ij} is negative, with other loops open, the change in y_i in response to a change in u_j is opposite in direction to that when the other loops are closed.

Similar to RGA and NI based pairing rules, the ERGA and NI based loop pairing rules requires that manipulated and controlled variables in a decentralized control system be paired:

- 1) Corresponding ERGA elements are closest to 1.0.
- 2) The NI is positive.

- 3) All paired ERGA elements are positive.
- 4) Large ERGA elements should be avoided.

Here, both ERGA and NI play important roles for control structure selection. ERGA is used to measure interactions, while NI is used as a sufficient condition to rule out the closed-loop unstable pairings. Comparing ERGA method with RGA and DRGA, we may expect that:

- 1) In addition to steady state gains, only bandwidth need to be calculated in ERGA method, it is far easier to calculate than those DRGA methods.
- 2) ERGA combines both steady state gains and bandwidth in measuring the loop interactions, it should provide better pairing results than that of RGA based pairing and comparable with DRGA ones.
- 3) Since ERGA uses only information of open-loop process transfer functions, it is controller type independent.

3.4 ERGA-based Controller Tuning Method

Based on the obtained ERGA, the PID detuning method can be easily employed for MIMO decentralized control system as follows:

- 1) Pairing the MIMO loops according to the ERGA such that the diagonal transfer functions are utilized for the decentralized controller design.
- 2) Using single loop tuning rules to obtain the initial controller gains K_{ci}^* for the individual controllers, for example the well-known Ziegler-Nichols method, or the Cohen-Coon method, etc.
- 3) Determining the decentralized controller gain K_{uif} from the obtained single

loop controller gain K_u by multiply an appropriate detuning factor.

The detuning factor method is very popular for estimating the tuning of decentralized control systems without the time consuming iterations associated with trial and error or the large computation associated with the optimization approach. The goal of the method is to provide initial tuning constants that are much closer to the best decentralized control values than single loop tuning constants. The general approach is to establish how much the controller tuning must be changed from single loop values when applied in a decentralized control system.

For a TITO process, the closed-loop transfer functions under decentralized control can be determined as in [82]

$$\begin{aligned}\frac{Y_1(s)}{Y_{1d}(s)} &= \frac{G_{c1}(s)G_{11}(s) + G_{c1}(s)G_{c2}(s)[G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)]}{G_{CL}(s)} \\ \frac{Y_2(s)}{Y_{2d}(s)} &= \frac{G_{c2}(s)G_{22}(s) + G_{c1}(s)G_{c2}(s)[G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)]}{G_{CL}(s)} \\ \frac{Y_1(s)}{Y_{2d}(s)} &= \frac{G_{c2}(s)G_{12}(s)}{G_{CL}(s)} \\ \frac{Y_2(s)}{Y_{1d}(s)} &= \frac{G_{c1}(s)G_{21}(s)}{G_{CL}(s)}\end{aligned}\quad (3.13)$$

where the characteristic equation of the overall system is given as

$$G_{CL}(s) = 1 + G_{c1}(s)G_{11}(s) + G_{c2}(s)G_{22}(s) + G_{c1}(s)G_{c2}(s)[G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)] \quad (3.14)$$

Notice that the DRGA for TITO process is

$$\lambda(s) = \frac{1}{1 - \frac{G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)}}$$

Equation (3.14) can be written as

$$\begin{aligned}
G_{CL}(s) &= 1 + G_{c1}(s)G_{11}(s) + G_{c2}(s)G_{22}(s) + G_{c1}(s)G_{c2}(s)G_{11}(s)G_{22}(s) \left(1 - \frac{G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right) \\
&= 1 + G_{c1}(s)G_{11}(s) + G_{c2}(s)G_{22}(s) + \frac{G_{c1}(s)G_{c2}(s)G_{11}(s)G_{22}(s)}{\lambda(s)}
\end{aligned} \tag{3.15}$$

Approximate DRGA by ERGA and divide Equation (3.15) by $1 + G_{c2}(s)G_{22}(s)$, the characteristic equation becomes

$$\begin{aligned}
G_{CL}(s) &= \frac{1 + G_{c2}(s)G_{22}(s) + G_{c1}(s)G_{11}(s) + \frac{G_{c1}(s)G_{c2}(s)G_{11}(s)G_{22}(s)}{\phi}}{1 + G_{c2}(s)G_{22}(s)} \\
&= 1 + G_{c1}(s)G_{11}(s) \left(\frac{1 + \frac{G_{c2}(s)G_{22}(s)}{\phi}}{1 + G_{c2}(s)G_{22}(s)} \right)
\end{aligned} \tag{3.16}$$

where ϕ is the ERGA index. The closed-loop characteristic expression given by Equation (3.16) determines the stability of the control system. To evaluate potential simplifications, the relative importance of each term must be determined at the critical frequency of the loop. The method considers three cases for tuning loop 1, but the same results can be obtained for loop 2 by transposing the subscripts.

Case 1: Loop 1 is much faster than Loop 2

When the loop 1 process is much faster than loop 2, the term $G_{c2}(j\omega)G_{22}(j\omega)$ is very small at the loop 1 critical frequency because of the tendency of processes to have amplitude ratios that decrease rapidly after the corner frequency. Assuming that ϕ is not a strong function of frequency, as is most often true,

$$\left| \frac{1 + G_{c2}(j\omega)G_{22}(j\omega)/\phi}{1 + G_{c2}(j\omega)G_{22}(j\omega)} \right| = 1.0$$

which gives

$$G_{CL}(j\omega) \approx 1 + G_{c1}(j\omega)G_{11}(j\omega) \quad (3.17)$$

Therefore, the very fast loop 1 in this case can be tuned like a single loop controller without interaction regardless of the ϕ value. This result confirms a qualitative argument in which we would consider the interaction from the slow loop to be a slow disturbance to the very fast loop 1, which could be tuned using single loop methods.

Case 2: Loop 1 much slower than Loop 2

When loop 1 is much slower, the term for the fast controller, $G_{c2}(j\omega)$, would have a very large magnitude at the critical frequency of loop 1, because the amplitude ratio of the integral mode in $G_{c2}(j\omega)$ will have a very high value at a frequency much less than the loop 2 critical frequency. Therefore, $G_{c2}(j\omega)G_{22}(j\omega) \gg 1.0$, which leads to the following simplification in the characteristic equation.

$$\begin{aligned} G_{CL}(j\omega) &\approx 1 + G_{c1}(j\omega)G_{11}(j\omega) \frac{G_{c2}(j\omega)G_{22}(j\omega)}{\phi \cdot G_{c2}(j\omega)G_{22}(j\omega)} \\ &\approx 1 + G_{c1}(j\omega) \frac{G_{11}(j\omega)}{\phi} \end{aligned} \quad (3.18)$$

Since effective relative gain has been used as an approximation for the frequency dependent relative gain, the gain of the process "seen" by the controller 1 in the decentralized control system is changed by $1/\phi$, from the single loop gain (K_{11}). Therefore, the controller gain has to be adjusted by approximately the inverse of ϕ , for $\phi < 1$, to maintain the proper stability margin. For $\phi > 1$, however, the controller gain should be kept the same to guarantee the control system integrity. Since the phase lag is not affected, the integral time is unchanged.

Case 3: Loops 1 and 2 have the similar dynamics

For the loops have similar dynamics, define

$$\Lambda_1 = g_{11}g_{c1}, \quad \Lambda_2 = g_{22}g_{c2},$$

To simplify the stability analysis as [82], let $\Lambda_1 = \Lambda_2 = \Lambda$, the system stability can be determined by the root of following equation

$$1 + 2\Lambda(s) + \frac{\Lambda^2(s)}{\phi} = 0 \quad (3.19)$$

in order to determine the closed-loop stability, rewrite Equation (3.19) as

$$1 + 2\Lambda(s) + \frac{\Lambda^2(s)}{\phi} = \left(1 + \frac{\Lambda(s)}{\phi + \sqrt{\phi^2 - \phi}}\right) \left(1 + \frac{\Lambda(s)}{\phi - \sqrt{\phi^2 - \phi}}\right)$$

and consider the following two scenarios:

a) $\phi > 1.0$: Since ϕ is a constant and greater than 1, ultimate frequency is unaffected by the interaction, that is

$$\omega_{u1I} = \omega_{u1}$$

as the factor with $-\sqrt{\phi^2 - \phi}$ will always determine the stability limit of the interacting system and the ultimate gain k_{u1I} can be calculated from

$$k_{u1I} = \frac{\phi - \sqrt{\phi^2 - \phi}}{|g_{11}|} \Big|_{\omega_{u1I}} \quad (3.20)$$

By the definition of gain margin

$$k_{u1} = \frac{1}{|g_{11}|_{\omega_{u1}}} \quad (3.21)$$

Substituting (3.21) into (3.20), we obtain

$$k_{u1I} = \left(\phi - \sqrt{\phi^2 - \phi}\right) k_{u1} \quad (3.22)$$

b) $\phi < 1.0$: The factor with $\phi + \sqrt{\phi^2 - \phi}$ will determine the stability limit of the interacting system and the gain margin K_{u1I} can be calculated from

$$K_{u1I} = \frac{\phi + \sqrt{\phi^2 - \phi}}{|g_{11}|} |a_{u1I}| \quad (3.23)$$

Since $\phi < 1.0$, the term $\phi + \sqrt{\phi^2 - \phi}$ is a complex constant. This term affects both phase angle and gain. There is no simple method to relate K_{u1I} with K_{u1} . Using the methods proposed [83], approximated formulation for first-order plus dead time process can be expressed by

$$\omega_{u1I} = \left(1 - \frac{1.5 \tan^{-1} \left(\sqrt{\frac{1}{\phi} - 1} \right)}{\pi} \right) \omega_{u1} \quad (3.24)$$

$$k_{u1I} = \sqrt{\phi} \left(1 - \frac{\tan^{-1} \left(\sqrt{\frac{1}{\phi} - 1} \right)}{\pi} \right) k_{u1} \quad (3.25)$$

thus for $\phi < 1.0$, not only the controller gain is reduced, but the reset time is increased as well due to the interaction.

To simplify the calculation, (3.24) and (3.25) in the region of $0.5 < \phi < 1.0$ can be approximated, using least square method, by the following functions:

$$\begin{cases} \omega_{u1I} = (0.85\phi + 0.1)\omega_{u1} \\ k_{u1I} = (0.62\phi + 0.3)k_{u1} \end{cases} \quad (3.26)$$

The correlation of $\omega_{u1I} - \omega_u$ and $k_{u1I} - k_u$ is shown in Fig. 3.3, where solid lines denote the original relation from Equations (3.24) and (3.25), dotted lines represent that from Equation (3.26)

Fig. 3.3 Detuning factor comparison for $\phi < 1.0$

Fig. 3.3 indicates that the linear relation directed from ERGA can be applied to detune the controller parameters and the approximate Equation (3.26) provides more convenient settings compare with Equations (3.24) and (3.25).

The detuning rules for all three cases are summarized in Table 3.1, which provide a guideline for rough tuning of TITO decentralized controller parameters. The method provided in this chapter takes advantage of simplifications to determine the tuning and can be easily accepted by field control engineers. However, fine-tuning based on plant experience or advanced control design methods is still required, we will discuss this issue in Chapter 4.

Table 3.1 Effects of interaction on Loop 1

Speed of $G_{c2}(j\omega)G_{22}(j\omega)$ relative to $G_{c1}(j\omega)G_{11}(j\omega)$	ϕ	Relation		Compare with single loop response
		K_{u1I} and K_{u1}	ω_{u1I} and ω_{u1}	
Much faster	$\triangleright$	$K_{u1I} = K_{u1}$	$\omega_{u1I} = \omega_{u1}$	More sluggish
Much slower	$\triangleright$	$K_{u1I} = K_{u1}$	$\omega_{u1I} = \omega_{u1}$	More sluggish

Almost equal	$\triangleright$	$K_{u1f} = (\phi - \sqrt{\phi^2 - \phi}) K_{u1}$	$\omega_{u1f} = \omega_{u1}$	More sluggish
Much faster	$\ll$	$K_{u1f} = \phi K_{u1}$	$\omega_{u1f} = \omega_{u1}$	More Oscillatory
Much slower	$\ll$	$K_{u1f} = K_{u1}$	$\omega_{u1f} = \omega_{u1}$	More Oscillatory
Almost equal	$\ll$	$K_{u1f} = (0.62\phi + 0.3) K_{u1}$	$\omega_{u1f} < \omega_{u1}$	More Oscillatory

3.5 Simulation Examples

In this section, various examples are used to show the effectiveness of ERGA method. The tuning is facilitated by Z-N tuning rule based on the critical gain K_{u1} and critical period T_{u1} as below

$$\text{PI} - K_P = 0.4K_{u1}, T_I = 0.8T_{u1}$$

$$\text{PID} - K_C = 0.4K_{u1}, T_I = 0.5T_{u1}, T_D = 0.125T_{u1}$$

Example 3.2 Continue with Example 3.1, by $\Phi_{RGA} = 0.8$, implies diagonal loop pairing. On the other hand, the ERGA is easily obtain as

$$\Phi_{ERGA} = \begin{bmatrix} 0.14 & 0.86 \\ 0.86 & 0.14 \end{bmatrix},$$

which implies the off-diagonal pairing.

For the off-diagonal pairing, controllers are designed for the two SISO systems. The ultimate gain K_u and ultimate frequency ω_u of each individual off-diagonal transfer function are obtained as, $K_{u1} = 1.1$, $\omega_{u1} = 2$, $K_{u2} = 2.2$, and $\omega_{u2} = 2$. Using both PI and PID controllers with parameters from Z-N tuning leads to the following parameters

$$\begin{aligned} \text{Loop1:} \quad & \text{PI} - K_C = 0.45, K_I = 0.18 \\ & \text{PID} - K_C = 0.45, K_I = 0.29, K_D = 0.17 \end{aligned}$$

$$\begin{aligned} \text{Loop2:} \quad & \text{PI} - K_C = 0.9, K_I = 0.36 \\ & \text{PID} - K_C = 0.9, K_I = 0.58, K_D = 0.35 \end{aligned}$$

As $\phi < 1.0$, a set of PI controller parameters can be detuned according to the Equation (3.26). The controllers are adjusted by the detuning factor as follows

$$\begin{aligned} \text{Loop1:} \quad & \text{PI} - K_C = 0.37, K_I = 0.13 \\ & \text{PID} - K_C = 0.37, K_I = 0.2, K_D = 0.14 \end{aligned}$$

$$\begin{aligned} \text{Loop2:} \quad & \text{PI} - K_C = 0.75, K_I = 0.25 \\ & \text{PID} - K_C = 0.75, K_I = 0.4, K_D = 0.28 \end{aligned}$$

Fig. 3.4 shows the closed-loop response of the designed off-diagonal PI and PID controllers with ERGA detuning method, where the set-points change occur at $t = 0$ ($y_1 = 1, y_2 = 0$) and at $t = 200$ ($y_2 = 1$), and unit load disturbance occurs at $t = 400$.

Fig. 3.4 PI and PID closed-loop responses for Example 3.2

It is clear that both PI and PID control using proposed tuning method can achieve a good performance. However under the same environment, the detuning method

according to RGA index which indicating diagonal pairing using Z-N SISO tuning method [84] results in an unstable system.

Example 3.3 Consider an industrial-scale polymerization reactor [85],

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{22.89}{4.572s+1}e^{-0.2s} & \frac{-11.64}{1.807s+1}e^{-0.4s} \\ \frac{4.689}{2.174s+1}e^{-0.2s} & \frac{5.80}{1.801s+1}e^{-0.4s} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{-4.234}{3.445s+1}e^{-0.4s} \\ \frac{-0.601}{1.982s+1}e^{-0.4s} \end{bmatrix} d(s)$$

where y_1 and y_2 are two measurements representing the reactor condition; u_1 and u_2 are the set-points of two reactor feed flow loops; and load disturbance d is the purge flow rate of the reactor.

