Design and analysis of structured controllers

Antonius Yudi Sendjaja

2011

Antonius, Y. S. (2011). Design and analysis of structured controllers. Doctoral thesis,
Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/44600>

<https://doi.org/10.32657/10356/44600>

DESIGN AND ANALYSIS OF STRUCTURED CONTROLLERS

by

ANTONIUS YUDI SENDJAJA

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

SCHOOL OF CHEMICAL AND BIOMEDICAL ENGINEERING

2011

DECLARATION

I hereby declare that the work presented in this thesis is, to the best of my knowledge and belief, original and this material has not been submitted, either in whole or in part, for any other degree at this or any other university.

Signature : _____

ANTONIUS YUDI SENDJAJA

School of Chemical and Biomedical Engineering

Nanyang Technological University

62 Nanyang Drive

Singapore 637459

Acknowledgement

Time flies on restless pinions - constant never. The quote by Friedrich Schiller is indeed true for my Ph.D.'s journey. It has been four years since I got Dr. Vinay Kariwala's signature stating his willingness to nurture me during one phase of my life's journey. During these four years, he indeed fulfilled his promise. He always asked me to explore many things on my own despite his constant intercession to make sure that I am on the right trajectory. I am truly grateful for his way of training. My Ph.D.'s voyage would have been a mess without his concern and caring. I really thank him for his patience, guidance and advice. I learned not only the knowledge, theory, and art of process control, but also the true meaning of living my life.

I also must acknowledge some undergraduate students in my research group, especially Ms. Lesley Wong, Ms. Jacqueline How, and Mr. Ng Zhen Fu who helped me with the thesis. They are the smartest and the most diligent persons that I can wish for working with me during my Ph.D.'s journey. I am really grateful to be able to collaborate with them. I also thank them for their contributions to this thesis. There are other students that I must give thank to, Jing Mian, Chen Nee, and Wei Qiang. All the time that I spent with you was a priceless experience for me. I wish all of you great future ahead and that you excel in your career, whichever path you choose.

I really feel that one needs friends who surround you when you live far away from home. They save you from the 'stranded in the middle of nowhere' feeling, moreover, most of us are in the same 'postgrad' boat. I must give credit to all of my friends, to Handarmin,

Susan, Herlina, Eveline, Wahyu, Nancy, Noel, Katherine, and Yenni. I am sure that they are one of the reasons, if not the only, for me to keep on going and not letting me down during my voyage on this boat.

I also would like to acknowledge the love and support from my family back home in Indonesia, especially from my mother and my sister, Silvi. I also want to pay tribute to my late father. This thesis is dedicated to him. Finally and most substantially, I have to give thanks to God Almighty for all His Grace throughout my life.

Antonius Yudi Sendjaja

Singapore, May 2011

Summary

For control of industrial processes, although fully centralized high-order controllers can theoretically provide good performance, simple controllers are used in practice due to the ease in design, maintenance, and operator acceptability. These simple controllers often have a structure, e.g. only certain elements of controller are allowed to be non-zero or the controller is restricted to be a low order transfer function. In this thesis, several tools are developed for the design, analysis and assessment of such structured controllers, namely RTD-A, PID, and decentralized controllers.

In the first part of this thesis, we consider the design and analysis of RTD-A and PID controllers used for control of SISO processes. A block diagram representation of the RTD-A controller is developed, which allows for a better understanding of the control law. Based on the block diagram representation, a semi-analytical tuning rule is proposed, which is shown to provide reasonable performance for a wide range of processes. Subsequently, the sums of squares (SOS) programming technique is presented to characterize the fundamental limitations on the achievable performance due to the PID structure of the controller. It is shown that the SOS programming provides a tight lower bound on the least output variance achievable using PID controllers.

In the second part of this thesis, decentralized control of multivariable systems is studied extensively. The SOS programming technique is shown to be applicable for finding a tight lower bound on the least output variance achievable using decentralized controllers. A modified minimum variance benchmark is derived for the performance assessment of

decentralized controllers on a loop by loop basis. This benchmark can be estimated reliably from the closed-loop operating data making it a practical tool for the controller performance assessment. For efficient selection of input-output pairings for large scale processes, a genetic algorithm is developed and illustrated using the relative gain array (RGA) and μ -interaction measure as the selection criteria. Lastly, using the case study of a benchmark solid oxide fuel cell, it is shown that a systematically designed decentralized controller can meet the operational objectives of processes, which are difficult to control, and can even outperform advanced controllers such as model predictive control. An offshoot of this work is the development of a systematic method for the selection of controlled variables for processes with tight operating constraints.

Contents

1	Introduction	1
1.1	An Overview of Structured Controllers	1
1.2	Motivation and Scope	5
1.3	Thesis Overview	7
2	Analysis and Tuning of RTD-A Controllers	11
2.1	Introduction	11
2.2	RTD-A Controller	13
2.2.1	RTD-A Algorithm	14
2.2.2	Ogunnaike's Tuning Rule	16
2.3	Block Diagram Representation	17
2.3.1	Derivation	18
2.3.2	Analysis	20
2.4	Tuning Rule for RTD-A Controller	24
2.4.1	Overall Aggressiveness Parameter θ_A	24
2.4.2	Setpoint Tracking Parameter θ_T	26
2.4.3	Disturbance Rejection Parameter θ_D	26
2.4.4	Robustness Parameter θ_R	27
2.5	Evaluation of Proposed Tuning Rule	28
2.6	Chapter Summary	39

2.A	Appendix: Lower Bound on θ_R	39
3	Achievable PID Performance	41
3.1	Introduction	41
3.2	Sums of Squares Programming	43
3.3	Lower bound on Achievable Performance	45
3.4	Weighted Input and Output Performance	54
3.5	Chapter Summary	56
4	Achievable Performance of Decentralized Controllers	57
4.1	Introduction	57
4.2	Problem Formulation	59
4.3	Conventional Approach	60
4.4	MV Benchmark for Decentralized Controllers	61
4.5	MV benchmark for Decentralized PID Controllers	64
4.6	Chapter Summary	68
5	Practical Performance Assessment of Decentralized Controllers	69
5.1	Introduction	69
5.2	MV Benchmark for Decentralized Controllers	73
5.2.1	Conventional Approach	74
5.2.2	Modified MV Benchmark	74
5.2.3	Generalization to $n_y \times n_y$ Systems	81
5.3	Estimation from Data	84
5.4	Chapter Summary	87
6	An Efficient Genetic Algorithm for Input-Output Pairing Selection	89
6.1	Introduction	89