It is easy to obtain the RGA and ERGA as follows

$$\Phi_{RGA} = \begin{bmatrix} 0.71 & 0.29 \\ 0.29 & 0.71 \end{bmatrix} \text{ and } \Phi_{ERGA} = \begin{bmatrix} 0.53 & 0.47 \\ 0.47 & 0.53 \end{bmatrix}$$

This result implies the diagonal pairing 1-12-2 is the right pairing choice. Therefore, individual controllers are designed for the two SISO loops. The ultimate gain K_u and ultimate frequency ω_u of each individual diagonal transfer function are obtained as, $K_{u1} = 1.6$, $\omega_{u1} = 8$, $K_{u2} = 1.3$, and $\omega_{u2} = 4.2$. Using both PI and PID controllers with parameters from Z-N tuning leads to the following parameters

Loop1:	PI - $K_C = 0.64, K_I = 1$
	PID - $K_C = 0.64, K_I = 1.62, K_D = 0.06$
Loop2:	PI - $K_C = 0.53, K_I = 0.45$
	PID - $K_C = 0.53, K_I = 0.72, K_D = 0.1$

We should note that, in this example, although RGA and ERGA suggest the same loop pairing, Φ_{ERGA} has a larger difference than Φ_{RGA} , which indicates more interaction between the loops. Again, as $\phi < 1.0$, the controller parameters can be

detuned according to Equation (3.26). The controllers are adjusted by the detuning factor as follows

$$\begin{array}{ll}
 \text{Loop1:} & \text{PI} - K_C = 0.36, K_I = 0.34 \\
 & \text{PID} - K_C = 0.36, K_I = 0.55, K_D = 0.03 \\
 \\
 \text{Loop2:} & \text{PI} - K_C = 0.29, K_I = 0.15 \\
 & \text{PID} - K_C = 0.29, K_I = 0.24, K_D = 0.06
 \end{array}$$

Fig. 3.5 shows the closed-loop response of the PI and PID controllers designed with ERGA detuning method, where the unit set-points change of y_1 at $t=0$ and y_2 at $t=30$, and unit load disturbance occurs at $t=60$. It is clear both PI and PID using the proposed tuning method can achieve a good performance. It is also noted that although RGA index has the same indication as ERGA, the RGA detuning method based on Z-N SISO tuning rule will result in an unstable response.

Fig. 3.5 PI and PID closed-loop responses for Example 3.3

Chapter 4

Fine-tuning Decentralized PID Controllers by Receding Horizon Optimization

4.1 Introduction

Despite the rapid development of many advanced system control technologies, PID control is still the most commonly used technique in the industrial process control. The main reason for the popularity is its simplicity and reliability, which can provide most process operators a straightforward manner to operate with reliable control performance. For MIMO processes, decentralized PID control is the dominant controller design technique compared with the centralized MIMO control

strategies such as MPC control, optimal control, etc.

The design and tuning of single loop PID controller have been extensively researched. However most of these techniques cannot be directly applied to design decentralized control systems due to the interactions between loops. Detuning method based on single loop design is undoubtedly the most straightforward and the simplest way to compensate loop interactions. Other than the method discussed in Chapter 3, several other detuning methods have also been reported in the literature. Among them, the most widely used detuning technique is perhaps the Biggest Log Modulus (BLT) tuning method proposed by Luyben [78]. In the BLT method, individual PI controllers are designed using the well known Z-N rule for the respective loops by ignoring interaction effects. The calculated controller gains are then scaled back by using the detuning factor, the loop gains are so adjusted that the biggest log modulus has a specified value. This method provides reasonable preliminary settings with one goal in mind which is to guarantee the close loop stability. However, despite the simple computations, the design technique views interactions as elements impeding system stability and attempts to suppress their effects rather than control them to speed up individual loop. Hence, Luyben's BLT method is rather conservative.

Other design methods have also been developed for decentralized PID control system design based on Nyquist stability analysis and frequency response information. Diagonal dominance and Gershgorin bands, which is considered MIMO version of the Nyquist curve, are often employed to tune the decentralized PID controllers. Lee *et al.* [73] extended the iterative continuous cycling method for SISO system to decentralized PI controller tuning. By refining the Nyquist array method, less conservative stability conditions were provided and controllers were designed based on these conditions. Loh *et al.* [47] and Shen *et al.* [86] proposed

sequential relay-feedback tests to locate the ultimate points of MIMO system and to design decentralized PID controllers. Similar idea was also proposed by Palmor *et al.* [52] by using simultaneous relays in all of the control loops. Two points of Gershgorin bands was used by Ho [87] to design decentralized PID control systems so that the gain and phase margin specifications can be satisfied at the gain and phase crossover frequencies of the diagonal elements. A similar but more accurate method was proposed by Chen *et al.* [88], where the ultimate point information defined for MIMO system is obtained based on Gershgorin bands and PI or PID controllers can be designed. More recently, a stability region concept is suggested by Chen *et al.* [80] to tune decentralized PI controller. Although the closed-loop stability is guaranteed, the main drawback of this method is that it is difficult to determine the critical points by Gershgorin band and a pre-compensator is often required for the systems that are not column diagonally dominant.

It has been well recognized that classical decentralized PID control design can not result in an optimal control system performances due to the structure limitations. However, the extensive use of PID controllers in industrial applications has made many researches attempt to use advanced control concepts such as optimal control, H_∞ , IMC, MPC, etc to obtain the near optimal gains for PID controllers [89-95]. Among these techniques, the centralized MPC has evolved, during the last decade, to become a powerful practical technique for MIMO control systems [96, 97]. The MPC scheme use a dynamic optimization approach to control problems which yield control systems capable of operating without expert intervention for extended periods of time. The key feature of the MPC algorithms is the receding horizon concept, also known as the open-loop optimal feedback. An identified process model predicts the future response and the control action is determined so as to obtain the desired performance over a finite time horizon. The major benefit of MPC is that the control problem is solved through online optimization of the

incorporated upcoming information of error and manipulated variables. Thus, it will be of great advantage if the PID controller parameters are adjusted according to the basic MPC ideas, as it will take advantage of the future information to regulate the controller parameters. A PID tuning method incorporated MPC based on a parallel bank of conventional PID controllers was proposed by Moradi *et al.* [93, 94], where each controller composed of the three gain terms and the current controller operates on the future error signals. The optimal PID gains can be obtained such that the difference between a control signals generated by a MPC algorithm and the predictive PID controller is minimized. However, this predictive PID tuning method can only be applied to centralized control structures, it can not be extended to the decentralized structure which is more commonly used in real world applications.

In Chapter 3, a decentralized PID controller was proposed based on ERGA index, which can be considered as a “coarse tuning” of the decentralized PID control parameters of the TITO systems. In order to further improve the decentralized control system performance, the multi-step prediction concept is employed to regulate the controller parameters, and a two-level PID controller tuning technique based on receding horizon optimization is presented in this chapter. Compared with the traditional decentralized control system design, the proposed tuning method makes use of the predicted future references, the present controller and system outputs to adjust the controller parameters to minimize the MPC-like cost function. The main advantage of the proposed approach is that it has relatively simple structure, which can be easily understood and implemented in practice. Since the method does not require changes in hardware and the rules of each controller parameter, it can be well accepted by practical engineers. Therefore, it can be applied to wide industry areas. Several simulation examples are provided to show that the proposed method performs better than those decentralized control systems based on classical design methods.

4.2 Model Transformation

Unlike the decentralized PID design methods by using frequency information, the time domain model is preferred to predict the process future information. For the first-order plus dead time model described by

$$G(s) = \frac{K}{Ts + 1} e^{-Ls} \quad (4.1)$$

where K is the system static gain, T is the system time constant and L is the pure dead time, the corresponding discrete transfer function model can be obtained as

$$G(z^{-1}) = \frac{bz^{-1}}{1 - az^{-1}} z^{-d} \quad (4.2)$$

where $a = e^{-T_s/T}$, $b = K(1 - a)$, T_s is the sampling interval, and d is normally chosen as $d = \text{round}(L/T_s)$. For SISO systems, such approximation of pure dead time is

acceptable through the properly selected sampling interval. However, since we are dealing with MIMO systems, the approximation may possibly lead to an unacceptable deviation for some elements of the transfer function matrix. To alleviate the problem, let

$$L = L_0 + \varepsilon T_s = dT_s + \varepsilon T_s \quad (4.3)$$

that is, divide the pure dead time into two parts: L_0 represents d times of sampling interval; and εL_s , where $0 \leq \varepsilon < 1$ is the remainder. Thus, Equation (4.1) can be rewritten as

$$G(s) = \frac{K}{Ts + 1} e^{-L_0 s} e^{-\varepsilon T_s s} \quad (4.4)$$

approximate $e^{\varepsilon T_s s}$ by Pade approximation

$$e^{-\varepsilon T_s} \approx \frac{1 - \varepsilon T_s/2}{1 + \varepsilon T_s/2}$$

such that Equation (4.4) becomes

$$G(s) = \frac{K(1 - \varepsilon T_s/2)}{(Ts+1)(1 + \varepsilon T_s/2)} e^{-L_0 s} \quad (4.5)$$

Define $p = e^{-T_s/T}$, $q = e^{-\frac{2}{\varepsilon}}$, $r = \frac{p(p^{-\varepsilon} - 1)}{1 - p}$ and notice that

$$\frac{1 + \frac{\varepsilon T_s}{2T}}{1 - \frac{\varepsilon T_s}{2T}} \approx e^{\frac{\varepsilon T_s}{T}} = p^{-\varepsilon}.$$

The discrete transfer function can be obtained as

$$G(z^{-1}) = \frac{b_0 z^{-1} + b_1 z^{-2}}{1 - a z^{-1}} z^{-d} \quad (4.6)$$

where, the discrete parameters a , b_0 , and b_1 , are obtained as

$$\begin{aligned} a &= p \\ b_0 &= K(1 - a)(1 - r) \\ b_1 &= K(1 - a)r \end{aligned}$$

Applying Equation (4.6) to each transfer function elements of the MIMO transfer function matrix for a pre-determined sampling time T_s , we can obtain the MIMO discrete description of the process, denote

$$y(k) = \tilde{G}(z^{-1})u(k)$$

where $u(k)$ is the $n \times 1$ input vector, $y(k)$ is the $n \times 1$ output vector and $\tilde{G}(z^{-1})$ is the transfer function matrix of $n \times n$ dimension. Each element $\tilde{G}_{ij}(z^{-1})$ in the

transfer function matrix $\tilde{G}(z^{-1})$ is of SISO transfer function form

$$\tilde{G}_{ij}(z^{-1}) = \frac{z^{-1}\tilde{B}_{ij}(z^{-1})}{\tilde{A}_{ij}(z^{-1})} z^{-d_{ij}} \quad (4.7)$$

where d_{ij} is the dead time of the transfer function between the j -input and i -output.

Define $A'_i(z^{-1})$ as the least common multiple of $\tilde{A}_{ij}(z^{-1})$ for $j=1,2,\dots,n$,

Equation (4.7) can be rewritten as

$$\tilde{G}_{ij}(z^{-1}) = \frac{z^{-1}B'_j(z^{-1})}{A'_i(z^{-1})} \quad (4.8)$$

where the elements $B'_j(z^{-1})$ can be easily determined from Equations (4.7) and (4.8).

The Controller Auto-Regressive and Integrated Moving-Average (CARIMA) model for $n \times n$ MIMO process can be obtained from the transfer matrix presentation of Equation (4.8), as follows

$$A(z^{-1})y(k) = B(z^{-1})u(k-1) + \frac{1}{\Delta} C(z^{-1})e(k) \quad (4.9)$$

where $A(z^{-1})$ is a diagonal matrix with element $A'_i(z^{-1})$, $B(z^{-1})$ is a polynomial matrix with elements $B'_j(z^{-1})$, $C(z^{-1})$ is a $n \times n$ polynomial matrix representing the polynomials of noise. $e(k)$ is a $n \times 1$ noise vector at time k and $\Delta = 1 - z^{-1}$. As Δ is a difference operator, for any matrix $A(z^{-1})$, $\Delta A(z^{-1})$ will be used in stead of $\Delta \cdot A(z^{-1})$. $A(z^{-1})$, $B(z^{-1})$, $C(z^{-1})$ can be defined as

$$\begin{aligned} A(z^{-1}) &= I + A_1 z^{-1} + A_2 z^{-2} + \dots + A_{n_a} z^{-n_a} \\ B(z^{-1}) &= B_0 + B_1 z^{-1} + B_2 z^{-2} + \dots + B_{n_b} z^{-n_b} \\ C(z^{-1}) &= I + C_1 z^{-1} + C_2 z^{-2} + \dots + C_{n_c} z^{-n_c} \end{aligned}$$

4.3 Multi-step Prediction

For the $n \times n$ MIMO process described by Equation (4.9), to obtain the predictive j -step outputs, let $C(z^{-1}) = I_{n \times n}$ without loss of generality and consider the following Diophantine equation

$$I_{n \times n} = E_j(z^{-1})\Delta A(z^{-1}) + z^{-j}F_j(z^{-1}) \quad (4.10)$$

where $E_j(z^{-1})$, $F_j(z^{-1})$ and $\Delta A(z^{-1})$ are polynomial matrices of order $j-1$, n_a and $n_a - 1$, with the form

$$\begin{aligned} E_j(z^{-1}) &= E_{j,0} + E_{j,1}z^{-1} + \dots + E_{j,j-1}z^{-(j-1)} \\ F_j(z^{-1}) &= F_{j,0} + F_{j,1}z^{-1} + \dots + F_{j,n_a}z^{-n_a} \end{aligned}$$

and

$$\begin{aligned} \Delta A(z^{-1}) &= I_{n \times n} + \Delta A_1 z^{-1} + \Delta A_2 z^{-2} + \dots + \Delta A_{n_a} z^{-n_a} \\ &= I_{n \times n} + (A_1 - I_{n \times n})z^{-1} + (A_2 - A_1)z^{-2} + \dots + (A_{n_a} - A_{n_a-1})z^{-n_a} - A_{n_a} z^{-(n_a+1)} \end{aligned}$$

respectively.

Multiplying $\Delta E_j(z^{-1})z^j$ on both sides of Equation (4.9), and using Equation (4.10), it results

$$y(k+j) = F_j(z^{-1})y(k) + E_j(z^{-1})B(z^{-1})\Delta u(k+j-1) + E_j(z^{-1})e(k+j) \quad (4.11)$$

Notice that the noise term in Equation (4.11) are all in the future and the expectation of $e(t)$, $E[e(t)] = 0$, thus the predicted output for $y(k+j)$ can be obtained as

$$\hat{y}(k+j|k) = E[y(k+j)] = F_j(z^{-1})y(k) + E_j(z^{-1})B(z^{-1})\Delta u(k+j-1) \quad (4.12)$$

To solve Equation (4.12), consider the difference of the two Diophantine equations corresponding to the predictions $\hat{y}(k+j+1|k)$ and $\hat{y}(k+j|k)$, we have

$$0_{n \times n} = [E_{j+1}(z^{-1}) - E_j(z^{-1})] \Delta A(z^{-1}) + z^{-j} [z^{-1} F_{j+1}(z^{-1}) - F_j(z^{-1})] \quad (4.13)$$

in which the matrix $(E_{j+1}(z^{-1}) - E_j(z^{-1}))$ is of degree j and can be written as

$$E_{j+1}(z^{-1}) - E_j(z^{-1}) = T(z^{-1}) + R_j z^{-j} \quad (4.14)$$

where $T(z^{-1})$ is a $n \times n$ polynomial matrix of degree lower than or equal to $j-1$, and R_j is a real matrix of $n \times n$ dimension. Substituting Equation (4.14) into Equation (4.13), results

$$0_{n \times n} = T(z^{-1}) \Delta A(z^{-1}) + z^{-j} [R_j \Delta A(z^{-1}) + z^{-1} F_{j+1}(z^{-1}) - F_j(z^{-1})] \quad (4.15)$$

For Equation (4.15) to hold, it is clear that both terms on the right hand side of the equation equal to $0_{n \times n}$. Since $\Delta A(z^{-1})$ is a monic polynomial matrix, it can be concluded that $T(z^{-1}) = 0_{n \times n}$. Therefore, $E_{j+1}(z^{-1})$ of Equation (4.14) can be computed recursively by

$$E_{j+1}(z^{-1}) = E_j(z^{-1}) + R_j z^{-j}$$

Using the coefficients of $\Delta A(z^{-1})$ and $F_j(z^{-1})$ defined in Equation (4.10), R_j and $F_{j+1}(z^{-1})$ can be recursively obtained as

$$\begin{aligned} F_{j,0} &= R_j \\ F_{j+1,i} &= F_{j,i+1} - R_j \Delta A_{i+1} \quad \text{for } i = 0, 1, \dots, n_a \end{aligned} \quad (4.16)$$

with the initial conditions

$$E_1 = I, \quad F_1 = z(I - \Delta A).$$

for the computation of Equation (4.14).

Now define

$$E_j(z^{-1}) B(z^{-1}) = G_j(z^{-1}) + z^{-j} G_p(z^{-1})$$

and substitute it into Equation (4.12)

$$\hat{y}(k+j|k) = F_j(z^{-1})y(k) + G_p(z^{-1})\Delta u(k-1) + G_j(z^{-1})\Delta u(k+j-1) \quad (4.17)$$

Note that in Equation (4.17), the first two terms of the right hand side $F_j(z^{-1})y(k)$ and $G_p(z^{-1})\Delta u(k-1)$ depend only on the past information of the process output and input variables, they represent the free response of the process if the control signals are kept constant. While $G_j(z^{-1})\Delta u(k+j-1)$ depends on the future values of the control signals, it can be interpreted as the forced response. Let

$$f_j = F_j(z^{-1})y(k) + G_p(z^{-1})\Delta u(k-1) \quad \text{for } j = 1, 2, \dots$$

then Equation (4.17) can be rewritten as

$$\hat{y}(k+j|k) = G_j(z^{-1})\Delta u(k+j-1) + f_j \quad (4.18)$$

For a set of N -step ahead prediction

$$\begin{aligned} \hat{y}(k+1|k) &= G_1(z^{-1})\Delta u(k) + f_1 \\ \hat{y}(k+2|k) &= G_2(z^{-1})\Delta u(k+1) + f_2 \\ &\vdots \\ \hat{y}(k+N|k) &= G_N(z^{-1})\Delta u(k+N-1) + f_N \end{aligned} \quad (4.19)$$

using the recursive properties of the polynomial matrix E_j , and define

$G_j(z^{-1}) = \sum_{i=0}^{j-1} \bar{G}_i z^{-i}$, Equation (4.19) can be presented in a matrix form

$$\begin{bmatrix} \hat{y}(k+1|k) \\ \hat{y}(k+2|k) \\ \vdots \\ \hat{y}(k+j|k) \\ \vdots \\ \hat{y}(k+N|k) \end{bmatrix} = \begin{bmatrix} \bar{G}_0 & 0 & \dots & 0 & \dots & 0 \\ \bar{G}_1 & \bar{G}_0 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \bar{G}_{j-1} & \bar{G}_{j-2} & \dots & \bar{G}_0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{G}_{N-1} & \bar{G}_{N-1} & \dots & \dots & \dots & \bar{G}_0 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+j-1) \\ \vdots \\ \Delta u(k+N-1) \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_j \\ \vdots \\ f_N \end{bmatrix} \quad (4.20)$$

where the free response term can be calculated recursively

$$f_{j+1} = z(I - \Delta A(z^{-1}))f_j + B(z^{-1})\Delta u(t+j)$$

with $f_0 = y(t)$ and $\Delta u(t+j) = 0$ for $j \geq 0$.

4.4 General Formulation of PID Controllers with Multi-step Prediction

To apply the future information presented in Equation (4.20) for updating the parameters of PID controllers, an incremental expression is often used. The conventional decentralized discrete configuration of PID controllers for the MIMO process can be represented as

$$u(k) = e(k)K_p + \sum_{j=0}^k e(j)K_i + [e(k) - e(k-1)]K_d \quad (4.21)$$

where, $u(k)$, $e(k)$, K_p , K_i , K_d are defined as,

$$u(k) = [u_1(k) \quad u_2(k) \quad \cdots \quad u_n(k)]^T$$

$$e(k) = \begin{bmatrix} e_1(k) & 0 & \cdots & 0 \\ 0 & e_2(k) & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & e_n(k) \end{bmatrix}$$

$$K_p = [k_p^1 \quad \cdots \quad k_p^i \quad \cdots \quad k_p^n]^T$$

$$K_i = [k_i^1 \quad \cdots \quad k_i^i \quad \cdots \quad k_i^n]^T$$

$$K_d = [k_d^1 \quad \cdots \quad k_d^i \quad \cdots \quad k_d^n]^T$$

The incremental control law can be determined by taking the difference operator on both sides of Equation (4.21) such that

$$\begin{aligned} \Delta u(k) = u(k) - u(k-1) &= e(k)[K_p + K_i + K_d] + \\ &e(k-1)[-K_p - 2K_d] + e(k-2)K_d \end{aligned} \quad (4.22)$$

or in a compact matrix form

$$\Delta u(k) = E(k)W \quad (4.23)$$

with

$$\Delta u(k) = [\Delta u_1(k) \quad \Delta u_2(k) \quad \cdots \quad \Delta u_n(k)]^T$$

$$E(k) = \begin{bmatrix} E_1(k) & 0 & \cdots & 0 \\ 0 & E_2(k) & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & E_n(k) \end{bmatrix}$$

$$W = [W_1 \quad W_2 \quad \cdots \quad W_n]^T$$

and

$$W_i = [k_p^i + k_i^i + k_D^i \quad -2k_D^i - k_p^i \quad k_D^i]$$

$$E_i(k) = [e_i(k) \quad e_i(k-1) \quad e_i(k-2)]$$

for $i=1,2,\dots,n$. The dimensions of each elements, respectively, are: $\Delta u \in R^{n \times 1}$, $e \in R^{n \times 1}$, $K_p \in R^{n \times 1}$, $K_i \in R^{n \times 1}$, $K_d \in R^{n \times 1}$, $W \in R^{3n \times 1}$ and $E \in R^{n \times 3n}$.

The difference between classical PID controllers and MPC is that in the classical PID controller, the coefficient vector W is fixed (constant), such that the controller coefficients at time k equal those at time $k+1$ for all time. While in the multi-step predictive PID control scheme, the coefficient vector W might be time varying during the process transition period according a criterion that minimize a certain cost function, that is

$$W_{k+j} \neq W_k, \quad j=1,2,\dots,N$$

where N is the prediction output horizon.

Define the cost function by the following finite horizon quadratic criterion

$$J = \sum_{j=N_1}^{N_2} [r(k+j) - \hat{y}(k+j|k)]^T [r(k+j) - \hat{y}(k+j|k)] + \lambda \sum_{j=1}^{N_u} [\Delta u(k+j-1)]^T [\Delta u(k+j-1)] \quad (4.24)$$

where $\hat{y}(k+j|k)$, $r(k+j)$ and λ are j -step ahead prediction of the system output up to k , a future reference sequence for the output vector and weight elements on the feedback control signals $u(k)$, respectively. λ indicates the degree to which the control actions are considered.

Substituting Equation (4.23) into Equation (4.18) yields

$$\hat{y}(k+j|k) = G_j(z^{-1})E(k+j-1)W(k+j-1) + f_j$$

The cost function becomes

$$J = \sum_{j=N_1}^{N_2} [r(k+j) - G_j(z^{-1})E(k+j-1)W(k+j-1) - f_j]^T [r(k+j) - G_j(z^{-1})E(k+j-1)W(k+j-1) - f_j] + \lambda \sum_{j=1}^{N_u} [E(k+j-1)W(k+j-1)]^T [E(k+j-1)W(k+j-1)] \quad (4.25)$$

Let $N_1 = 1$, $N_2 = N$ and $N_u = N_2 - N_1 + 1$, denote

$$W_N = \begin{bmatrix} W(k) \\ \vdots \\ W(k+N-1) \end{bmatrix}_{N \times 1}, \quad B_N = \begin{bmatrix} r(k+1) - f_1 \\ \vdots \\ r(k+N) - f_N \end{bmatrix}_{N \times 1},$$

$$E_N = \begin{bmatrix} E(k) & 0 & \dots & 0 \\ 0 & E(k+1) & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & E(k+N-1) \end{bmatrix}_{N \times N}$$

and assume that the system input and output pairing have been made. From the decentralized control structure, we have $\Delta u_N = E_N W_N$.

Writing Equation (4.25) in the matrix form

$$J = [B_N - GE_N W_N]^T [B_N - GE_N W_N] + \lambda [E_N W_N]^T [E_N W_N]$$

Rearrange above equation to result in

$$\begin{aligned} J &= [B_N^T - W_N^T E_N^T G^T] [B_N - GE_N W_N] + \lambda [W_N^T E_N^T] [E_N W_N] \\ &= B_N^T B_N - W_N^T E_N^T G^T B_N - B_N^T GE_N W_N + W_N^T [E_N^T (G^T G + \lambda I) E_N] W_N \end{aligned} \quad (4.26)$$

Differentiating Equation (4.26) with respect the vector W_N

$$\frac{\partial J}{\partial W_N} = 2 [E_N^T (G^T G + \lambda I) E_N] W_N - 2 (E_N^T G^T B_N)$$

Minimizing J requires $\frac{\partial J}{\partial W_N} = 0$, and W_N can be solved as

$$W_N = [E_N^T (G^T G + \lambda I) E_N]^{-1} (E_N^T G^T B_N) \quad (4.27)$$

Equation (4.27) provides the best decentralized PID controller settings in terms of minimizing the cost function (4.25). Because the receding horizon strategy is used, the control action $\Delta u(k)$ applied to the process is calculated recursively. The optimal control solution is required to be causal. The solution to this problem can be obtained by substituting the necessary parameters into the cost function and solving for required control coefficients.

- At time k , the vector of controller parameters $W(k)$ is used as the PID controller parameters to calculate the control action $\Delta u(k)$.
- At time $k+1$, a new vector of controller parameters $W(k+1)$ is used as the PID controller parameters to calculate the control action $\Delta u(k+1)$.

4.5 Tuning of Decentralized PID Parameters with Multi-step Prediction

Equation (4.27) provides a general formula for searching the “best” direction of PID parameter settings for decentralized control systems based on multi-step prediction cost function of Equation (4.24). The procedure can be applied to fine tune the decentralized PID controllers derived from ERGA and process dynamic information given in Chapter 3. Similarly, three different cases are considered as follows:

Case 1: Loop 1 is much faster than Loop 2

Using the results of Chapter 3, the very fast loop 1 can be tuned like a single loop controller regardless of the value ϕ . In this situation, the dimension is reduced to the single loop prediction as it can provide the sufficient tuning accuracy. Therefore, a single loop prediction formula based on the single loop model of Equation (4.6) can be applied.

Based on the single loop CARIMA model of Equation (4.9), where $A(z^{-1})$, $B(z^{-1})$ here are polynomials in the backward shift operator z^{-1} with $\deg A = n_a$, $\deg B = n_b$, similar procedure can be applied as the multi-step MIMO system prediction, a Diophantine equation is defined as

$$\begin{cases} 1 = E_j(z^{-1})\Delta A(z^{-1}) + z^{-j}F_j(z^{-1}) \\ E_j(z^{-1})B(z^{-1}) = G_j(z^{-1}) + z^{-j}H_j(z^{-1}) \end{cases}$$

where E_j , F_j , H_j and L_j are polynomials of the backward shift operator z^{-1} . And the j-step prediction can be easily obtained as Equation (4.18),

$$\hat{y}(k+j|k) = G_j(z^{-1})\Delta u(k+j-1) + f_j$$

where $f_i = F_j y(k) + H_j \Delta u(t-1)$. And a matrix form as Equation (4.20) can be rewritten as

$$\underbrace{\begin{bmatrix} \hat{y}(k+1|k) \\ \hat{y}(k+2|k) \\ \vdots \\ \hat{y}(k+N|k) \end{bmatrix}}_y = \underbrace{\begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N-1} & g_{N-2} & \cdots & g_0 \end{bmatrix}}_G \underbrace{\begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N-1) \end{bmatrix}}_{\Delta u} + \underbrace{\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}}_f$$

The corresponding optimal solution which is applied for fine tuning can be obtained by Equation (4.27), with element $E(k+j)$ in E_N defined for the fast loop 1 as

$$E(k+j) = [e_1(k+j) \quad e_1(k+j-1) \quad e_1(k+j-2)]$$

and elements of B_N are based on reference value of r and f from the single loop prediction.

Case 2: Loop 1 is much slower than Loop 2

From Chapter 3, the gain of the process “seen” by controller loop 1 in the decentralized control system is changed by $1/\phi$, from the single loop gain (K_{11}).

Similar procedures as in Case 1 can be employed to obtain the prediction information from the single loop model, and the obtained controller parameters from Equation (4.27) is multiplied with the ERGA index ϕ for $\phi < 1$, or keep same for $\phi \geq 1$.

Case 3: Loops 1 and 2 have the similar dynamics

Unlike case 1 and 2, in this case, both loops will be considered and the fine tuning of PID parameters should follow Equation (4.27). However, direct use of Equation (4.24) and consequently the result obtained from Equation (4.27) may cause two problems due to the decentralized control structure:

1. Equation (4.27) is the optimal solution for Equation (4.24) without considering the stability issue. Since the closed-loop performance in the decentralized control system can not achieve a better performance than the performance when the other loop is open, there is deterioration in performance caused by the interactions from the other loop. Therefore, the decentralized control structure may pose certain limitations to the aggressiveness of the solution.
2. The existence of W_N requires that the matrix to be inverted in Equation (4.27) has a full rank at every recursive step, while in some situations, W_N may not have a full rank due to the limited diagonal elements of E_N .

The solutions of above two problems are proposed as follows, which constitutes the main contribution of this Chapter. Since the closed-loop performance in the decentralized control system can not achieve a better performance than the performance when the other loop is open, the original reference signals may be too aggressive to be realized. Therefore, a realizable reference is introduced to make the solution of Equation (4.27) more reasonable. The logical candidate for such references is the closed-loop output response of the single loop without considering interaction from the other loop, as the existence of the loop interactions when the other loop is closed can only deteriorate the system performance. Let the controller determined by single loop tuning rules with PID formulation,

$$g_c(z^{-1}) = \frac{g_0 + g_1 z^{-1} + g_2 z^{-2}}{1 - z^{-1}}$$

with

$$\begin{cases} g_0 = k_p + k_I + k_D \\ g_1 = -2k_D - k_p \\ g_2 = k_D \end{cases}$$

and discrete transfer function model $g(z^{-1})$ is obtained as Equation (4.6), consider the reference $r(k)$, the revised reference (the projected single loop output response without interaction) can be easily obtained as

$$\bar{r}(k) = \frac{g_c(z^{-1})g(z^{-1})}{1 + g_c(z^{-1})g(z^{-1})} r(k)$$

Replace the reference vector in Equations (4.24) and (4.27), $r(k+j)$, by the new vector $\bar{r}(k+j)$, solution based on the realizable references can be obtained and illustrated as in Fig. 4.1, where curve 1 is the closed-loop response of the single loop without interaction, while curve 2 is the response when the other loop is closed. Even though both curves are the responses to the same reference, curve 1 will put a limit on curve 2 for both performance and stability. Otherwise, the solution of Equation (4.27) to the cost function Equation (4.24) may sometimes get over the limit and the particular loop will then become unstable.

Fig. 4.1 Closed-loop responses with loops open and close

To solve the second problem, a two-level multi-step prediction with PID control structure is proposed as shown in Fig. 4.2. The Lower level is the conventional PID control system with its initial setting specified using the method

proposed in Chapter 3, while the higher level adjusts PID controller parameters through predicting the output of the system and minimizing the generalized predictive control cost function.

Fig. 4.2 Multi-step prediction fine-tuning diagram

The advantage of using this two-level structure is that the control signal Δu is realized through PID controller function, rather than recursively calculated in real time. Since the both process and PID controllers are considered linear at the specified working point, the solution Equation (4.27) to cost function Equation (4.24) at ever recursive step is close to optimal if the prediction horizon is no less than 3. As our objective here is to achieve fine tuning of the decentralized control setting already been obtained in Chapter 3, a constant coefficient “suboptimal” solution will be obtained by letting

$$W_{k+1} = W_k$$

and stop the recursive calculation if E_N is not full rank. As a rule of thumb, set the prediction horizon and control horizon to $3 \leq N \leq 5$ for the proposed multi-step

prediction and the fine tuning is suggested only in case of large system errors.

The proposed multi-step prediction method together with the decentralized PID controller tuning method given in Chapter 3 can be implemented in real applications using the following procedure.

Step 1: Initial decentralized PID setting

1. Determine the ERGA index for the designed system
2. Calculate the controller settings for the selected transfer functions
3. Multiply detuning factor according to ERGA index.
4. If a fine tuning is needed, go to Step 2.

Step 2: Fine tuning of the initial PID

1. Convert the system to discrete time domain using Equation (4.6)
2. Obtain the CARIMA model of the system by Equation (4.9).
3. Choose the prediction and control horizon, define the cost function and calculate the prediction matrix in Equation (4.24).
4. Calculating the predicted future error E_N of Equation (4.25) using the current PID controller.
5. Determining the optimal solution of $W(k+1)$ from Equation (4.27).
6. If further fine tuning is required, iterative 3 and 4. Otherwise, implement the current controller settings.

4.6 Simulation Examples

In this section, several simulation examples are provided to compare the multi-step prediction based PID tuning method with the conventional PID tuning methods to show its effectiveness. In order to measure the performance differences, a cost function defined as

$$J(e, \Delta u) = \sum_{i=1}^2 \sum_{t=0}^{istop} [e_i^2(t) + 0.1 \Delta u_i^2(t)]$$

is used throughout the comparison studies.