6.2	Genetic Algorithm for Pairing Selection	91
6.2.1	General Principle	91
6.2.2	Application to Pairing Selection	93
6.3	Input-Output Pairing Selection Method	95
6.3.1	Relative Gain Array	95
6.3.2	μ -Interaction Measure	97
6.4	Numerical Tests	98
6.5	Chapter Summary	101
6.A	Appendix: Illustrative Example	101
7	Decentralized Control of Solid Oxide Fuel Cells	105
7.1	Introduction	105
7.2	Nonlinear Dynamic Model	108
7.3	Model Linearization	111
7.4	Decentralized Controller Design	113
7.4.1	Selection of Controlled Variables	113
7.4.1.1	Single Variable as CV	115
7.4.1.2	Measurement Combination as CV	116
7.4.2	Pairing Selection	118
7.4.3	PID Controller Tuning	119
7.4.3.1	Single Variable as CV	120
7.4.3.2	Measurement Combination as CV	120
7.5	Closed-loop Simulations	122
7.6	Chapter Summary	126
7.A	Appendix: Linearization of SOFC Model	127
8	Conclusions and Future Work	129

8.1	Conclusions	129
8.2	Directions for Future Work	130

List of Figures

1.1	Schematic diagram of Model Predictive Controller.	3
1.2	Distillation column control using (a) Fully centralized controller (b) Decentralized controller	4
2.1	Schematic diagram of closed loop system with RTD-A controller	14
2.2	Schematic diagram of Internal Model Controller.	19
2.3	Closed-loop block diagram of RTD-A controller	20
2.4	Equivalent closed-loop block diagram for RTD-A algorithm	23
2.5	Closed-loop simulation for Case 1	31
2.6	Closed-loop simulation for Case 2	31
2.7	Closed-loop simulation for Case 3	32
2.8	Closed-loop simulation for Case 4	33
2.9	Closed-loop simulation for Case 5	35
2.10	Closed-loop simulation for Case 6	35
2.11	Closed-loop simulation for Case 7	37
2.12	Closed-loop simulation for Case 8	37
2.13	Closed-loop simulation for Case 9	38
3.1	Block diagram of closed loop system	45
3.2	Frequency response of closed-loop system with and without weighting factor for Example 3.3	54

3.3	Trade-off between $\text{Var}(y)$ and $\text{Var}(\Delta u)$ for Case 1	55
4.1	Block diagram of closed loop system	59
5.1	Illustration of improved operation through output variance reduction	70
5.2	Insufficiency of available MV benchmarks for performance assessment of decentralized controllers	71
5.3	Presence of parallel path (shown with thick line) from u_1 to y_1 for 2×2 systems	76
5.4	Impulse response coefficients of disturbance models of open-loop and partially closed-loop systems for Example 5.1	78
5.5	Impulse response coefficients of disturbance models of open-loop and partially closed-loop systems for Example 5.2	80
5.6	Impulse response coefficients of disturbance models of open-loop and partially closed-loop systems for Example 5.3	84
6.1	Flow diagram of genetic algorithm	92
6.2	Computational time for pairing assignment based on RGA	99
6.3	Relative error for GA method based on RGA	99
6.4	Computational time for pairing assignment based on μ -IM	100
6.5	Relative error for GA method based on μ -IM	101
7.1	Dynamic model of solid oxide fuel cells	108
7.2	Comparison of dynamics between nonlinear (solid line) and linear (dashed line) models	112
7.3	Magnitudes of $ P_u(j\omega) $ and $ P_y(j\omega) $ with control of V_r using $q_{O_2}^{in}$	115
7.4	Magnitudes of $ P_u(j\omega) $ and $ P_y(j\omega) $ with control of V_r and U_f	116
7.5	Magnitudes of $ P_u(j\omega) $ and $ P_y(j\omega) $ with control of V_r and a linear combination of MVs	118

7.6	Schematic diagram of closed-loop system using linear prediction of U_f . . .	121
7.7	Schematic diagram of closed-loop system using measurement combination as CV	122
7.8	Closed-loop responses for nominal conditions	123
7.9	Closed-loop responses for 1 st robustness test	124
7.10	Closed-loop responses for 2 nd robustness test	124

List of Tables

2.1	Ogunnaike's RTD-A tuning rule	17
2.2	Proposed RTD-A tuning rule	28
2.3	Case studies and their first order approximation	30
2.4	RTD-A/PI controller parameters	32
2.5	IAE of RTD-A and SIMC-PI controllers for setpoint tracking (SP) and disturbance rejection (D)	33
2.6	TV of RTD-A and SIMC-PI controllers for setpoint tracking (SP) and disturbance rejection (D)	34
2.7	Robustness properties of RTD-A and SIMC-PI controllers	36
3.1	Case studies for performance assessment of PID controllers	49
3.2	Lower (LB) and upper (UB) bounds on achievable PI control performance .	51
3.3	Lower (LB) and upper (UB) bounds on achievable PID control performance	52
5.1	Performance assessment of sidestream column/stripper distillation column using loop by loop analysis	83
6.1	Selection procedure on the first generation.	102
6.2	Selection procedure on the second generation.	103
7.1	Parameters for the dynamic model of SOFC [92, 139]	109
7.2	Operating conditions and constraint limits for SOFC [139]	110

7.3	Integral Absolute Error (IAE) and Total Variation (TV) for SOFC	125
-----	---	-----

Nomenclature

In this thesis, unless otherwise stated, the following notations is used.

Symbols

$(\cdot)_i$	i^{th} element of vector, i^{th} column of matrix
$(\cdot)_{ij}$	ij^{th} element of matrix
$(\cdot)^T$	Transpose of matrix
$(\cdot)^{-1}$	Inverse of matrix
$(\cdot)^\dagger$	Pseudo-inverse of matrix
$(\cdot)!$	Factorial, $n! = \prod_{i=1}^n i$
$(\cdot) \geq 0$	Positive semi-definite Matrix
$(\cdot) \circ (\cdot)$	Hadamard or element-wise matrix multiplication
$ \cdot $	Absolute value
$\ \cdot\ _2$	$\mathcal{H}_2$ -norm transfer function
$\ \cdot\ _{sum}$	sum-norm of matrix, defined as $\sum_i \sum_j (\cdot)_{ij} $
$det(\cdot)$	Determinant of matrix
$diag(\cdot)$	Diagonal matrix consists of elements in $(\cdot)$
q^{-i}	i -step Back-shift operator
$tr(\cdot)$	Trace of matrix
$E[\cdot]$	Expectation operator
$\mathbb{R}$	Space of real numbers
$\mathbb{R}^{m \times n}$	$m \times n$ dimensional space of real numbers
a	Disturbance
d	Order of interactor matrix, time delay (Chap 5)
e	Error
$\hat{e}_D$	Unbiased error prediction
g, G	Transfer matrix connecting controlled and manipulated variables
h, H	Transfer function connecting controlled variable and disturbances
j	Imaginary number, $\sqrt{-1}$
k, K	Transfer function of controller
m	Time delay of discrete-time transfer function
n_y	Number of outputs of a transfer matrix
s	Laplace variable (Chap 1,2,6,7), Sensitivity function (Chap 3)

u	Input, manipulated variable
$\bar{u}$	Scaled input
y	Output, controlled variable
$\hat{y}$	Predicted output (Chap. 2), Scaled output (Chap. 7)
y_{sp}	Setpoint
y^*	Output reference trajectory
z	Subsets of output, controlled variable
D	Interactor matrix
D_d	Disturbance scaling matrix
D_u	Input scaling matrix
D_y	Output scaling matrix
$\bar{G}$	Invertible part of transfer function G
I	Identity matrix
M	Combination matrix
K_c	Controller gain
K_p	Steady state gain
N	Prediction horizon
N_c	Number of objective functions
N_p	Number of individuals in population
P_d	Partially closed-loop transfer matrix connecting controlled and disturbances
P_u	Partially closed-loop transfer matrix connecting controlled and manipulated variables
S	Sensitivity matrix, $S = (I + GK)^{-1}$
T	Sensitivity matrix, $T = GK(I + GK)^{-1}$
α	Pole of first order discrete-time model
ϵ	Expected gain margin
η	Minimum variance benchmark (Chap. 5); Weighted factor for IMC controller (Chap. 2)
λ_{ij}	Relative gain between the i^{th} input and j^{th} output
μ	Structured singular value
θ	Time delay for continuous models
θ_A	RTD-A tuning parameter for overall aggressiveness
θ_D	RTD-A tuning parameter for disturbance rejection
θ_R	RTD-A tuning parameter for robustness
θ_T	RTD-A tuning parameter for setpoint tracking
ρ	Composite parameter (Chap. 2); Penalty factor on multiobjective optimization (Chap. 3)
τ	Time constant for continuous model
τ_c	Desired closed-loop time constant
τ_D	Derivative time of PID controller
τ_I	Integral time of PID controller
ω	Frequency
Δ	Difference operator (Chap 3,4), Uncertainty, perturbation matrix (Chap 6)
ΔT	Sampling time
Λ	Relative gain array

Abbreviations

B3	Bidirectional Branch and Bound
BAB	Branch and Bound
CV	Controlled Variable
FCOR	Filtering and Correlation
FOPTD	First Order plus Time Delay
GA	Genetic Algorithm
GAP	Generalized Analytical Predictor
IAE	Integral Absolute Error
IMC	Internal Model Controller
IO	Input-Output
LMI	Linear Matrix Inequalities
LP	Linear Prediction
LQG	Linear Quadratic Gaussian
LTI	Linear Time Invariant
MC	Measurement Combination
MILP	Mixed Integer Linear Programming
MIMO	Multi Input Multi Output
MPC	Model Predictive Controller
MPHC	Model Predictive Heuristic Controller
MV	Manipulated Variable (Chap. 7); Minimum Variance (Chap. 3, 4, 5)
PI	Proportional Integral
PID	Proportional Integral Derivative
RGA	Relative Gain Array
RTD-A	Robustness, setpoint Tracking, disturbance Rejection-overall Aggressiveness
SDP	Semi-Definite Programming
SIMC	Simple (Skogestad's) Internal Model Control
SISO	Single Input Single Output
SOFC	Solid Oxide Fuel Cells
SOS	Sums of Squares
TV	Total Variation
μ -IM	μ -Interaction Measure

Publications Related to the Thesis

Journal Papers

1. A. Y. Sendjaja and V. Kariwala, Minimum variance benchmark for performance assessment of decentralized controllers, submitted to *Journal of Process Control*, 2010.
2. A. Y. Sendjaja and V. Kariwala, Decentralized control of solid-oxide fuel cells, accepted for publication in *IEEE Transactions on Industrial Informatics*.
3. A. Y. Sendjaja, Z. F. Ng, S. S. How, and V. Kariwala, Analysis and tuning of RTD-A controllers, *Industrial and Engineering Chemistry Research*, 50(6), pp. 3415-3425, 2011.
4. A. Y. Sendjaja and V. Kariwala, Achievable PID performance using sums of squares programming, *Journal of Process Control*, 19(6), pp. 1061-1065, 2009.

Conference Proceedings

1. Z. F. Ng, S. S. How, A. Y. Sendjaja and V. Kariwala, Analysis and Tuning of RTD-A Controllers, In *Proceedings of PSE ASIA 2010*, Singapore, 2010.
2. A. Y. Sendjaja and V. Kariwala, A Practical Approach for Performance Assessment of Decentralized Controllers, In *Proceedings of 2nd CUTSE International Conference*, Miri, Malaysia, 2009.

3. A. Y. Sendjaja and V. Kariwala, Performance Assessment of Decentralized Controllers, In *Proceedings of ADCHEM*, Istanbul, Turkey, 2009
4. A. Y. Sendjaja, W. K. M. Lesley and V. Kariwala, Decentralized Control of Solid Oxide Fuel Cell, In *Proceedings of 15th Regional Symposium of Chemical Engineering*, Kuala Lumpur, Malaysia, 2008.
5. A. Y. Sendjaja and V. Kariwala, On Achievable PID Performance, In *Proceedings of 14th Regional Symposium on Chemical Engineering*, Yogyakarta, Indonesia, 2007.