Example 4.1 Consider the well-known Wood and Berry TITO binary distillation column plant with the transfer function given as in Example 2.1. It is a typical TITO plant with strong interactions and significant time delays.

It is easy to obtain the RGA and ERGA as follows

$$\Phi_{RGA} = \begin{bmatrix} 2.01 & -1.01 \\ -1.01 & 2.01 \end{bmatrix} \text{ and } \Phi_{ERGA} = \begin{bmatrix} 2.12 & -1.12 \\ -1.12 & 2.12 \end{bmatrix}$$

This result implies the diagonal pairing 1-1/2-2 is the right pairing selection. Following ERGA tuning rules, controllers are designed for the two SISO systems. The ultimate gain K_u and ultimate frequency ω_u of each individual diagonal transfer function are obtained as, $K_{u1} = 2.1$, $\omega_{u1} = 1.61$, $K_{u2} = 0.43$, and $\omega_{u2} = 0.71$.

The PI controllers based on Z-N tuning rule leads to the following parameters

$$\begin{aligned} \text{Loop 1:} \quad & \text{PI} - K_p = 0.84, K_I = 0.27 \\ \text{Loop 2:} \quad & \text{PI} - K_p = -0.17, K_I = -0.024 \end{aligned}$$

As $\phi > 1$, a set of PI controller parameters can be detuned according to Equation (3.19). The detuning factor calculated to multiply the original controller gains is 0.57. For comparison, the PI controller tuning method by Luyben [78],

which is widely used as a benchmark for practical PI controller tuning, and a more recently proposed method by Chen, are also simulated [88]. The PI controller setting of Luyben, that of Chen, and that detuned by ERGA are shown in Table 4.1.

Table 4.1 PI controller settings for Example 4.1

PI Parameters	Luyben PI	Chen PI	ERGA PI
K_{P1}	0.38	0.44	0.47
K_{I1}	0.05	0.09	0.15
K_{P2}	-0.08	-0.09	-0.1
K_{I2}	-0.003	-0.009	-0.01

Closed-loop responses of the proposed PI controller compared with Luyben and Chen’s methods are shown in Fig. 4.3, where the set-point changes occur at $t = 0$ ($X_D = 1, X_B = 0$) and at $t = 200$ ($X_B = 1$), and unit load disturbance occurs at $t = 400$.

Fig. 4.3 PI closed-loop responses for Example 4.1

To improve the performance, the online multi-step prediction tuning method is applied with both the prediction and control horizons set at 3. The cost functions of each simulated control system and the corresponding closed-loop response are calculated and shown in Table 4.2 and Fig. 4.4, respectively. From simulation and comparison results, it is clear that a better control performance is achieved through the proposed multi-step tuning method in viewing of less oscillatory performance for both set-point changes and disturbances.

Table 4.2 Cost value comparison for Example 4.1

Controller type	Luyben PI	Chen PI	ERGA PI	MP PI
$J(e,\Delta u)$	34.57	23.87	24.45	19.98

Fig.4.4 Fine-tuning controller performance for Example 4.1

Example 4.2 Consider the industrial-scale polymerization reactor of Example 3.3, the PID controller settings of proposed ERGA method, as well as Luyben and Chen’s methods for comparison are shown in Table 4.3. Closed-loop responses of the proposed PI controller compared with Luyben and Chen’s methods are shown in Fig. 4.5, where the unit set-point changes for y_1 at $t = 0$ and y_2 at $t = 30$, and unit load disturbance occurs at $t = 60$.

Table 4.3 PI controller settings for Example 4.2

PI Parameters	Luyben PI	Chen PI	ERGA PI
K_{P1}	0.21	6.67	0.36
K_{I1}	0.09	1.67	0.34
K_{P2}	0.18	6.41	0.29
K_{I2}	0.04	1.06	0.15

Fig. 4.5 PI closed-loop responses for Example 4.2

As we can see, the ERGA PI setting for this example are more oscillatory for both set-point changes and disturbances comparing with the other two methods. Similar as Example 4.1, the proposed ERGA detuning method provide an acceptable control performance. However, compared with Luyben and Chen’s methods, it is less conservative. Luyben’s method is more conservative setting that requires more iterative steps to find the detuning factor. On the other hand, Chen’s method needs a pre-compensator or static decoupler to make the process dominant before designing decentralized controllers.

To improve the performance, the online multi-step prediction tuning method is applied by setting the prediction and control horizons both equal to 3. The cost function for each simulated controller and the corresponding closed-loop response are calculated and shown in Table 4.4 and Fig. 4.6 respectively. Simulation and comparison results verified the validity of the proposed multi-step tuning method. It is clear that a better control performance is also achieved with less oscillation for both set-point changes and disturbances.

Table 4.4 Cost value comparison for Example 4.2

Controller type	Luyben PI	Chen PI	ERGA PI	MP PI
$J(e, \Delta u)$	28.97	24.91	32.12	17.35

Fig.4.6 Fine-tuning controller performance for Example 4.2

Chapter 5

Construction of Global Multi-model Control Structure

5.1 Introduction

Multi-model approach provides an effective way to deal with the difficult control problems of complex nonlinear MIMO systems by partitioning the system full operating range into a set of smaller regimes in which the overall control problems can be solved. Due to complexity of the modern processes, this research topic has drawn intensive interest in the past few years. Many multi-model approaches have been proposed and applied to industrial process control systems. Among those approaches, Murray and Johansen [24] proposed four possible axes to

divide the operation regions. Foss *et al.* [98] suggested a strategy for decomposing of the nonlinear space into multiple linear structures using a priori process information. A heuristic strategy was also presented by Johansen and Foss [29], where decomposition of operating regimes follows a search strategy guided by heuristics and local model validity function. Similarly, Ogunnaike and Ray [3] proposed a steady state map based on a priori knowledge of the input and output ranges to decompose the whole operating region into simpler regimes. Oliver Hecker *et al.* [99] developed a local linear model tree for identifying a nonlinear heat exchange process, which combines the classification and regression tree [100] and multiple adaptive regression algorithms [101]. Banerjee *et al.* [102] presented a decomposition based on a bayesian representation of local models. And the well known Takagi-Sugeno [103] fuzzy modeling method is probably the most popular multi-model approach for nonlinear system representation by using the fuzzy interpolations.

Generally, there are four basic problems involved when developing a multi-model control approach: 1) construction of local models; 2) controller design for local models; 3) selection of proper number of operating regimes; and 4) determination of overall control output. The first two problems have already been discussed in Chapters 2, 3 and 4. In this chapter, we will focus on the last two problems, i.e. the approaches to obtain a proper number of local regimes and the combined local controller outputs. Normally, multi-model methods make use the possible causes of nonlinearity to partition the operating range. For example, the operating regimes can be defined according to the set-points values, when the set-point values change, a set of data is collected to identify the local model. However, the major constraint for such a division strategy, also known as model banks, is that the underlying nonlinearities of the practical process and some of its physical parameters have to be known a priori. Since many controllers such as the widely

used PID controller may not lead to a poor performance when set-point value changes, the model banks based on expert experience may not provide a proper division of the operating regimes.

Different from the existing methods, a novel methodology to progressively divide the operating regimes is proposed in this chapter. An additional local controller as well as model is to be added to the global multi-model control system only when unsatisfied performance occurs. The argument for such a division strategy is that the objective of the proposed multi-model control system is to achieve a satisfactory performance with minimum number of local controller. Consequently, the control performance based division will provide the proper structure, i.e. a minimum number of local models, for the whole operating range. In order to achieve such divisions, monitoring and assessment of control system performance is required. The study of this topic has relatively long history [104-106], and may date back to the work of Astrom [107], when the Minimum Variance Control (MVC) criterion was proposed. The system input and output data is collected and compared with possible MVC input and output data to indicate how good the existing controller performance is. After a period of dreariness, a recent resurgence of this research has gained the focus of researchers' attention again due to the work of Harris in 1989 [108]. In his work, Harris proposed to use the MVC as a lower bound to assess the performance of single loop controllers, where the lower bound is estimated from closed-loop operating data. Recently, research on control system evaluation methods based on the values of MPC cost function have been very active, as it offers a convenient way for the MIMO system analysis [109-112].

Once the multi-model structure has been determined, the overall controller output will be obtained by combining the local controller outputs which depends on how the operating regimes are partitioned. Normally, there are two main methods to

obtain the overall controller output, i.e. hard combination and fuzzy combination [24]. For hard combination, there are no overlaps between the local models, and the global approximation is a piecewise approximation of the local ones. In practice, the overall controller output of hard combination is generally set by the process specialist. A supervisory structure is often applied to handle the various situations such as startup, shutdown, exceptions, and load or product changes. It is obvious that hard combination with crisp boundaries lacks the sufficient treatment for the nonlinearity at the intermediate or transition regimes, since the crisp split functions themselves are discontinuous in nature. For MIMO processes, it is more difficult to find a hard partition for the multi-dimensional operating range. In contrast, fuzzy combination, also called soft combination, is suitable for the situation where there is no clear separation between operating regimes such as no sudden behavior change between the operating regimes. The local models or controllers are combined by the soft boundaries between different regimes, using the fuzzy set [17, 103] and interpolation method [24]. Through membership of fuzzy sets, the combination of operating regimes can be described more smoothly and naturally. In process industry, the nonlinearities are usually change gradually with the changing operating regimes, the fuzzy means will be more suitable for combining the local models in multi-model control.

In this chapter, a novel multi-model control structure is proposed. By adopting fuzzy combination method, the characteristic of every operating point inside the operating window is represented by the existing local models/controllers. For a given arbitrary number of existing local models/controllers, the repartitioning of the whole operating range is achieved through an incremental strategy based on a performance assessment criterion of the existing control structure. An additional local regime is added and boundaries of fuzzy functions for partitioning are redistributed only when an unsatisfactory controller performance caused by model

mismatch is detected. Comparing with existing multi-model methods, the proposed method is simple, straightforward, without requiring a priori process knowledge, and can be fully automated.

5.2 General Multi-model Control Structure

Consider a general complex nonlinear system, using multi-model control concept, the whole system can be partitioned into n local regimes. In each regime, the local models and the corresponding decentralized controllers have been identified and designed following the procedures of Chapter 2, 3 and 4, respectively. Fig. 5.1 illustrates the structure of the multi-model control systems.

Fig. 5.1 Multi-model controller structure

The partition of regime can be determined by either reference signal or manipulated variable working points. In general, for an operating point at time t of a $d \times d$ system, we can define the operating point $z \in Z = R^d$ as

$$z(t) = \begin{bmatrix} z_1(t) \\ \vdots \\ z_d(t) \end{bmatrix}$$

where Z is the operating space. Following the multi-model principle, the operating space Z can be partitioned into n subsets Z_i , $i \in I_n = \{1, 2, \dots, n\}$, which corresponds to the n local models and controllers, and

$$Z = \bigcup_{i \in I_n} Z_i$$

For each operating regime Z_i , a local validity function $\rho_i(z)$ can be defined to indicate the relative validity of the local regimes to the overall system operating range. Consider input u , which can be the input to either the local models or the local controllers. In correspondence to the input, let the local regime output be f_i , $i = 1, 2, \dots, n$. Then, the global output y can be represented as

$$y(u) = \sum_{i \in I_n} f_i(u) w_i(z) \quad (5.1)$$

where

$$w_i(z) = \frac{\rho_i(z)}{\sum_{j \in I_n} \rho_j(z)} \quad \text{for } i = 1, 2, \dots, n$$

are the interpolation functions. If the local models and the corresponding control systems are defined as M_i , $i = 1, 2, \dots, n$, the overall multi-model structure can be characterized as shown in Fig.5.2.

Fig. 5.2 Multiple local regimes structure

5.3 Performance Assessment of Control Systems

In multi-model strategy, the overall operating window is divided into smaller regimes to simplify the complex nonlinear relationships. Since the computational effort is proportional to the number of local models, there exists a trade-off between the number of regimes and the resulting control system performances. Unlike the existing predetermined number of multi-model approaches, we here propose an incremental procedure for constructing the multi-model control systems. That is, based on the designed local controllers, the system performances are checked and an additional local model as well as controller will be added into the overall system only if they are needed.

To determine the necessity of an additional regime, the system performance has to be checked. However, the widely used performance evaluation methods, such as MVC or the MPC cost function based performance criteria used to evaluate fixed

model controller performances, may not be suitable for this incremental multi-model structure. We propose a new criterion to evaluate the performance of the systems based on the existing local models and to decide if an addition regime is needed according to the performance evaluation criteria.

Assuming a multi-model system currently composed of j local models with j corresponding local decentralized controllers. The interpolation function values of local controllers are based on the local model validity function which indicates the relative effectiveness of the local controllers to the overall control system performance. The whole integration of local model and local controller is shown in Fig 5.3.

Fig. 5.3 Integration of local models and local controllers

For a typical TITO system, each local control system consists of two local controllers. Thus the controller output u_i of i^{th} local model, $i = 1, 2, \dots, j$, can be calculated by

$$u_i(k) = \begin{bmatrix} g_{c1}^i(z^{-1}) & 0 \\ 0 & g_{c2}^i(z^{-1}) \end{bmatrix} \begin{bmatrix} e_1(k) \\ e_2(k) \end{bmatrix} \quad (5.2)$$

where $g_{c1}^i(z^{-1})$, $g_{c2}^i(z^{-1})$ are the i^{th} local decentralized controllers, $e_1(k)$ and $e_2(k)$ are the current errors between set-points and corresponding process outputs.

Form interpolation function w_i of i^{th} local control system determined by Equation (5.1), the overall controller output to the process is obtained through integrating of local controllers

$$u(k) = \sum_{i=1}^j u_i(k) w_i \quad (5.3)$$

At every sample instant, the real process output $y(k)$ under the control output u can be measured.

On the other hand, the calculated system output $\tilde{y}_i$ of i^{th} local model can be obtained as

$$\tilde{y}_i(k) = G_i u_i(k)$$

where G_i is the i^{th} local model, and the overall system output $\tilde{y}(k)$ can be obtained based on the combination rules of local output $\tilde{y}_i$ by

$$\tilde{y}(k) = \sum_{i=1}^j \tilde{y}_i(k) w_i \quad (5.4)$$

Here $\tilde{y}(k)$ can be considered as the predicted output based on the available models and it can be used as the standard output to evaluate the multi-model controller performance. Assume that the local controllers have already been designed for the local models with satisfactory performance, there should be very little difference between the real process output and the predicted output if the current operating point can be effectively controlled by $u(k)$ of Equation (5.3). A big difference

between the real process output and the predicted process output implies that the current operating point is not well represented by the existing local models. Then, an additional local model and a corresponding local controller are needed at that operating point.

In order to evaluate the overall control system performance based on the existing local controllers, a MPC like error cost function is introduced. Define a time horizon P , say 50 sampling periods, and the error cost function J_e for the real process control system output as

$$J_e(k) = \frac{1}{P} \left(\sum_{i=1}^P (r(k+i-P) - y(k+i-P))^T (r(k+i-P) - y(k+i-P)) \right) \quad (5.5)$$

and the corresponding error cost function for the predicted process output as

$$\tilde{J}_e(k) = \frac{1}{P} \left(\sum_{i=1}^P (r(k+i-P) - \tilde{y}(k+i-P))^T (r(k+i-P) - \tilde{y}(k+i-P)) \right) \quad (5.6)$$

The performance measure is defined as a ratio

$$\gamma(k) = \frac{\tilde{J}_e(k)}{J_e(k)} \quad (5.7)$$

In every sample time, the error cost functions of Equations (5.5) and (5.6) are computed for a defined time horizon, and the performance ratio is calculated from Equation (5.7). In general, the ratio which reflects the multi-model performance will be smaller than 1 because of the imperfect models, actuator noise or uncertainties.

In practice, $\gamma(k)$ is a stochastic variable and statistical analysis is advocated to detect statistically significant changes in the controller performance. $\gamma(k)$ is assumed to be generated by an autoregressive moving average (ARMA) model

$$A(z^{-1})\gamma(k) = C(z^{-1})v(k) \quad (5.8)$$

where z^{-1} is the backward shift operator, $C(z^{-1})$ and $A(z^{-1})$ are monic polynomials and $v(k)$ is a zero-mean, uncorrelated, Gaussian noise signal. Collecting a sequence of $\gamma(k)$ values in a time interval in which the controller performs as expected, polynomials A and C and the variance of v can be estimated. Since $\gamma(k)$ is highly serially correlated and the autoregressive part is first order [110, 112], Equation (5.8) can be written as

$$(1 - a_1 z^{-1})\gamma(k) = v(k)$$

and define

$$\Delta\gamma(k) = \frac{\hat{C}(z^{-1})}{\hat{A}(z^{-1})} \gamma(k) \quad (5.9)$$

where $\hat{C}(z^{-1})$ and $\hat{A}(z^{-1})$ are estimated polynomials. The estimated noise variance is used to compute 95% confidence intervals on $\Delta\gamma(k)$. Violation of these control limits indicates a statistically significant change in controller performance. According to Equations (5.5) and (5.6), $\Delta\gamma(k)$ is a prediction residual and should have a normal distribution. Prediction residuals are frequently used to monitor variations in autocorrelated random variables. We can use control limits at ± 3 standard deviation (3σ limits) to monitor the residuals $\gamma(k)$ and consequently $\Delta\gamma(k)$.

5.4 Multi-model Division Strategy

For TITO processes, the local regimes in the operating range can be described by 2-dimensional operating points and the overall operating range can thus be characterized by a 2-dimensional operating window. Usually, the two manipulated variables are considered as the variables to decide the operating points, denoted as

$$z(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \end{bmatrix}$$

The fuzzy combination of different operating regimes can be described as following rules

$$R_i : \text{If } \begin{bmatrix} z_1(t) \\ z_2(t) \end{bmatrix} \text{ is } A_i(z_1(t), z_2(t)), \text{ then } y_i = d_i \quad (5.10)$$

where $i = 1, \dots, n$ is the rule index, i.e. the number of local operating regimes, z is considered as the premise value, $A_i(z)$ is a fuzzy set defined on z , y_i is the consequent output of the i^{th} rule, i.e. output value of the local operating regimes.

If the premise space is partitioned by the triangle-shaped membership functions, without loss of generality, suppose two local models and two corresponding local controllers are initially assigned for the process, which are obtained at two operating points Z_0 and Z_1 according to the operating conditions. The models and controllers designed for the operating points are valid for the whole regime with different weighting values. Fig. 5.4 shows current operating regimes distributions, and two fuzzy weight functions defined around these two operating points.

The membership functions of $Z_0(z_1(t))$ and $Z_0(z_2(t))$ are, respectively, defined as

$$Z_0(z_1(t)) : \begin{cases} \rho_{Z_0}(z_1(t)) = 1 & z_1(t) \geq z_1(0) \\ \rho_{Z_0}(z_1(t)) = \frac{z_1(t) - z_1(1)}{z_1(0) - z_1(1)} & z_1(1) < z_1(t) < z_1(0) \\ \rho_{Z_0}(z_1(t)) = 0 & z_1(t) \leq z_1(1) \end{cases}$$

and

$$Z_0(z_2(t)) : \begin{cases} \rho_{Z_0}(z_2(t)) = 1 & z_2(t) \leq z_2(0) \\ \rho_{Z_0}(z_2(t)) = \frac{z_2(1) - z_2(t)}{z_2(1) - z_2(0)} & z_2(0) < z_2(t) < z_2(1) \\ \rho_{Z_0}(z_2(t)) = 0 & z_2(t) \geq z_2(1) \end{cases}$$

Fig. 5.4 Initial distribution

Similar to Z_0 , the membership functions of regime Z_1 : $Z_1(z_1(t))$ and $Z_1(z_2(t))$ are defined as

$$Z_1(z_1(t)) : \begin{cases} \rho_{Z_1}(z_1(t)) = 1 & z_1(t) \leq z_1(1) \\ \rho_{Z_1}(z_1(t)) = \frac{z_1(0) - z_1(t)}{z_1(0) - z_1(1)} & z_1(1) < z_1(t) < z_1(0) \\ \rho_{Z_1}(z_1(t)) = 0 & z_1(t) \geq z_1(0) \end{cases}$$

and

$$Z_1(z_2(t)) : \begin{cases} \rho_{Z_1}(z_2(t)) = 1 & z_2(t) \geq z_2(1) \\ \rho_{Z_1}(z_2(t)) = \frac{z_2(t) - z_2(0)}{z_2(1) - z_2(0)} & z_2(0) < z_2(t) < z_2(1) \\ \rho_{Z_1}(z_2(t)) = 0 & z_2(t) \leq z_2(0) \end{cases}$$

If the combination of existing controllers can not provide the satisfied performance as operating point moves into zones which is not well represented by the two regimes, an additional local model is required, the corresponding local controller needed to be designed, and a new fuzzy function has to be defined, which will affect the fuzzy weights of the existing fuzzy functions. For TITO systems, the additional local operating points will affect the existing distribution in two ways on the axis of the operating window. The fuzzy weights for different situations will be redistributed as follows:

1. Additional operating point is outside the boundaries enclosed by the two regimes of the existing operating points as shown in Fig. 5.5

As the additional Z_2 is added along the boundary of the operating window, the membership functions that need to be redistributed are those that have been defined at the boundary. In Fig. 5.5, it is clear that $\rho_{Z_0}(z_2(t))$ and $\rho_{Z_1}(z_1(t))$ need to be revised according to the added new operating point Z_2 and the resulting revised membership for this kind of situations are defined as following:

For regime Z_0 ,

$Z_0(z_1(t))$: Unchanged

$$Z_0(z_2(t)) : \begin{cases} \rho_{Z_0}(z_2(t)) = \frac{z_2(t) - z_2(2)}{z_2(0) - z_2(2)} & z_2(2) < z_2(t) \leq z_2(0) \\ \rho_{Z_0}(z_2(t)) = \frac{z_2(1) - z_2(t)}{z_2(1) - z_2(0)} & z_2(0) < z_2(t) < z_2(1) \\ \rho_{Z_0}(z_2(t)) = 0 & z_2(t) \leq z_2(2) \text{ or } z_2(t) \geq z_2(1) \end{cases}$$

For regime Z_1 ,

$$Z_1(z_1(t)) : \begin{cases} \rho_{z_1}(z_1(t)) = \frac{z_1(0) - z_1(t)}{z_1(0) - z_1(1)} & z_1(1) \leq z_1(t) < z_1(0) \\ \rho_{z_1}(z_1(t)) = \frac{z_1(t) - z_1(2)}{z_1(1) - z_1(2)} & z_1(2) < z_1(t) < z_1(1) \\ \rho_{z_1}(z_1(t)) = 0 & z_1(t) \leq z_1(2) \text{ or } z_1(t) \geq z_1(0) \end{cases}$$

$Z_1(z_2(t))$: Unchanged

Fig. 5.5 Boundary distribution

For the new regime Z_2 ,

$$Z_2(z_1(t)) : \begin{cases} \rho_{z_2}(z_1(t)) = 1 & z_1(t) \leq z_1(2) \\ \rho_{z_2}(z_1(t)) = \frac{z_1(1) - z_1(t)}{z_1(1) - z_1(2)} & z_1(2) < z_1(t) < z_1(1) \\ \rho_{z_2}(z_1(t)) = 0 & z_1(t) \geq z_1(1) \end{cases}$$

and

$$Z_2(z_2(t)) : \begin{cases} \rho_{Z_2}(z_2(t)) = 1 & z_2(t) \leq z_2(2) \\ \rho_{Z_2}(z_2(t)) = \frac{z_2(0) - z_2(t)}{z_2(0) - z_2(2)} & z_2(0) < z_2(t) < z_2(2) \\ \rho_{Z_2}(z_2(t)) = 0 & z_2(t) \geq z_2(0) \end{cases}$$

2. Additional operating point is inside the enclosure of some of the existing operating points, as shown in Fig. 5.6.

As the additional Z_3 is added inside the boundaries of the obtained operating points Z_0 and Z_1 , it is clear that the membership function of Z_0 and Z_1 , i.e. $\rho_{Z_0}(z_1(t))$, $\rho_{Z_0}(z_2(t))$, $\rho_{Z_1}(z_1(t))$ and $\rho_{Z_1}(z_2(t))$ need to be revised according to the added new operating point Z_3 , while the membership definition of Z_2 that is not aligned will remain as before. The resulting revised memberships for this kind of situations are defined as following:

Fig. 5.6 Additional points inside enclosure of some existing operating points

For regime Z_0

$$Z_0(z_1(t)) : \begin{cases} \rho_{Z_0}(z_1(t)) = 1 & z_1(t) \geq z_1(0) \\ \rho_{Z_0}(z_1(t)) = \frac{z_1(t) - z_1(3)}{z_1(0) - z_1(3)} & z_1(3) < z_1(t) < z_1(0) \\ \rho_{Z_0}(z_1(t)) = 0 & z_1(t) \leq z_1(3) \end{cases}$$

and

$$Z_0(z_2(t)) : \begin{cases} \rho_{Z_0}(z_2(t)) = \frac{z_2(t) - z_2(2)}{z_2(0) - z_2(2)} & z_2(2) < z_2(t) \leq z_2(0) \\ \rho_{Z_0}(z_2(t)) = \frac{z_2(3) - z_2(t)}{z_2(3) - z_2(0)} & z_2(0) < z_2(t) < z_2(3) \\ \rho_{Z_0}(z_2(t)) = 0 & z_2(t) \leq z_2(2) \text{ or } z_2(t) \geq z_2(3) \end{cases}$$

For regime Z_1 ,

$$Z_1(z_1(t)) : \begin{cases} \rho_{Z_1}(z_1(t)) = \frac{z_1(3) - z_1(t)}{z_1(3) - z_1(1)} & z_1(1) \leq z_1(t) < z_1(3) \\ \rho_{Z_1}(z_1(t)) = \frac{z_1(t) - z_1(2)}{z_1(1) - z_1(2)} & z_1(2) < z_1(t) < z_1(1) \\ \rho_{Z_1}(z_1(t)) = 0 & z_1(t) \geq z_1(3) \text{ or } z_1(t) \leq z_1(2) \end{cases}$$

and

$$Z_1(z_2(t)) : \begin{cases} \rho_{Z_1}(z_2(t)) = 1 & z_2(t) \geq z_2(1) \\ \rho_{Z_1}(z_2(t)) = \frac{z_2(t) - z_2(3)}{z_2(1) - z_2(3)} & z_2(3) < z_2(t) < z_2(1) \\ \rho_{Z_1}(z_2(t)) = 0 & z_2(t) \leq z_2(3) \end{cases}$$

For regime Z_3 ,

$$Z_3(z_1(t)) : \begin{cases} \rho_{Z_3}(z_1(t)) = \frac{z_1(0) - z_1(t)}{z_1(0) - z_1(3)} & z_1(3) \leq z_1(t) < z_1(0) \\ \rho_{Z_3}(z_1(t)) = \frac{z_1(t) - z_1(1)}{z_1(3) - z_1(1)} & z_1(1) < z_1(t) < z_1(3) \\ \rho_{Z_3}(z_1(t)) = 0 & z_1(t) \leq z_1(1) \text{ or } z_1(t) \geq z_1(0) \end{cases}$$

and

$$Z_3(z_2(t)) : \begin{cases} \rho_{Z_3}(z_2(t)) = \frac{z_2(t) - z_2(0)}{z_2(3) - z_2(0)} & z_2(0) < z_2(t) \leq z_2(3) \\ \rho_{Z_3}(z_2(t)) = \frac{z_2(1) - z_2(t)}{z_2(1) - z_2(3)} & z_2(3) < z_2(t) < z_2(1) \\ \rho_{Z_3}(z_2(t)) = 0 & z_2(t) \leq z_2(0) \text{ or } z_2(t) \geq z_2(1) \end{cases}$$

Following the two procedures, more operating regimes may be added to the operating window if required.

5.5 Determination of Overall Controller Output

According to the membership functions defined in Section 5.4, each operating point inside the operating window can be given a fuzzy weight by using

$$\rho_{Z_i}(t) = \rho_{Z_i}(z_1(t)) \wedge \rho_{Z_i}(z_2(t)) \quad (5.11)$$

where “ $\wedge$ ” denotes intersection performed on membership functions, via the *min* operator, a subscript *i* denotes the *i*th operating regime. $\rho_{Z_i}(z_1(t))$ and $\rho_{Z_i}(z_2(t))$ denote the two weight values of the corresponding operating axes $z_1(t)$ and $z_2(t)$ for different regimes. The global controller output can consequently be obtained through different fuzzy weights of involved local controllers according to Fig. 5.1, where fuzzy weight for each involved local regime is determined by Equation (5.11).

Fig. 5.7 Weight determination of involved local regimes

Assuming that the local regimes are defined as in Fig. 5.6, there are two situations need to be considered for the determination of global output. Using Z and Z' in Fig. 5.7 as examples to represent operating points under these two situations, the determination procedure of the global output is illustrated as follows.

1. Operating point Z :

As the operating point Z is located between the regimes Z_1 and Z_3 , the global output is the combination of local controllers obtained in Z_1 and Z_3 with their corresponding fuzzy weights. As illustrated in Fig. 5.7, the fuzzy weight for regimes Z_1 and Z_3 can be obtained from Equation (5.10)

$$\rho_{Z_1} = \rho_{z_1}(z_1(t)) \wedge \rho_{z_2}(z_2(t))$$

$$\rho_{Z_3} = \rho_{z_1}(z_1(t)) \wedge \rho_{z_2}(z_2(t))$$

From Equation (5.1), the correspond interpolation values for regimes Z_1 and Z_3 can

be easily determined as

$$w_{Z_1} = \frac{\rho_{Z_1}}{\rho_{Z_1} + \rho_{Z_3}}$$

$$w_{Z_3} = \frac{\rho_{Z_3}}{\rho_{Z_1} + \rho_{Z_3}}$$

and the global controller output can be calculated by using Equation (5.3) based on the local controllers of regimes Z_1 and Z_3 .

2. Operating point Z' :

If operating point is Z' , the fuzzy weights of local regimes Z_0, Z_1, Z_2 , and Z_3 are all equal to 0 according to Equation (5.11). This is the case where Z' does not belong to any of the existing regimes (An uncommon but possible case for our proposed method). For such situations, redefinition of fuzzy indications is required by using all the neighboring local regimes. As fuzzy weights of different regimes can be considered as a distance measure from the current operating point to the central point of the obtained regimes, the fuzzy weights can consequently be defined as the distances in the operating window. For the neighboring regimes Z_0, Z_1 , and Z_3 of the operating point Z' , the distances d_0, d_1, d_3 from Z_0, Z_1 , and Z_3 to Z' are given as

$$\rho_{Z_0} = \frac{d_0}{d_0 + d_1 + d_3}$$

$$\rho_{Z_1} = \frac{d_1}{d_0 + d_1 + d_3}$$

$$\rho_{Z_3} = \frac{d_3}{d_0 + d_1 + d_3}$$

and the interpolation values can be easily determined as

$$w_{Z_0} = \frac{\rho_{Z_0}}{\rho_{Z_0} + \rho_{Z_1} + \rho_{Z_3}}$$

$$w_{Z_1} = \frac{\rho_{Z_1}}{\rho_{Z_0} + \rho_{Z_1} + \rho_{Z_3}}$$

$$w_{Z_3} = \frac{\rho_{Z_3}}{\rho_{Z_0} + \rho_{Z_1} + \rho_{Z_3}}$$

respectively. The same procedure can be applied to obtain the global controller output by the calculating of Equation (5.3) based on the local controllers of regimes Z_0 , Z_1 , and Z_3 .

By using the above procedure, any operating points in the operating window can be described by using the neighborhood regimes and the global controller can consequently combined by using the involved local ones.

5.6 Overall Structure of Multi-model Control Strategy

The central part of the proposed multi-model structure is the supervisor, which coordinate with other components to form an overall multi-model strategy. The procedures for determining the multi-model structure is given in the following algorithm:

Algorithm 5.1

Step 1 Define the sampling frequency and the time horizon P , collect the process output $y(k)$, and calculate the global system output of combined local models $\tilde{y}(k)$ by Equation (5.4). Calculate error cost function $J_e(k)$ and $\tilde{J}_e(k)$ by Equation (5.5) and (5.6).

Step 2 Calculate the performance index $\gamma(k)$ by Equation (5.7) and formulate the prediction residual $\Delta\gamma(k)$ using Equation (5.9). Utilize statistical process monitoring tools by defining ± 3 standard deviation (3σ limits) to monitor the controlled process. If statistically significant changes are detected, go to next step (*Step 3*).

Step 3 Add an additional regime (model and controllers) into the multi-model structure, and reschedule existing distribution together with the added operating regime on the axis of the operating window by the procedure given in Section 5.4.

Step 4 Determine the updated interpolation values for different regimes based on the operating point for the two situations presented in Section 5.5, and obtained the overall controller output. Go to Step 1.