Chapter 1

Introduction

1.1 An Overview of Structured Controllers

Automatic controllers are commonly used for maintaining an industrial process at the desired operating point. In general, a controller can have an arbitrary (provided it is causal and realizable) form, *e.g.* a high order transfer function matrix. In practice, however, simple controllers are preferred due to the ease in design, maintenance, and operator acceptability. These simple controllers often have a structure, *e.g.* only certain elements of controller are allowed to be non-zero or the controller is restricted to be a low order transfer function. This thesis deals with the design, maintenance, and analysis of three structured controllers, namely proportional-integral-derivative (PID) controller [134], RTD-A controller [89], and decentralized controller [124].

For control of single-input single-output (SISO) systems, PID controllers are regularly used in the process industries. Recent survey shows that about 98% of the controllers used

in chemical plants are PID controllers [18]. The PID control law is defined as [134]

$$u(s) = K_c \left(1 + \frac{1}{\tau_I s} + \tau_D s \right) e(s) \quad (1.1)$$

where $u(s)$ denotes the process input, $e(s)$ is the error and K_c , τ_I , and τ_D are the proportional gain, integral time, and derivative time, respectively.

The tuning of a PID controller is an established research area in process control, for which numerous methods have been proposed and applied, such as Ziegler-Nichols (ZN) [155], direct synthesis method [15], setpoint overshoot method [116], gain and phase margin [145], optimal error [140] and Internal Model Control (IMC) [83]. Its popularity stems from its simplicity and ease of implementation [80]. Nevertheless, the PID controller has some drawbacks, such as the necessity of trade-off between the setpoint tracking and disturbance rejection. Furthermore, the controller parameters can take any values making transparent tuning difficult and the control law does not take long term process behavior into account.

The weaknesses of a PID controller has motivated the researchers to develop alternate control algorithms. One such algorithm is model predictive control (MPC) [79]. MPC can handle multivariable interactions and input and output constraints explicitly [96]. MPC uses a model to predict output behavior for the next R time steps, where R is the prediction horizon. The controller minimizes the weighted sum of output deviation, defined as the difference between predicted output and its setpoint trajectory, and rate of input change for the next R and L time steps, respectively, where L is control horizon. The value of the input for the next time step obtained by solving the optimization problem is applied to the process. The procedure is repeated at every time step. The schematic diagram of MPC is shown in Figure 1.1.

Another algorithm is the recently developed RTD-A controller [89] that consists of three primary tuning parameters (θ_R , θ_T and θ_D) and one secondary tuning parameter (θ_A). An

Figure 1.1: Schematic diagram of Model Predictive Controller.

RTD-A controller offers transparent and independent tuning of its parameters for setpoint tracking and disturbance rejection, in addition to its predictive ability like MPC [79]. As the simplicity and ease of the implementation of PID controller are retained, the RTD-A controller can be seen as a potential candidate to complement or even replace the PID controller in future.

In most of the industrial processes, several variables need to be controlled by using the available manipulated variables. As an example, let us consider a distillation column shown in Figure 1.2, where the control objective is to maintain the compositions of the overhead and bottom streams at the specified setpoints by manipulating the reflux ratio and steam flowrate. One possibility is to control the composition of both streams using a fully centralized controller, which takes information from both the compositions and manipulates both the inputs simultaneously. Another strategy involves the use of decentralized controller, which uses the information of overhead composition to manipulate reflux ratio and the information of bottom composition to manipulate steam flowrate.

Figure 1.2: Distillation column control using (a) Fully centralized controller (b) Decentralized controller

A centralized controller can always provide better performance and constraint handling than a decentralized controller. However, in addition to their inherent simplicity, a decentralized control system exhibits several advantages over a fully centralized design. In the ideal case, these advantages include [124]

1. The individual controllers can be brought in and out of service providing flexibility of operation.
2. Due to the localized effect of the individual controllers, the system can be made fault tolerant with ease, particularly in the case of sensor or actuator failure.
3. The individual controllers are easier to tune online in presence of changing process conditions.
4. Simpler models can be used to design and tune the controllers *e.g.* independent tuning of different loops reducing the modeling requirements.
5. The online computational effort is less than their centralized counterparts and implementation is simpler.

Note that all of the mentioned advantages may not be realized simultaneously or may only be realized at the cost of degraded performance. Nevertheless, decentralized control seems to be the almost exclusive choice for control of large-scale systems.

In process industries, the individual controllers of the decentralized control system are usually restricted to be of PID-type. For design of decentralized PID controllers, several approaches have been proposed. A common way involves independent design of each PID controller, where a detuning factor is used to overcome the loop interactions. Simple and effective algorithms for independent tuning for decentralized PID controller are available in [41, 145, 147]. These methods are suitable for design of decentralized PID controller for processes with any dimensions.

1.2 Motivation and Scope

In this thesis, we consider control of linear time invariant (LTI) systems with structured controllers. The objective of this thesis is to develop systematic methods for the design and analysis of RTD-A, PID, and decentralized controllers.

As mentioned earlier, the RTD-A controller has been proposed as an alternative to the PID controller for the control of SISO processes. For tuning of RTD-A controller, a tuning rule has been proposed based on robust stability analysis [90]. The available tuning rules for the RTD-A controller are, however, aggressive and their application to practical problems can be difficult. This thesis focuses on finding a better tuning rule for the RTD-A controller.

Before a controller is designed, it is important to know whether the performance requirement can be satisfied using a controller with specified structure. The fundamental limitations arising due to process characteristics, such as time delay, unstable poles, and unstable zeros are well-known, see *e.g.* [45, 115, 124]. However, few results are available on fundamental limitations arising from the controller structure, such as PID [67] and decentralized controllers [57]. This thesis explores the effect of controller structure on

the achievable performance.

The knowledge about fundamental limitation on the achievable performance is useful for various tasks including controller performance assessment. Minimum variance (MV) benchmark, defined as ratio of achievable and observed output variances, is used to determine whether significant improvements in the closed-loop performance are possible through controller retuning. The MV benchmarks available for SISO [133] and multi-input multi-output (MIMO) processes [45], however, do not take the controller structure into account and can lead to incorrect conclusions regarding the performance of structured controllers, such as decentralized controllers. In this thesis, a modified MV benchmark is developed for performance assessment of decentralized controllers on a loop by loop basis using closed-loop data.

Another problem addressed in this thesis is related to the selection of input output pairings for the decentralized controller. In the literature, many methods are available for the pairing selection, such as the relative gain array (RGA) [11], relative interaction array [154], μ -interaction measure (μ -IM) [26, 27], ν -gap metric [106], and dissipativity [4, 5]. These methods require exhaustive search of all possible alternatives, which is infeasible for large processes. Thus, an efficient algorithm for finding nearly optimal pairings is sought in this thesis.

Lastly, the design of a decentralized control system for a solid oxide fuel cell (SOFC) based on a benchmark model [92] is considered. The control of SOFC is challenging due to its slow response time, tight operating constraints and unavailability of some key measurements [64]. These challenges have motivated researchers to propose the use of advanced control strategies such as $\mathcal{H}_\infty$ controller [64] and data-driven predictive controller [139]. In this thesis, a decentralized control system is systematically designed and shown to meet the operational objectives of SOFC.

1.3 Thesis Overview

The main objective of this thesis is to develop a simple and systematic procedure for design and analysis of structured controllers, such as PID and decentralized controllers. Systematic controller design includes three steps, namely control structure design, performance analysis, and controller tuning. This thesis aims at solving a number of salient issues for each of the steps. Control structure design, in particular pairing assignment, is the main contribution of Chapter 6, while achievable performance using structured controllers is discussed in Chapter 3 - 5. Chapter 2 focuses on development of controller tuning method for a structured controller, *i.e.* RTD-A controller. In Chapter 7, a case study on designing a simple control system for SOFC is presented. In this chapter, it is also shown that systematic controller design has the potential to provide good control performance that exceeds general expectation. It is anticipated that the tools developed in this thesis will pave the way for a systematic procedure for design of structured controllers.

This thesis can be divided into two parts. In the first part, the design and analysis of structured controllers for SISO systems is dealt with (Chapters 2 and 3), while the focus of the second part is on decentralized controllers (Chapters 4-7). A brief overview of each chapter is given next.

- Chapter 2 analyzes the newly developed RTD-A controller [89]. In this chapter, a block diagram representation of the RTD-A controller is developed. It is shown that the RTD-A controller can be viewed as a generalized analytical predictor (GAP) [141, 142] augmented with noise and setpoint filters. Based on the block diagram representation, semi-analytical tuning rule is proposed. It is shown that the RTD-A controller tuned using the proposed rule provides reasonable performance for a wide class of processes including inverse response and delay dominant processes.
- Chapter 3 deals with fundamental performance limitations arising due to the PID

structure of the controller. A key difficulty in characterizing the limitations due to the controller structure is the non-convexity of the resulting optimization problem. In this thesis, we present a global lower bound on the achievable PID performance, defined in terms of output variance, by solving a series of convex programs using sums of squares (SOS) programming. The result is also extended to minimize the weighted sum of the variances of input rate and output.

- Chapter 4 extends the SOS programming approach for characterizing fundamental limitations on the achievable performance of decentralized controllers. The result is also extended for the case when each individual controllers in the decentralized controller are restricted to be of PID type.
- Chapter 5 deals with the closed-loop performance assessment of decentralized controllers on a loop by loop basis using the MV benchmark. A modified MV benchmark is developed, where the interactions from other loops are accounted. Further, it is shown that the proposed MV benchmark can be directly estimated from the closed-loop data using filtering and correlation (FCOR) algorithm [45] with the knowledge of time delay between all the inputs and outputs.
- Chapter 6 explores the potential of evolutionary algorithms for the efficient selection of nearly optimal input-output pairings. A genetic algorithm (GA) with customized crossover and mutation operators is developed for the pairing selection using the RGA [11] and μ -IM [26] as the selection criteria. The efficiency is compared with the other available algorithms for the pairing selection in terms of the computational time and relative error.
- Chapter 7 discusses the systematic design of a decentralized PID controller for the SOFC to track changing power demand in the presence of disturbance. The procedure involves selection of controlled variables using the frequency

response analysis, pairing selection based on the RGA approach [11], and the PID controller tuning using SIMC tuning rules [121]. The performance and robustness of the controller are evaluated by applying the controller under different operating conditions. It is shown that the systematically designed decentralized PID controller meets all the operational objectives and outperforms data-driven predictive controller [139].

- Chapter 8 provides some concluding remarks and summarizes the contributions of this thesis. Some potential issues, which can be investigated in future, are also suggested.

Chapter 2

Analysis and Tuning of RTD-A Controllers

2.1 Introduction

The proportional-integral-derivative (PID) controller is the most widely used control algorithm for the regulation purposes in industrial plants [19, 69]. The PID controller has many advantages, such as simple controller structure, ease of implementation, and robustness to model errors [80]. The PID control algorithm, however, also has several weaknesses. For example, the tuning parameters of the PID controller affect the performance for both setpoint tracking and disturbance rejection. In such cases, achieving good performance for both setpoint tracking and disturbance rejection simultaneously can be difficult [89]. Furthermore, the controller parameters can take any values up to infinity,

¹Part of this chapter was presented in PSE Asia Conference, Singapore, 2010 [85].

²The contents of this chapter have been published in *Industrial and Engineering Chemistry Research*, 2011 [114]

which makes transparent tuning difficult. Lastly, PID controller does not take the long-term process behavior into account as it only reacts to reduce the error at the next time step.

A more advanced algorithm used in process industries is the model predictive controller (MPC) [79, 84, 100]. The MPC requires solving an optimization problem at every time step, which makes MPC computationally expensive. Thus, the application of MPC has largely been limited to medium and large scale multivariable processes in supervisory mode. In addition, despite the availability of some guidelines [20, 79, 117], the selection of tuning parameters for the MPC can be difficult due to its complex structure. Thus, many researchers have focussed on finding alternate control algorithms, which can overcome the drawbacks of the PID controller, while maintaining simplicity in the design and algorithm [22, 89, 93].

Recently, a new control algorithm, namely the RTD-A controller, has been proposed by Ogunnaike and Mukati [89]. The term “RTD-A” stands for robustness, setpoint tracking, disturbance rejection, and overall aggressiveness. The RTD-A controller has four tuning parameter (θ_R , θ_T , θ_D , and θ_A), which are directly related to the corresponding properties of the closed-loop system. All these parameters are normalized between 0 and 1, where a value closer to 0 implies that the corresponding property of the closed-loop system is more pronounced and *vice versa*. Among these parameters, θ_R , θ_T , and θ_D can be tuned independent of each other [89], *e.g.* improving the setpoint tracking ability does not decrease its disturbance rejection and robustness abilities. The RTD-A controller only requires a first order plus time delay (FOPTD) model, hence maintains the simplicity like PID controller, and has predictive ability as the MPC. With these advantages, the RTD-A controller has recently been applied for the control of vapor deposition process [90] and hypnosis regulation [128, 151].