By combining the methods proposed in Chapters 2, 3, 4, and 5, the multi-model structure integrates the methods of performance monitoring and supervising function with online identification and controller parameter tuning. The global structure of multi-model approach is shown as in Fig. 5.8, which can be classified as two layers, three modules: the lower layer consists of parameter identification and controller tuning modules and higher layer for monitoring and supervision module. The performance monitor observes the controller performance online and informs the identification module when an unsatisfactory performance occurs due to a model mismatch. The controller tuning module is triggered by the identification module to perform the local controller tuning task. Both identified local models and designed controller will be added to the multi-model structure in supervision layer. According to the regime division method and the model validity function, the supervision module then redistribute regimes and fuzzy weights. Such a procedure including identification, multi-model formulation and controller designing can be

adaptively performed online. Therefore, the proper multi-model division in the whole operating range is realized.

Fig. 5.8 Global structure of multi-model control

Combining the local model identification, local controller tuning, and the multi-model construction method, a systematic strategy for nonlinear MIMO process control system is presented as following procedures:

1. A decentralized control system is monitored through the multi-model controller performance assessment procedures given in Section 5.4. If poor performance is detected, go to 2. Otherwise, go to 6.
2. Activate parameter identification module and using the identification

procedure provided in Chapter 2 to identify the parameters of additional local model.

3. Pair the loops based on the ERGA analysis and tune the decentralized PID controllers using detuning factor method given in Chapter 3. If fine tuning is required, go to 4. Otherwise, go to 5.

4. Fine-tune the PID parameters by employing multi-step receding horizon optimization method given in Chapter 4.

5. Redistribute the premise variables of existing and newly added regime fuzzy sets for different situations discussed in Section 5.4.

6. Calculate the controller output as discussed in Section 5.5 and go to 1.

Chapter 6

Applications in HVAC Process Control Systems

6.1 Introduction

In modern society, air-conditioning has become an indispensable part of our life. Different kinds of HVAC systems have been installed in housing estates, commercial complexes, office buildings, and industrial estates, etc. These systems are playing a crucial role in providing thermal conditions for people inside the air-conditioned space. With the wide uses of air-conditioning system, the electricity usage for space air-conditioning amounts up to 30% of the total electricity energy consumption for countries with mild climate [113, 114]. As Singapore is a tropical

city country with an annual average ambient temperature of 29.4°C and an annual average ambient relative humidity of 85%, the indoor thermal condition of buildings is very important since people live and work in buildings with air-conditioning systems running all the time. Consequently, electricity usage in building air-conditioning and ventilation systems in Singapore is very high, which amounts to more than 50% of total electric energy produced. Therefore, it is very important to increase the system efficiency and reduce the energy consumption. The research in the area of energy efficiency for air-conditioning system has, therefore, a great impact to the nation's economy.

As shown in Fig. 6.1, the HVAC system consists of indoor air loop, chilled water loop, refrigerant loop, condenser water loop, and outdoor air loop. Each component involves several variables, the working principle and system analysis can be very complicated. To minimize the energy consumption while meeting the human comfort requirement, an accurate mathematical model for each component is needed to describe its working conditions. Based on the models, optimization problem for buildings can be formulated as minimization of the energy cost with the constraints that all the requirements on air quality in the building such as room temperatures, humidities, air movement and oxygen concentrations be met. While for every set-point change determined by the optimization algorithm, the control system is to make sure that each device can reach its new set-point smoothly and promptly.

During the years, different control strategies have been developed to control fully or partially of the HVAC process operation [115-117]. However, classical control techniques, such as on-off switching controller (thermostats) and PID controllers are still widely used in practice due to easier implementation, low cost and reliable performance etc. Recently, some complex control strategies based on

the classical control concepts have been proposed in attempting to improve the system performance. Among those works, T. I. Salsbury [118] described a feedforward controller as a supplement of the conventional PI feedback controller based on simplified physical models. M. Kasahara and T. Matsuba *et al.* [119] proposed a robust PID control scheme to deal with the model uncertainty caused by changing characteristics of the plant. Bi and Cai *et al.* [61] developed an advanced auto-tuning PID controller for both temperature and pressure control loops. However, in the long run, these controllers may not provide satisfactory performance since they are designed for specified cooling loads while the actual cooling loads affecting the system are time varying. To cope with strong nonlinearity and time varying characteristics caused by varying cooling load, a multi-model control approach for single loop temperature control was proposed in [120], which was based on a hierarchical two-level structure. The proposed multi-model method has demonstrated its effectiveness for system performance at different operating points.

Fig. 6.1 Schematic of a typical building HVAC system

Since there are many loops involving multiple variables to be controlled in HVAC systems, which poses a great challenge for control engineers. In the previous chapters, we have discussed the design issues for multi-model control strategy for nonlinear MIMO complex processes. We shall combine all these procedures to

solve the real world HVAC control problem in this chapter to cope with nonlinearity and time varying characteristics caused by varying cooling load. The application results have demonstrated the proposed method results in much better control performance compared with existing methods.

6.2 Single-zone HVAC Control System

System components and description

The single-zone HVAC system described in [121] is shown in Fig. 6.2. It consists of the following components: a cooling coil component, a conditioned space (single-zone), a control valve, control dampers, the supply and return air fans. The basic operating principle of the system can be described as follows:

1. 25% outside air is allowed into the system and get mixed with 75% of the return air to form the mixed air.
2. The mixed air passes through the cooling coil, where the air gets conditioned.
3. The conditioned air coming out of the cooling coil, which is called supply air, enters into the conditioned space.
4. The supply air enters the conditioned space to offset the cooling load acting upon the system to achieve the desired room temperature and relative humidity.
5. Finally, the air in the conditioned space is drawn through return fan, and 75% of return fan is circulated and the rest is exhausted from the system.

In the derivation of the model, the following assumptions were made: 1) ideal gas behavior; 2) perfect mixing; 3) constant pressure process; 4) negligible wall and

thermal storage; 5) negligible infiltration and exfiltration effects; and 6) negligible transient effects in the flow-splitter and mixer.

Fig. 6.2 Structure of the single-zone HVAC system

Based on the assumptions, the differential equations describing the dynamics of the system can be derived from energy conservation principles and are given by

$$\begin{cases} \frac{dT_z}{dt} = \alpha_1 f(T_s - T_z) - \alpha_2 f(W_s - W_z) + \alpha_3 (Q_z - h_{fg} M_z) \\ \frac{dW_z}{dt} = \alpha_1 f(W_s - W_z) + \alpha_4 M_z \\ \frac{dT_s}{dt} = \beta_1 f(T_z - T_s) - \beta_2 f(T_0 - T_s) - \\ \beta_3 f(0.25W_0 + 0.75W_z - W_s) - \beta_4 gpm \end{cases} \quad (6.1)$$

where

$$\alpha_1 = 60/V_z, \quad \alpha_2 = 60h_{fg}/(c_p V_z), \quad \alpha_3 = 1/(0.25c_p V_z \rho), \quad \alpha_4 = 1/(\rho V_z),$$

$$\beta_1 = 60/V_c, \quad \beta_2 = 60 \cdot 0.25/V_c, \quad \beta_3 = 60h_w/(c_p V_c), \quad \beta_4 = 6000/(c_p V_c \rho)$$

h_w Enthalpy of liquid water

W_o Humidity ratio of outside air

- h_{fg} Enthalpy of water vapor
 V_c Volume of cooling coil
 W_s Absolute humidity of supply air
 W_z Absolute humidity of conditioned space
 c_p Specific heat of air
 T_o Temperature of outdoor air
 M_z Moisture load
 Q_z Sensible heat load
 T_s Temperature of supply air
 T_z Temperature of conditioned space
 V_z Volume of conditioned space
 ρ Air mass density
 f Volumetric flow rate of air
 gpm Chilled water flow rate

The dynamics of temperature and absolute humidity in conditioned space is shown in Equation (6.1). However in real application, relative humidity which indicates the amount of moisture in percentage is usually used for HVAC systems. An empirical relation between absolute and relative humidity is given by the following approximate equation based on data from the Psychrometric chart [122]

$$\Phi(\%) = 5000W - 1.388T + 107$$

where $\Phi(\%)$ is the relative humidity and T is the dry bulb temperature in $^{\circ}F$, and the above approximation is valid around operating point of $T = 75^{\circ}F$ and $W = 0.009lb/lb$. Consider the derivation of relative humidity

$$\frac{d\Phi(\%)}{dt} = 5000 \frac{dW}{dt} - 1.388 \frac{dT}{dt} \quad (6.2)$$

Combining Equation (6.1) and Equation (6.2), results in the following state space representation

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} f_1(x) \\ 0 \\ f_3(x) \end{bmatrix} + \begin{bmatrix} g_{11}(x) \\ g_{21}(x) \\ g_{31}(x) \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ g_{22} \\ 0 \end{bmatrix} u_2 \quad (6.3)$$

where $x_1 = \Phi(\%)$, $x_2 = T_s$, $x_3 = T_z$, $u_1 = f$, $u_2 = gpm$ and

$$f_1(x) = 5000\alpha_4 M_z - 1.388\alpha_3 (Q_z - h_{fg} M_z)$$

$$f_3(x) = \alpha_3 (Q_z - h_{fg} M_z)$$

$$g_{11}(x) = 5000\alpha_1 W_s + 107\alpha_1 + 0.03\alpha_2 + 1.388\alpha_2 W_s + (-\alpha_1 - 2.8 \times 10^{-4} \alpha_2) x_1 - 1.388\alpha_1 x_2 - 3.85 \times 10^{-4} \alpha_2 x_3$$

$$g_{12}(x) = -1.5 \times 10^{-4} \beta_3 x_1 - \beta_1 x_2 + (\beta_1 - \beta_2 - 2.1 \times 10^{-4} \beta_3) x_3 + \beta_2 T_0 - 0.25 \beta_3 W + \beta_3 W_s + 0.016 \beta_3$$

$$g_{13}(x) = (\alpha_2 (-W_s - 0.0214)) + 2 \times 10^{-4} \alpha_2 x_1 + \alpha_1 x_2 + (2.8 \times 10^{-4} \alpha_2 - \alpha_1) x_3$$

$$g_{22}(x) = -\beta_4$$

Single model control

The system parameters for the single-zone HVAC system are shown in Table 6.1 [121, 122].

Table 6.1 System parameters for single-zone application

Parameter	Nominal value	Parameter	Nominal value
ρ	0.074 lb/ft^3	H_w	340 Btu/lb
c_p	$0.24 \text{ Btu/lb}^\circ\text{F}$	M_z	166 lb/hr
V_c	60.75 ft^3	Q_z	290000 Btu/hr
V	58464 ft^3	W_s	0.007 lb/lb

H_{fg}	1075Btu/lb	W_0	0.018lb/lb
f	17000cfm	gpm	60gpm
T	75°F		

The main object is to maintain the conditioned space temperature and relative humidity at the set-points values for comfortable condition. The manipulated inputs for the system are the flow rate of chilled water from the chiller to coil and the circulating air flow rate supplied by the variable speed fan. From Equation (6.3), the system is obviously nonlinear, and the cooling load is a time varying variable affecting the system dynamics. The multivariable process can be presented as $y = Gu$, where, G is the identified transfer matrix

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_3 \\ x_1 \end{bmatrix}, \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

In HVAC system, cooling load Q is a time varying variable which may cause poor performance when Q drifts from the initial value Q_0 . For the cooling load of $Q_0 = 290000 \text{ Btu/hr}$, the model parameters and ERGA values for the process are obtained as

$$G_0 = \begin{bmatrix} \frac{-0.015}{0.29s+1} e^{-0.1s} & \frac{-3.6}{0.33s+1} e^{-0.2s} \\ \frac{0.011}{0.27s+1} e^{-0.1s} & \frac{4.1}{0.33s+1} e^{-0.2s} \end{bmatrix}$$

and

$$\Phi_{ERGA}(Z_0) = \begin{bmatrix} 2.84 & -1.84 \\ -1.84 & 2.84 \end{bmatrix}$$

respectively. The results imply the diagonal pairing 1-1/2-2 is the right pairing choice. According to the method proposed in Chapter 3, a set of PI controller can be calculated based on ERGA index

Z_0 : Loop1: $K_C = -78, K_I = -283$
 Loop2: $K_C = 0.22, K_I = 0.34$

Simulation is performed as follows, set initial set-point values for the conditioned space temperature and the relative humidity at $73^\circ F$ and 45%, respectively, change set-point values at $t = 0.5, t = 1$ (hours), where the condition space temperature changes to $77^\circ F$ and the relative humidity changes to 40% for initial cooling load Q_0 . Additional set-point values are changed at $t = 1.5, t = 2$ and 2.5 (hours) for a changed cooling load $Q_1 = 250000 Btu/hr$. The corresponding closed-loop response for different operating conditions is shown in Fig. 6.3 with the corresponding manipulated variables; air flow rate in volume and chilled water flow rate in *gpm* are shown as in Fig. 6.4.

Fig. 6.3 Closed-loop responses of single-zone application

Fig. 6.4 Manipulated variables of single-zone application

Multi-model control results

It can be easily seen that the designed controller for the initial operating condition can provide acceptable performance. However, once the operating condition experiences a relatively big change, it no longer provides satisfactory closed-loop performance. Through the statistical analysis of the performance measure $\gamma(k)$ and the prediction residual $\Delta\gamma(k)$, an additional local model is needed for the process at the changed operating point. Consequently, the supervisor in the higher level notify identification module to identify additional local models. The addition local model, ERGA and controller for operating regime Z_1 are obtained, respectively, as

$$G_1 = \begin{bmatrix} \frac{-0.028}{0.58s+1}e^{-0.2s} & \frac{-5.4}{0.71s+1}e^{-0.3s} \\ \frac{0.021}{0.55s+1}e^{-0.2s} & \frac{6.8}{0.73s+1}e^{-0.3s} \end{bmatrix}$$

$$\Phi_{\text{ERGA}}(Z_1) = \begin{bmatrix} 2.5 & -1.5 \\ -1.5 & 2.5 \end{bmatrix}$$

and

$$\begin{array}{ll} Z_1: & \text{Loop1: } K_C = -45, K_I = -78 \\ & \text{Loop2: } K_C = 0.15, K_I = 0.19 \end{array}$$

According to the operating regime division method proposed in Chapter 5, together with the initially obtained operating regime Z_0 , two local operating regimes are constructed to form the whole multi-model structure. The operating regime division is based on the two manipulated variables and the corresponding operating regimes distribution is shown as in Fig. 6.5.

Fig. 6.5 Operating regimes distribution of single-zone application

The resulting closed-loop responses and the corresponding manipulated outputs are shown in Fig. 6.6 and Fig. 6.7 respectively. Compare with Fig. 6.3, it is clear that a better performance is achieved through the proposed multi-model approach.

Fig. 6.6 Multi-model approach performance of single-zone application

Fig. 6.7 Multi-model manipulated variables of single-zone application

However it is noted from Fig. 6.6 that the control system responses appear sluggish. To improve the performance, the fine tuning method through multi-step horizon optimization proposed in Chapter 4 is applied, and the resulting closed-loop responses are shown as in Fig. 6.8.

Fig. 6.8 Fine-tuning performance for single-zone application

To further demonstrate the effectiveness of the multi-model strategy, now consider the situation where the cooling load is drifted from the initial condition Q_0 to $Q_2 = 270000 \text{ Btu/hr}$. Since this operating point is located inside the boundary enclosed by operating regimes Z_0 and Z_1 , the interpolation value of the two operating regimes are easily obtained as 0.55 and 0.45 according to Chapter 5. The simulation results under three cooling load conditions are shown as in Fig. 6.9, where

$$t = 0 - 0.5, Q = Q_0; t = 0.5 - 1, Q = Q_2; t = 1 - 1.5; Q = Q_1.$$

Fig. 6.9 Performance for different cooling load conditions of single-zone application

6.3 Two-zone Air Temperature Control System

System components and description

Here we consider the temperature control problem of a multi-zone fan-coil process in HVAC system. The schematic diagram of the two-zone system is shown as in Fig. 6.10.

In this process, two temperatures are controlled by regulating the position of two Variable Air Volume (VAV) dampers. When the VAV damper opens wider, more cooling air enters the room and the room temperature will drop, and vice versa.

Fig. 6.10 Schematic diagram of two-zone system

In the room of the pilot plant, there are two temperature sensors, T_{z1} and T_{z2} located as zone 1 and zone 2 respectively. There are two control dampers DM_1 and DM_2 on the room ceilings of zone 1 and zone 2, respectively. The change of any damper position will cause the change of both T_{z1} and T_{z2} . Thus, it forms a coupled TITO process. The main control objective is to maintain the conditioned spaces temperatures T_{z1} and T_{z2} at the set-point values. The manipulated inputs for the system are the positions of the two dampers DM_1 and DM_2 .

The dynamics of the two zone fan coil system can be modeled as followings [123]

$$\begin{cases} C_{z1} \frac{dT_{z1}}{dt} = u_1 \xi_1 (T_s - T_{z1}) - a_{z1} (T_{z1} - T_o) - a_{z12} (T_{z1} - T_{z2}) + hA_{d1} (T_{d11} - T_{z1}) \\ C_{z2} \frac{dT_{z2}}{dt} = u_2 \xi_2 (T_s - T_{z2}) - a_{z2} (T_{z2} - T_o) - a_{z12} (T_{z1} - T_{z2}) + hA_{d2} (T_{d12} - T_{z2}) \end{cases}$$

where u_1 and u_2 are the controller output through the two dampers, a_{z1} and a_{z2} are the heat loss coefficient of zone 1 and zone 2, a_{z12} is the heat loss coefficient between the two zones, ξ_1 and ξ_2 are the heat transfer coefficients, C_{z1} and C_{z2} are the thermal capacity of air, T_s is the supply air, which is controlled as a constant

value, T_o is the outside temperature, $hA_{d1}(T_{d11} - T_{z1})$ and $hA_{d2}(T_{d12} - T_{z2})$ are two terms representing how the building enclosure (elements such as walls, floor etc.) stores and releases energy to the indoor environment. It is obviously a nonlinear system with time varying characters.

A robust adaptive controller was introduced in [123] and an auto-tuning two zone temperature control application was illustrated in [61]. However, the method used in [123] is not practical in real application, as the complete nonlinear model is very difficult to model accurately. On the other hand, the practical method proposed in [61] also has its constraints, i.e. the approximate model has its operating validity, as the nonlinear dynamics changes for different operating conditions.

Multi-model control application and results

The two-zone air temperature control application for HVAC systems is tested in a pilot centralized HVAC, shown in Fig. 6.11. The system has three chillers, three zones with three Air Handling Units (AHUs), three cooling towers and flexible partitions up to twelve rooms. All motors (fans, pumps and compressors) are equipped with Variable Speed Drive (VSD). The system is made very flexible to configure these three units to form different schemes.

Fig. 6.11 Pilot plant of centralized HVAC system

The experiment is conducted under the following conditions: The chilled water supply temperature is fixed; the cooling load variation is achieved through the air and water flow rates. In VAV system, the supply air temperature is controlled at a fixed set-point value. The supply fan is varied according to the time varying cooling load. A hierarchical two-level multi-model control system is applied for the supply air temperature [120].

During the test, the air flow rate, which in VAV system reflects the time varying cooling load, is chosen as the scheduled variable. According to the operating regime division method proposed in Chapter 5, two local operating regimes are constructed to form the whole model. Based on model validity function

for each operating regime, the operating regimes distribution is shown in Fig. 6.12.

Fig. 6.12 Operating regimes distribution of two-zone application

The local model for operating regimes Z_0 and Z_1 are obtained as following transfer matrix

$$G_0 = \begin{bmatrix} \frac{-0.051}{132s+1} e^{-28s} & \frac{-0.031}{116s+1} e^{-53s} \\ \frac{-0.029}{143s+1} e^{-25s} & \frac{-0.053}{117s+1} e^{-21s} \end{bmatrix}$$

and

$$G_1 = \begin{bmatrix} \frac{-0.049}{122s+1} e^{-23s} & \frac{-0.025}{109s+1} e^{-43s} \\ \frac{-0.023}{137s+1} e^{-18s} & \frac{-0.047}{108s+1} e^{-14s} \end{bmatrix}$$

respectively. It is easy to obtain the ERGA for Z_0 and Z_1 as follows

$$\Phi_{ERGA}(Z_0) = \begin{bmatrix} 1.19 & -0.19 \\ -0.19 & 1.19 \end{bmatrix} \text{ and } \Phi_{ERGA}(Z_1) = \begin{bmatrix} 1.12 & -0.12 \\ -0.12 & 1.12 \end{bmatrix}$$

The results imply the diagonal pairing 1-1/2-2 is the right pairing choice for both local regimes. And from ERGA, the parameters of controller from Z-N method are adjusted by the detuning factor as follows

Z_0 :	Loop1:	$K_C = 44.2, K_I = 0.5$
	Loop2:	$K_C = 49.6, K_I = 0.8$
Z_1 :	Loop1:	$K_C = 53.9, K_I = 0.8$
	Loop2:	$K_C = 81.4, K_I = 1.9$

To verify the effectiveness of the proposed multi-model approach, the Luyben's PI controller setting is applied for comparison. According to Luyben's method, PI controller setting for Z_0 can be obtained as $K_C = 45.1, K_I = 0.4$ for Loop 1 and $K_C = 50.6, K_I = 0.6$ for Loop 2. Figs. 6.13, and 6.14 show the performance of Luyben's method, in which the operating condition changes from Z_0 to Z_1 at $t = 23$, and the set-point values change at $t = 10$ and 40 . Figs. 6.15 and 6.16 show the performance of proposed multi-model approach under the same condition.

Fig. 6.13 Closed-loop performance of Luyben's method

Fig. 6.14 Manipulated variables of Luyben's method

Fig. 6.15 Closed-loop performance of multi-model method

Fig. 6.16 Manipulated variables of multi-model method

From the above real-time experiments, it is obvious that the Luyben's method for one operating condition cannot provide the satisfactory performance for the changed operating conditions. Whereas the proposed multi-model control scheme works well in the two-zone HVAC application, with less settling time for the changed operating conditions. With this control scheme, a good control performance can be expected even though a process is heavily nonlinear, which is common in industry. Since many processes in other industries have similar characteristics as those in HVAC systems, in spite of the fact that the control scheme was extensively tested only in HVAC processes, with suitable modifications, the developed control method can be applied in various other industrial areas.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

In this thesis, a systematic multi-model control strategy for complex nonlinear MIMO processes has been proposed and its applications to HVAC systems have been investigated. Four important issues related to constructing the proposed multi-model control system have been extensively discussed, namely, parameter identification of local models, design of decentralized local controllers, partition of operating regimes and determination of overall control outputs. The main contributions of this thesis are summarized as follows.

1. For local model identification, a novel identification method based on step tests for TITO process was proposed. The closed-loop MIMO control system was

decoupled into multiple single open-loop systems with same input signal acting on each transfer function. To effectively determine the critical frequency of each transfer function, an iterative Min-Max CFS searching method was developed. By using an iterative algorithm, the transfer functions with higher critical frequencies can make use of the results of transfer functions with lower critical frequencies, which could greatly reduce computational burden. The proposed method requires no prior knowledge of the process and the first-order or second-order plus dead time model elements of the transfer function matrices can be obtained directly which are very suitable for the construction of local models for the multi-model control approach. The main advantages of the proposed methods are: 1) the testing procedure is straightforward, the computation is simple and can be easily implemented; 2) it is valid for closed-loop tests; and 3) it can identify both close-coupled and weak-coupled processes. Various processes have been employed to demonstrate the effectiveness and practicability of the identification method. Simulation results show that the proposed identification method is practical and accurate even in a noisy environment, especially in the frequency range from zero to the frequency corresponding to π phase lag. Even though the method is derived for TITO systems, the extension to general MIMO system is straightforward. The obtained models are adequate and ready for tuning of PID and advanced controllers.

2. For local decentralized controller design, a new criterion for loop pairing in terms of ERGA was proposed. The proposed ERGA criterion combines the advantages of RGA and DRGA by using both steady state gain and the bandwidth of each process transfer function matrix element. It is a practical method with less computation and can provide a comprehensive description of dynamic interactions. Consequently, a decentralized controller tuning formula by using SISO tuning methods plus detuning factor evaluated by ERGA was proposed. The tuning method is very simple to be understood by field engineers, yet very effective to find suitable

controller parameters. To further improve the control system performance, a two-level decentralized PID controller tuning technique based on receding horizon optimization was developed to regulate the original controller parameters. A realizable reference was introduced into the MPC-like cost function to reflect the decentralized control structure limitations. As the proposed method does not require change of the existing hardware and the rules of each controller parameter, it can be well accepted in practice. Examples on typical TITO processes demonstrated the effectiveness and practicability of the proposed method.

3. For construction of multi-model control structure, an increment strategy of adding local regimes based on the evaluation of controller performance was proposed. Fuzzy integration of local regimes was employed to form the global multi-model controller. Additional local regimes will be added and corresponding boundaries of fuzzy functions for partitioning will be redistributed only when the unsatisfactory controller performance caused by model mismatch is detected. The membership function of local regimes is adjusted according to the situations of additional local regime interpolation. From the global point of view, a supervisor was defined to coordinate the whole system, including the performance monitoring, local model identification, local controller tuning and multi-model control construction. It is a systematic structure and suitable for the complex nonlinear process control system, which is simple, without requiring prior process knowledge, and can be fully automated.

At last, the proposed decentralized MIMO control system designing strategy was applied to control of HVAC processes. Both application results demonstrated the effectiveness and practicality of the proposed strategy.

7.2 Recommendations

In this thesis, we have proposed an engineering oriented control method based on multi-model approach for complex nonlinear MIMO processes. As the multi-model control method covers many topics including process modeling, controller implementation, and multiple regimes construction. Each topic poses a special challenge for control engineers. In this thesis, the research work was focused on the practical implementation issues regarding decentralized TITO processes, and does not cover all the aspects of these research topics. In closing this thesis, some problems that are important for future research on decentralized control and multi-model approach are briefly described below.

- TITO as typical MIMO processes were exclusively considered in this thesis, all the proposed methods and examples were based on TITO systems. Although the methods can be extended to the general MIMO processes, there exist some challenges such as the complex interactions analysis, the multi-model division strategy, etc. For a general MIMO process, the interactions will not be easily handled as in TITO processes. The direct controller tuning based on single loop tuning methods and evaluation of ERGA index are no longer valid, less conservative detuning methods are yet to be developed. Similarly, the operating point will be in a higher dimensional operating space, which results in the complex operating regime division and the integration of local regimes.
- In order to further improve the decentralized control system performance, the multi-step horizon optimization was applied in the thesis. Although the proposed method keeps the widely applied PID controller structure and the fine tuning is stopped at stable closed-loop conditions, it still has its limitations in situations where the online controller parameters adjustments are not permitted.

Develop a controller with fixed parameters which can provide an optimization performance in the sense of minimizing multi-step horizon cost function would be a possible future study area.

- For a complex nonlinear MIMO process, there may be many local regimes. In such a situation, the determination of the right number of local regime and the combination of the local regimes become a complex problem. The multi-model approach will lose its advantages that solve the complex problem by dividing it into simpler ones. Even though the incremental strategy can effectively alleviate this problem, the redistribution of the validity functions, when the additional regimes are added may cause some uncertainties. How to reschedule the operating regimes so that each regime can has its maximum validity regime would be a future study topic.
- At last, a recommended future work is to incorporate the proposed method into an easily implemented toolbox. As one main objection of this thesis is to solve the complex problem in practical way, and several engineering oriented algorithms has been presented, consequently it is of practical importance to develop a fully integrated software tool box with friendly interface for control applications.

Author's Publications

1. M. He, Wen-Jian Cai and Shao-Yuan Li, "Multiple fuzzy model-based temperature predictive control for HVAC systems," *Information Sciences*, vol. 169(1-2), pp. 155-174, 2005.
2. M. He, Wen-Jian Cai, Shao-Yuan Li and Q. Xiong, "Decentralized parameter identification in frequency domain for TITO process from closed-loop step responses," Submitted to *Computers & Chemical Engineering*.
3. M. He, Wen-Jian Cai, Q. Xiong and Mao-Jun He, "Decentralized multi-model control of multivariable thermal processes in HVAC systems," Submitted to *Energy Conversion and Management*.
4. Q. Xiong, Wen-Jian Cai and M. He, "A practical decentralized PID auto-tuning method for TITO systems under closed-loop control," *International Journal of Innovative Computing, Information and Control*, vol. 2(2), pp.1-18, 2006.
5. Q. Xiong, Wen-Jian Cai, Mao-Jun He and M. He, "Decentralized controller design based on effective relative gain array," Submitted to *Journal of Process Control*.
6. Mao-Jun He, Wen-Jian Cai, B. F. Wu and M. He, "A simple decentralized PID controller design method based on dynamic relative interaction analysis," Submitted to *Ind. Eng. Chem. Res.*

Bibliography

- [1] H. Nyquist, "Regeneration Theory," *Bell Syst. Tech. J.*, 1932.
- [2] B. W. Bequette, "Nonlinear control of chemical processes: A review," *Ind. Eng. Chem. Res.*, vol.30, pp.1391-1413, 1991.
- [3] B. A. Ogunnaike, W. H. Ray, *Process Dynamics, Modeling and Control*. Oxford University Press, 1994.
- [4] K. J. Astrom, and T. Hagglund, *PID Controllers: Theory, Design, and Tuning (2nd edition)*. Instrument Society of America. Research Triangle Park, NC, 1995.
- [5] W. J. Rugh, J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, pp. 1401-1425, 2000.
- [6] G. Continillo, *Nonlinear Dynamics and Control in Process Engineering: Recent Advances*, Springer, 2001.
- [7] A. Isidori, *Nonlinear Control System*. Springer Verlag, 1995.
- [8] C. Kravaris and J. C. Kantor, "Geometric methods for nonlinear process control, I and II," *Ind. Eng. Chem. Res.*, vol. 29, pp. 2295-2323, 1990.
- [9] P. L. Lee, and G. R. Sullivan, "Generic model control (GMC)," *Comput. Chem. Eng.*, vol. 12, pp. 578-580, 1988.
- [10] M. A. Henson and D. E. Seborg, "Nonlinear control strategies for continuous fomenters," *Proceedings of American Control Conference*, San Diego, pp.2723-2728, 1990.

- [11] J. C. Kantor, "Nonlinear sliding-mode controller and objective function for surge tanks," *Int. J. Control*, vol. 50, pp. 2025-2047, 1989.
- [12] J. Baillieul and J. C. Willems (Eds.), *Mathematical Control Theory*. Springer Verlag, New York, 1999.
- [13] D. Q. Mayne, "Nonlinear model predictive control: An assessment," *Proc. of 5th International Conference on Chemical Process Control*, pp. 217-231, 1997.
- [14] A. A. Patwardhan, J. B. Rawlings, and T. F. Edgar, "Nonlinear model predictive control," *Chem. Eng. Commun.*, vol. 87, 123-141, 1990.
- [15] G. F. Page, J. B. Gomm, and D. Williams (Eds), *Application of Neural Networks to Modeling and Control*. Chapman & Hall, 1993.
- [16] K. J. Hunt, *et al.*, "Neural networks for control systems – A survey," *Automatica*, vol. 28, pp.1083-1112, 1992.
- [17] L. A. Zadeh, "Fuzzy sets," *Information Control*, vol. 8(2), pp. 338-358, 1965.
- [18] E. H. Mamdani, "Application of fuzzy algorithms for simple dynamic plant," *Proc. IEE*, 121, pp. 1585-1588, 1974.
- [19] L. X. Wang, "Fuzzy systems are universal approximations," *Proc. IEEE Conference on Fuzzy Systems*, pp.1163-1170, 1992.
- [20] S. Miltra, Y. Hayashi, "Neuro-fuzzy rule generation: survey in soft computing framework," *IEEE Trans on Neural Networks*, vol.11, pp. 748-768, 2000.
- [21] J. Vieira, F. M. Dias and A. Motac, "Artificial neural networks and neuro-fuzzy systems for modeling and controlling real systems: A comparative study," *Engineering Applications of Artificial Intelligence*, vol. 17, pp. 265-273, 2004.
- [22] D. Nauck, F. Klawonn, R. Kruse, *Foundations of Neuro-Fuzzy Systems*. Wiley, New York, 1997.
- [23] J. Jang, "ANFIS: Adaptive network based fuzzy inference systems," *IEEE Trans on SMC*, vol.23, pp.665-685, 1993.

- [24] R. Murray Smith and T. A. Johansen, *Multiple Model Approaches to Nonlinear Modeling and Control*. Taylor & Francis, 1997.
- [25] K. S. Narendra, and J. Balakrishnan, "Adaptive control using multiple models," *IEEE Trans. AC*, vol. 42 (2), pp. 171-187, 1997.
- [26] J. A. Rodriguez, J. A. Romagnoli, G. C. Goodwin, "Supervisory multiple regime control," *Journal of Process Control*, vol. 13, pp.177-191, 2003.
- [27] A. N. Venkat, P. Vijaysai, R. D. Gudi, "Identification of complex nonlinear process based on fuzzy decomposition of the steady state space," *Journal of Process Control*, vol. 13, pp.473-488, 2003.
- [28] T. H. Gottsche, K. J. Hunt, T. A. Johansen, "Nonlinear dynamics modeling via operating regime decomposition," *Mathematics and Computers in Simulation*, vol. 46, pp.543-550, 1998.
- [29] T. A. Johansen, B. A. Foss, "Identification of nonlinear system structure and parameters using regime decomposition," *Automatica*, vol. 31(2), pp. 321-326, 1995.
- [30] B. Aufderheide and B. W. Bequette, "Extension of dynamic matrix control to multiple models," *Computers & Chemical Engineering*, vol. 27, pp. 1079-1096, 2003.
- [31] D. Dougherty and D. Cooper, "A practical multiple model adaptive strategy for single-loop MPC," *Control Engineering Practice*, vol.11 (2), pp. 141-159, 2003.
- [32] D. Dougherty and D. Cooper, "A practical multiple model adaptive strategy for multivariable model predictive control," *Control Engineering Practice*, vol.11 (6), pp. 649-664, 2003.
- [33] T. A. Johansen, *Operating Regime based Process Modeling and Identification*. Ph.D thesis, Department of Engineering Cybernetics, Norwegian Institute of Technology, Norway, 1994.
- [34] S. Skogestad, *Multivariable Feedback Control – Analysis and Design*. John Wiley& Sons, 1996.