Although the RTD-A controller has been shown to have distinct advantages over the PID controller [89], its properties are not well understood. Recently, tuning guidelines for the

RTD-A controller have been proposed based on robust stability analysis [90], where trade-off between robust stability and controller aggressiveness are sought. This tuning rule is referred to as Ogunnaike's tuning rule. This tuning rule, however, only provide bounds on the allowable tuning parameters for stability of the closed-loop system and requires the selection of an uncertainty parameter, which can be difficult in practice.

In this chapter, a block diagram representation of RTD-A controller is proposed for easier interpretation of the controller. Subsequently, the block diagram is analyzed to understand the role of each tuning parameter in the overall controller structure. It is shown that the RTD-A controller can be viewed as a generalized analytical predictor (GAP) [141, 142] augmented with setpoint and noise filters. Based on the block diagram representation, a semi-analytical tuning rule is developed. This rule is derived by drawing analogies between the various blocks in the RTD-A structure with the established concepts in process control literature. The performance of the proposed tuning rule is evaluated using a wide class of numerical examples, including delay dominant, higher order, and inverse response processes. It is shown that the RTD-A controllers tuned using the proposed rule provide reasonable performance for these processes with the performance being comparable to the proportional-integral (PI) controller tuned using Skogestad's Internal Model Control (SIMC) rules [121]. On the other hand, the RTD-A controller tuned by Ogunnaike's tuning rule provides aggressive response, which leads to instability, even for processes, which can be accurately represented by FOPTD model.

2.2 RTD-A Controller

In this section, a brief overview of RTD-A control scheme is given; see the work of Ogunnaike and Mukati [89] for further details. Subsequently, Ogunnaike's tuning rule for tuning the RTD-A controllers [90] are described.

2.2.1 RTD-A Algorithm

Figure 2.1: Schematic diagram of closed loop system with RTD-A controller

The block diagram of the closed-loop system with the RTD-A controller is shown in Figure 2.1, where y , u , d and y_{sp} denote the output, input, disturbance, and setpoint, respectively. In this algorithm, it is assumed that the process is stable, linear, and its behavior can be adequately described by an FOPTD model given as

$$G_M(s) = \frac{K_p}{\tau s + 1} \exp(-\alpha s) \quad (2.1)$$

where K_p , τ and α are the steady-state gain, time constant and time delay of the model, respectively. The continuous-time model can be discretized with sampling time ΔT to obtain the equivalent discrete-time representation of the FOPTD model given as

$$G_M(q^{-1}) = \frac{bq^{-(m+1)}}{1 - aq^{-1}} \quad (2.2)$$

where

$$a = \exp\left(-\frac{\Delta T}{\tau}\right); \quad b = K_p(1 - a); \quad m = \frac{\alpha}{\Delta T} \quad (2.3)$$

Note that in comparison to the work of Ogunnaike and Mukati [89], where m is taken as $m = \text{round}(\alpha/\Delta T)$, we assume that α is an integer multiple of ΔT . In the numerical examples considered in Section 2.5, this assumption is satisfied through appropriate selection of ΔT .

In the RTD-A algorithm, the input $u(k)$ is updated by minimizing the deviation between

the predicted process output $\hat{y}(k)$ and the reference trajectory $y^*(k)$ over the prediction horizon N . In particular, the following optimization problem is solved

$$\min_{u(k)} \sum_{i=1}^N (y^*(k+i) - \hat{y}(k+m+i))^2 \quad (2.4)$$

The reference trajectory $y^*(k)$ is determined as

$$y^*(k+1) = \theta_T y^*(k) + (1 - \theta_T) y_{sp}(k) \quad (2.5)$$

where θ_T is the tuning parameter for setpoint tracking. Assuming that the setpoint remains the same over the entire prediction horizon, *i.e.* $y_{sp}(k+i) = y_{sp}(k)$, $i = 1, 2, \dots, N$, the future reference trajectory can be expressed as

$$y^*(k+i) = \theta_T^i y^*(k) + (1 - \theta_T^i) y_{sp}(k); \quad i = 1, 2, \dots, N \quad (2.6)$$

Based on the process model in (2.2), the predicted output can be equivalently expressed by the following difference equation,

$$\hat{y}(k+1) = a\hat{y}(k) + bu(k-m) \quad (2.7)$$

The RTD-A algorithm assumes that the input remains the same for the next N steps, *i.e.* $u(k+i) = u(k)$; $i = 1, 2, \dots, N$. Thus, the predicted output can be written as

$$\hat{y}(k+m+i) = a^{m+i} \hat{y}(k) + a^{i-1} b \sum_{i=1}^m a^i u(k-i) + b\eta_i u(k) + \hat{e}_D(k+m+i); \quad i = 1, 2, \dots, N \quad (2.8)$$

where

$$\eta_i = \frac{1 - a^i}{1 - a} \quad (2.9)$$

In (2.8), $\hat{e}_D$ denotes the non-biasing prediction error, which is determined by the robustness tuning parameter θ_R as

$$\hat{e}_D(k+1) = \theta_R \hat{e}_D(k) + (1 - \theta_R)e(k) \quad (2.10)$$

where the current error $e(k)$ is the difference between the measured output $y(k)$ and its prediction $\hat{y}(k)$ obtained using (2.7), *i.e.* $e(k) = y(k) - \hat{y}(k)$. In addition, the future estimates of the disturbance are determined using the disturbance rejection tuning parameter θ_D as

$$\hat{e}_D(k+m+i) = \hat{e}_D(k) + \frac{1-\theta_D}{\theta_D} [1 - (1-\theta_D)^{m+i}] [\hat{e}_D(k) - \hat{e}_D(k-1)] \quad (2.11)$$

By solving the optimization problem in (2.4), the following explicit expression for $u(k)$ can be obtained,

$$u(k) = \frac{1}{b} \frac{\sum_{i=1}^N \eta_i (y^*(k+i) - a^{m+i} \hat{y}(k) - a^{i-1} b \sum_{i=1}^m a^i u(k-i) - \hat{e}_D(k+m+i))}{\sum_{i=1}^N \eta_i^2} \quad (2.12)$$

Equation (2.12) is the control law for the RTD-A controller, based on which the input $u(k)$ is updated at every time step.