- [35] J. C. Doyle and G. Stein, "Multivariable feedback design: Concepts for a classical/modern synthesis," *IEEE Trans. AC*, vol. 26(1), pp.4-16, 1981.
- [36] D. E. Seborg, T. F. Edgar, and D. A. Mellichamp, *Process Dynamics and Control* (Second Edition). John Wiley & Sons, 2003.
- [37] H. H. Rosenbrock, *Computer-aided Control System Design*. New York: Academic Press, 1974.
- [38] A. G. J. Macfarlane, and B. Kouvaritakis, "A design technique for linear multivariable systems," *Int. J. Control*, vol. 25, 837-874, 1977.
- [39] J. M. Maciejowski, *Multivariable Feedback Design*. Workingham, U.K.: Addison-Wesley, 1989.
- [40] W. L. Luyben, *Process Modeling, Simulation and Control for Chemical Engineers*. McGraw-Hill, New York, 1990.
- [41] P. C. Young, "An instrumental variable method for real time identification of a noisy process," *Automatica*, vol. 6, pp.271-287, 1970.
- [42] Q. G. Wang, X. Guo, and Y. Zhang, "Direct identification of continuous time delay systems from step responses," *J. of Process Control*, vol. 11, pp.531-542, 2001.
- [43] W. L. Luyben, "Derivation of transfer functions for highly nonlinear distillation columns," *Ind. Eng. Chem. Res.*, vol. 26, pp. 2490-2595, 1987.
- [44] T. Hagglund, *Process Control in Practice*. Chartwell-Bratt Ltd, Bromley, UK, 1991.
- [45] Y. G. Wang, and H. H. Shao, "PID autotuner based on gain and phase margin specifications," *Ind. Eng. Chem. Res.*, vol. 38, pp. 3007-3012, 1999.
- [46] Y. G. Wang, and W. J. Cai, "Decentralized relay-based multivariable process identification in frequency domain," *IEEE Trans. AC*, vol.48, pp.873-877, 2003.
- [47] A. P. Loh, C. C. Hang, C. K. Quek and V. U. Vasnani, "Autotuning of multiloop proportional-integral controllers using relay feedback," *Ind. Eng. Chem. Res.*, vol. 32, pp. 1102-1107, 1993.

- [48] A. P. Loh, V U. Vasnani, "Necessary conditions for limit cycles in multi-loop relay systems," *IEE Proc.-D*, vol. 141, pp.163-168, 1994.
- [49] K. J. Astrom, and T. Hagglund, *Automatic Tuning of PID Controllers*. Research Triangle Park, NC: Instrum. Soc. Amer, 1988.
- [50] Q. G. Wang and Y. Zhang, "Robust identification of continuous systems with dead-time from step responses," *Automatica*, vol. 37, pp. 377-390, 2001.
- [51] J. Y. Choi, J. Lee, J. H. Jung, M. Lee, *et al.*, "Sequential loop closing identification of multivariable process models," *Computers and Chemical Engineering*, vol. 24, pp. 809-814, 2000.
- [52] Z. J. Palmor, Y. Halevi, and N. Krasney, "Automatic tuning of decentralized PID controllers for TITO processes," *Automatica*, vol. 31(7), pp. 1001-1010, 1995.
- [53] Q. G. Wang and Y. Zhang, "A novel FFT-based robust multivariable process identification method," *Ind. Eng. Chem. Res*, vol. 40, pp. 2485-2494, 2001.
- [54] S. Panchapakesan, P. R. Krishnaswamy and G. P. Rangaiah, "On-line process identification from closed-loop tests under PI control," *Trans. IChemE (Part A)*, 76, 451- 457, 1998.
- [55] D. L. Melo, and J. C. Friedly, "On-line closed-loop identification of multivariable systems," *Ind. Eng. Chem. Res*, vol.31, pp. 274-281, 1992.
- [56] P. Grosdidier, and M. Morari, "Computer aided methodology for the design of decentralized controllers," *Computers & Chemical Engineering*, vol.11, pp. 423-433, 1987.
- [57] M. S. Chiu and Y. Arkun, "Decentralized control structure selection based on integrity considerations," *Ind. Eng. Chem. Res.*, vol. 29, pp. 369-373 , 1990.
- [58] E. H. Bristol, "On a new measure of interactions for multivariable process control," *IEEE Trans. AC*, vol.11, pp.133-134, 1966.
- [59] M. T. Avoy, Y. Arkun, *et al*, "A new approach to defining a dynamic relative gain," *Control Engineering Practice*, vol. 11, pp. 907-914, 2003.

- [60] Q. Bi, W. J. Cai., E. L. Lee, *et al.*, "Robust identification for first-order plus dead-time model from step response," *Control Engineering Practice*, vol. 7, pp. 71-77, 1999.
- [61] Q. Bi, W. J. Cai, Q. G. Wang and C. C. Hang, "Advanced controller auto-tuning and its application in HVAC systems," *Control Engineering Practice*, vol. 8, pp. 633-644, 2000.
- [62] L. Wang and W. R. Cluett, "System identification based on closed-loop step response data," *IEE Proc.-D*, vol.141, pp. 107-110, 1994.
- [63] Q. G. Wang, B. Huang, and X. Guo, "Auto-tuning of TITO decoupling controllers from step tests," *ISA Trans*, vol. 39, pp. 407-418, 2000.
- [64] Z. X. Zhu and A. Jutan, "Loop decomposition and dynamic interaction analysis of decentralized control systems," *Chemistry Engineering Science*, vol. 51, pp. 3325-3335, 1995.
- [65] F. G. Shinskey, *Process Control Systems*. McGraw-Hill, New York, 1988.
- [66] E. A. Wolff and S. Skogestad, "Operation of integrated three-product (Petlyuk) distillation columns," *Ind. Eng. Chem. Res.*, vol. 34, pp. 2094-2103, 1995.
- [67] J. E. Hansen, S. B. Jorgensen, *et. al.*, "Control structure selection for energy integrated distillation column," *Journal of Process Control*, vol. 8, pp. 185-195, 1998.
- [68] P. Grosdidier and M. Morari, "Closed-loop properties from steady-state gain information," *Ind. Eng. Chem. Fundam.*, vol. 24, pp. 221-235, 1985.
- [69] M. Witcher and T. J. McAvoy, "Interacting control systems: Steady state and dynamic measurement of interaction," *ISA Transactions*, vol.16, pp. 35-41, 1977.
- [70] E. H. Bristol, "Recent results on interactions in multivariable process control," *71st Annual AIChE Meeting*, Houston, TX, 1979.
- [71] L. Tung, and T. Edgar, "Analysis of control-output interactions in dynamic systems," *AIChE Journal*, vol. 27, pp. 690- 693, 1981.

- [72] M. S. Chiu and Y. Arkun, "A methodology for sequential design of robust decentralized control systems," *Automatica*, vol. 28, pp. 997-1001, 1992.
- [73] J. Lee, W. Cho and T. F. Edgar, "Multiloop PI controller tuning for interacting multivariable processes," *Computers & Chemical Engineering*, vol. 22, pp. 1711-1723, 1998.
- [74] D. Q. Mayne, "The design of linear multivariable systems," *Automatica*, vol. 9, pp. 201-207, 1973.
- [75] P. Grosdidier and M. Morari, "Interaction measures under decentralized control," *Automatica*, vol. 22, pp. 309-319, 1986.
- [76] T. K. Lee, J. Shen and M. S. Chiu, "Independent design of robust partially decentralized controllers," *Journal of Process Control*, vol. 11, pp. 419-428, 2001.
- [77] S. Skogestad and M. Morari, "Robust performance of decentralized control systems by independent design," *Automatica*, vol. 25, pp. 119-125, 1989.
- [78] W. L. Luyben, and A. Jutan, "Simple method for tuning SISO controllers in multivariable system," *Ind. Eng. Chem. Proc. Des. Dev.*, vol. 25, pp. 654-660, 1986.
- [79] D. Semino and C. Scali, "Improved identification and autotuning of controllers for MIMO processes by relay techniques," *Journal of Process Control*, vol. 8, pp. 219-227, 1997.
- [80] D. Chen and D. E. Seborg, "Design of decentralized PI control systems based on Nyquist stability analysis," *Journal of Process Control*, vol. 13, pp. 27-39, 2003.
- [81] A. Niederlinski, "A heuristic approach to the design of linear multivariable interacting subsystems," *Automatica*, vol. 7, pp. 691-701, 1971.
- [82] T. J. McAvoy, *Interaction Analysis Principles and Applications*. Instrument Society of America, Research Triangle Park, 1983.

- [83] M. M. Galarrage, T. J. McAvoy, T. E. Marlin, "Short-cut operability analysis. 2. estimation of f_i detuning parameter for classical control systems," *Ind. Eng. Chem. Res.*, vol. 26, pp. 511-521, 1987.
- [84] T. J. McAvoy, "Connection between relative gain and control loop stability and design," *AIChE Journal*, vol. 27, pp 613-619, 1981.
- [85] I. L. Chien, H. P. Huang, and J. C. Yang, "A simple multiloop tuning method for PID controllers with no proportional kick," *Ind. Eng. Chem. Res.*, vol. 38, pp.1456-1468, 1999.
- [86] S. H. Shen and C. C. Yu, "Use of relay-feedback test for autotuning of multivariable systems," *AIChE Journal*, vol. 40, pp.627-646, 1994.
- [87] W. K. Ho, T. H. Lee and O. P. Gan, "Tuning of multiloop proportional-integral-derivative controllers based on gain and phase margin specifications," *Ind. Eng. Chem. Res.*, vol. 36, pp. 2231-2238, 1997.
- [88] D. Chen and D. E. Seborg, "Multiloop PI/PID controller design based on Gershgorin bands," *IEE Proc-C.*, vol. 149, pp.68-73, 2002.
- [89] I. Rusnak, "Generalized PID controllers and its application to control of ultrasonic and electric motors," *IFAC Workshop PID*, Spain, pp. 125-130, 2000.
- [90] M. J. Grimble, "H-infinity PID controllers," *Trans. IMC*, vol. 13, pp. 112-120, 1991.
- [91] D. E. Rivera, S. Skogestad and M. Morari, "Internal model control for PID controller design," *Ind. Eng. Chem. Proc. Des. Dev.*, vol. 25, pp. 252-265, 1986.3
- [92] P. Vega, C. Prada and V. Aleixandre, "Self-tuning predictive PID controller," *IEE Proc-C.*, vol. 138, pp. 303-311, 1991.
- [93] M. R. Katebi and M. H. Moradi, "Predictive PID controllers," *IEE Proc-C.*, vol. 148, pp.478-487, 2001.
- [94] M. H. Moradi, M. R. Katebi and M. A. Johnson, "The MIMO predictive PID controller design," *Asian Journal of Control*, vol. 4, pp.452-463, 2002.

- [95] D. Uduehi, A. Ordys and M. J. Grimble, "Multivariable PID controller design using online GPC optimization," *Proceedings of IEEE International Conference on Control Applications*, Glasgow, U.K., 2002.
- [96] D. Clarke, *Advances in Model-Based Predictive Control*. Oxford University Press, 1994.
- [97] E. F. Camacho, C. Bordons, *Model Predictive Control*. Springer, 1999.
- [98] B. A. Foss, T. A. Johansen A. V. Sorensen, "Nonlinear predictive control using local model," *Control Engineering Practice*, vol. 3, pp.389-396, 1995.
- [99] O. Hecker, *et al.* "Nonlinear system identification and predictive control of a heat exchanger based on local linear fuzzy models," *Proceedings of the American Control Conference*, Albuquerque, New Mexico, pp.3294-3298, 1997.
- [100] Breiman, *et al.* *Classification and Regression Trees*. Wadsworths & Brooks, Monterey, CA, 1984.
- [101] J. H. Friedman, "Multivariable adaptive regression splines," *Annals of Statistics*, vol.19, 1991.
- [102] A. Banerjee and Y. Arkun, *et al.*, "Estimation of nonlinear systems using linear multiple models," *AIChE Journal*, vol. 43, pp. 1204-1226, 1997.
- [103] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Transactions on SMC*, vol. 15, pp. 116-132, 1984.
- [104] P. Kesavan and J. H. Lee, "Diagnostic tools for multivariable model-based control systems," *Ind. Eng. Chem. Res.*, vol. 36, pp. 2725-2738, 1997.
- [105] T. J. Harris, C. T. Seppal, L. Desborough, "A review of performance monitoring and assessment techniques for univariate and multivariate control systems," *Journal of Process Control*, vol. 9, pp.1-17, 1999.
- [106] S. J. Qin, "Control performance monitoring – A review and assessment," *Computer and Chemical Engineer*, vol. 23, pp.173-186, 1998.

- [107] K. J. Astrom, *Introduction to stochastic control theory*. Academic Press, New York, 1970.
- [108] T. J. Harris, "Assessment of control performance," *Can. J. Chem. Eng.*, vol. 67, pp. 856-861, 1989.
- [109] B. Huang and S.L. Shah, *Performance Assessment of Control Loops*. Springer-Verlag, London, 1999.
- [110] S. Jochen and C. Ali, "Multivariable MPC system performance assessment, monitoring, and diagnosis," *Journal of Process Control*, vol. 14, pp.1-17, 2004.
- [111] R. S. Patwardhan and S. L. Shah, "Issues in performance diagnosis of model-based controllers," *Journal of Process Control*, vol. 12, pp.413-427, 2002.
- [112] Y. Zhang and M.A. Henson, "A performance measure for constrained model predictive controllers," *European Control Conference*, Germany, 1999.
- [113] M. Yaqub and S. M. Zubair, "Capacity control for refrigeration and airconditioning systems: A comparative study," *Int. J. Energy Res. Tech.*, vol. 123, pp. 92-99, Apr. 2001.
- [114] R. Shoureshi, "Intelligent control systems: Are they are real," *J. Dynamic Syst., Measur. Contr.*, vol. 115, pp. 15-19, Jan. 1993.
- [115] J. Teeter and M. Y. Chow, "Application of functional link neural network to HVAC thermal dynamic system identification", *IEEE Trans. Industrial Electronic*, vol. (45), pp. 170-175, 1998.
- [116] J. M. Sousa, R. Babusla and H. B. Verbruggen, "Fuzzy predictive control applied to an air-conditioning system", *Control Engineering Practice*, vol. 10(5), pp. 1395-1406, 1997.
- [117] C. Ghiaus, "Fuzzy model and control of a fan-coil", *Energy and Buildings*, Vol. 33, pp. 545-551, 2001.
- [118] T. I. Salsbury, "A temperature controller for VAV air-handing units based on simplified physical models", *HVAC&R Research*, vol. (3), pp.264-279, 1998.

- [119] M. Kasahara, T. Matsuba, *et al.* "Design and tuning of robust PID controller for HVAC systems", *ASHRAE Trans.* vol.105(2), pp.154-166, 1999.
- [120] Ming He, Wen-Jian Cai and Shao-Yuan Li, "Multiple fuzzy model-based temperature predictive control for HVAC systems," *Information Sciences*, (In Press).
- [121] A. S. Betzaida and V. R. Miguel, "Nonlinear control of a heating, ventilating, and air conditioning system with thermal load estimation," *IEEE Trans. Contr. Syst. Technol.*, vol. 7, pp. 56-63, Jan. 1999.
- [122] R. G. Carlos and V. R. Miguel, "Decoupled control of temperature and relative humidity using a variable air volume HVAC system and non-interacting control," *Proc. of the IEEE International Conf.-CA*, pp.1147-1151, 2001.
- [123] G. Singh, M. Zaheer-uddin and R. V. Patel, "Adaptive control of multivariable thermal processes in HVAC systems," *Energy Conversion & Management*, vol. 41, pp.1671-1685, 2000.