2.2.2 Ogunnaike's Tuning Rule

In RTD-A algorithm, θ_R determines controller's ability to handle the plant-model mismatch, while θ_T and θ_D determine the future reference trajectory and disturbance prediction, respectively. The prediction horizon N is related to the overall aggressiveness tuning parameter θ_A as

$$N = 1 - \frac{\tau}{\Delta T} \ln(1 - \theta_A) \quad (2.13)$$

To select these tuning parameters, Ogunnaike *et al.* [90] proposed a set of generic rules.

This tuning rule is derived based on the robust stability analysis performed using hundreds of FOPTD models with different parameters and assumed plant-model mismatch. The tuning rule (referred to as Ogunnaike's tuning rule) is shown in Table 2.1. Here ρ is a composite variable defined as

$$\rho = \lambda \left(\frac{1 + \lambda}{1 - \lambda} \right) \frac{\alpha}{\tau} \quad (2.14)$$

where λ is the multiplicative uncertainty parameter. The recommended value for λ is 0.1 for processes, which can be closely represented by an FOPTD model and higher otherwise. In addition, Ogunnaike *et al.* [90] recommended that ΔT be chosen to be 0.1τ or lower. The RTD-A controllers tuned using this rule have been used for the control of quadruple tank process and vapor deposition process [90]. A shortcoming of this rule, however, is that it often leads to aggressive closed-loop behavior, see Section 2.5 for details. Furthermore, this rule only provide bounds on θ_R and θ_D , and an unambiguous selection of λ can be difficult in practice.

Table 2.1: Ogunnaike's RTD-A tuning rule

Composite Parameter	θ_R	θ_T	θ_D	θ_A
<i>Low noise processes:</i>				
$\rho < 1$	0.5	0.8	≥ 0.5	0.9ρ
$1 \leq \rho \leq 1.6$	0.9	0.8	≥ 0.1	$1.25(1 - e^{-\rho})$
$\rho \geq 1.6$	> 0.95	0.8	$\geq 1 - \theta_R$	0.995
<i>Noisy processes:</i>				
$\rho < 1$	0.9	0.8	≥ 0.37	0.9ρ
$1 \leq \rho \leq 1.6$	0.9	0.8	≥ 0.37	$1.25(1 - e^{-\rho})$
$\rho \geq 1.6$	> 0.95	0.8	$\geq 1 - 0.7\theta_R$	0.995

2.3 Block Diagram Representation

The derivation of a simple tuning rule for the RTD-A controller requires a proper

understanding of the role of each tuning parameter. For this purpose, a block diagram representation of the RTD-A controller is developed in this section.

2.3.1 Derivation

The central idea of the block diagram representation is to simplify the different terms in the control law given in (2.12) and express them in terms of setpoint $y_{sp}(k)$ and error $e(k)$. First, we note that based on (2.5), the expression for $y^*(k+1)$ can be equivalently written in transfer function form as

$$y^*(k+1) = \frac{1 - \theta_T}{1 - \theta_T q^{-1}} y_{sp}(k) \quad (2.15)$$

The following expression for $y^*(k+i)$ is found by substituting (2.15) in (2.6)

$$y^*(k+i) = G_{T,i}(q^{-1}) y_{sp}(k); \quad i = 1, 2, \dots, N \quad (2.16)$$

where

$$G_{T,i}(q^{-1}) = \theta_T^i \left[\frac{1 - \theta_T}{1 - \theta_T q^{-1}} \right] + (1 - \theta_T^i) \quad (2.17)$$

Similar to (2.15), the expression for current estimated error $\hat{e}_D$ can also be written in transfer function form as

$$\hat{e}_D(k) = G_R(q^{-1}) e(k) \quad (2.18)$$

where

$$G_R(q^{-1}) = \frac{1 - \theta_R}{1 - \theta_R q^{-1}} \quad (2.19)$$

As before, the following expression of $\hat{e}_D(k+m+i)$ is obtained by substituting (2.18) in (2.11)

$$\hat{e}_D(k+m+i) = G_R(q^{-1}) G_{D,i}(q^{-1}) e(k); \quad i = 1, 2, \dots, N \quad (2.20)$$

where

$$G_{D,i}(q^{-1}) = 1 + \frac{1 - \theta_D}{\theta_D} [1 - (1 - \theta_D)^{m+i}] (1 - q^{-1}) \quad (2.21)$$

We note that

$$\sum_{i=1}^m a^i u(k-i) = (aq^{-1} + a^2 q^{-2} + \dots + a^m q^{-m}) u(k) \quad (2.22)$$

$$= \frac{aq^{-1} [1 - (aq^{-1})^m]}{1 - aq^{-1}} u(k) \quad (2.23)$$

where the last expression has been obtained by summing the geometric series.

With these preliminaries and some straightforward algebra, the control law in (2.12) can be written as

$$\left(b \sum_{i=1}^N \eta_i^2 \right) u(k) = \sum_{i=1}^N \eta_i \left(G_{T,i}(q^{-1}) y_{sp}(k) - \frac{ba^i q^{-1}}{1 - aq^{-1}} u(k) - G_R(q^{-1}) G_{D,i}(q^{-1}) e(k) \right) \quad (2.24)$$

Now, (2.24) can be simplified as

$$u(k) = G_C(q^{-1}) \sum_{i=1}^N \eta_i (G_{T,i}(q^{-1}) y_{sp}(k) - G_R(q^{-1}) G_{D,i}(q^{-1}) e(k)) \quad (2.25)$$

where

$$G_C(q^{-1}) = \frac{1}{b \sum_{i=1}^N \eta_i \left[\eta_i + \left(\frac{a^i q^{-1}}{1 - aq^{-1}} \right) \right]} \quad (2.26)$$

Figure 2.2: Schematic diagram of Internal Model Controller.

The closed-loop block diagram for the RTD-A controller, derived based on (2.25), is

shown in Figure 2.3, where $G_P(q^{-1})$ represents the process. It can be noted that the block diagram resembles the IMC structure [83], where a model is used for output prediction and the difference between the actual output and its prediction is fed back to the controller; see Figure 2.2.

In the subsequent discussion, each block on RTD-A controller structure is analyzed to understand its role in the overall controller structure.

Figure 2.3: Closed-loop block diagram of RTD-A controller

2.3.2 Analysis

Based on Figure 2.3, we note that $G_T(q^{-1})$ acts as a first order setpoint filter for the RTD-A controller. Similar setpoint filters are also used in MPC [132] and Model Predictive Heuristic Control (MPHC) [79]. The $G_R(q^{-1})$ plays the role of a first order noise filter. For the IMC structure, it is well-known that the noise filter can increase the controller robustness without affecting overall control performance significantly [12, 83] and is also helpful in attenuating the oscillations in the plant output [87].

To understand the role of $G_C(q^{-1})$, let the model in (2.2) be partitioned as

$$G_M(q^{-1}) = q^{-(m+1)} \bar{G}_M(q^{-1}) \quad (2.27)$$

where $\bar{G}_M = b/(1 - aq^{-1})$ is the invertible part of the model. Now, $G_C(q^{-1})$ in (2.26) can

be expressed as

$$G_C(q^{-1}) = \bar{G}_M^{-1} \frac{1}{\sum_{i=1}^N \eta_i (\eta_i (1 - aq^{-1}) + a^i q^{-1})} = \bar{G}_M^{-1} \frac{1 / \sum_{i=1}^N \eta_i^2}{1 - \left(a - \frac{\sum_{i=1}^N a^i \eta_i}{\sum_{i=1}^N \eta_i^2} \right) q^{-1}} \quad (2.28)$$

Thus, $G_C(q^{-1})$ contains an inverse of the invertible part of the model augmented with a first order filter. The first order filter is the result of multi-step prediction used in the RTD-A controller. This can be confirmed by noting that for $N = 1$, the pole of the filter is at $q = 0$ and thus $G_C(q^{-1}) = \bar{G}_M^{-1}$.

In RTD-A algorithm, $G_D(q^{-1})$ allows the prediction of the effect of disturbance on the process output. A related concept used earlier in process control literature is the GAP [141, 142]. The GAP generalized the analytical predictor scheme [82] and provides an alternative to the use of Smith predictor [48, 125] for deadtime compensation. In the GAP, for an FOPTD model of the process, the predicted value of load disturbance at time $k + p$, *i.e.* $\hat{d}(k + p)$ is given as [141].

$$\hat{d}(k + p) = A(q^{-1})e(k) \quad (2.29)$$

where

$$A(q^{-1}) = a^p + \frac{1 - a^p}{1 - a} (1 - aq^{-1}) = 1 + \frac{1 - a}{a} (1 - a^p) (1 - q^{-1}) \quad (2.30)$$

Based on (2.30), it can be noted that $G_{D,i}(q^{-1})$ in (2.21) can be interpreted as the GAP for $i = 1, 2, \dots, N$ with $\theta_D = 1 - a$ and $p = m + i$. Overall, the RTD-A controller can be interpreted as the GAP augmented with noise and setpoint filters. The main distinguishing feature between the GAP and RTD-A is that the available GAP schemes [141, 142] use single-step prediction ($N = 1$), while the RTD-A employs multi-step ahead prediction.

To gain further insight into the RTD-A algorithm, the following closed-loop transfer

function of the RTD-A controller is derived as

$$y(k) = G_{sp}(q^{-1})y_{sp}(k) + G_d(q^{-1})d(k) \quad (2.31)$$

where $G_{sp}(q^{-1})$ and $G_d(q^{-1})$ are the closed-loop setpoint and disturbance transfer functions, respectively, and are given as

$$G_{sp}(q^{-1}) = \frac{G_P(q^{-1})G_C(q^{-1})\eta G_T(q^{-1})}{1 + \eta G_D(q^{-1})G_R(q^{-1})G_C(q^{-1})(G_P(q^{-1}) - G_M(q^{-1}))} \quad (2.32)$$

$$G_d(q^{-1}) = \frac{G_P(q^{-1})(1 - \eta G_D(q^{-1})G_R(q^{-1})G_C(q^{-1})G_M(q^{-1}))}{1 + \eta G_D(q^{-1})G_R(q^{-1})G_C(q^{-1})(G_P(q^{-1}) - G_M(q^{-1}))} \quad (2.33)$$

Based on (2.31), the following observations are made:

1. In absence of plant-model mismatch, *i.e.* $G_P(q^{-1}) = G_M(q^{-1})$, $G_{sp}(q^{-1})$ depends on θ_T and θ_A , while $G_d(q^{-1})$ depends on θ_A , θ_R and θ_D . Hence, θ_T and θ_D can be tuned independently for setpoint tracking and disturbance rejection, respectively [89].
2. The characteristic polynomial for the RTD-A controller is a function of $G_D(q^{-1})$ and $G_R(q^{-1})$ and thus θ_D and θ_R . Based on closed-loop simulations, it is seen that changing θ_D usually has limited effect on controller robustness. This observation partially justifies the claim that RTD-A controller can be tuned independently for disturbance rejection and robustness [89, 90], although this claim does not seem to hold in general.
3. The characteristic polynomial does not depend on θ_T and thus the stability of RTD-A controller is independent of the choice of θ_T . This can also be noted from Figure 2.3, where the setpoint filter $G_T(q^{-1})$ lies outside the feedback loop. Ogunnaike and Mukati [90] reached the same conclusion based on state-space analysis.

4. By defining the compensator $C(q^{-1})$ as

$$C(q^{-1}) = \frac{G_C(q^{-1})}{1 - \eta G_D(q^{-1})G_R(q^{-1})G_C(q^{-1})G_M(q^{-1})} \quad (2.34)$$

the expression for the closed-loop transfer function simplifies as

$$y(k) = \frac{G_P(q^{-1})C(q^{-1})\eta G_T(q^{-1})}{1 + \eta G_D(q^{-1})G_R(q^{-1})C(q^{-1})G_P(q^{-1})} y_{sp}(k) + \frac{G_P(q^{-1})}{1 + \eta G_D(q^{-1})G_R(q^{-1})C(q^{-1})G_P(q^{-1})} d(k) \quad (2.35)$$

Figure 2.4: Equivalent closed-loop block diagram for RTD-A algorithm

The resulting block diagram is shown in Figure 2.4, which resembles the standard feedback loop more closely than the block diagram shown in Figure 2.3. Furthermore, for $|a| < 1$, since $G_{D,i}(1) = G_R(1) = 1$ and $G_C(1) = (1/\sum_{i=1}^N \eta_i) \bar{G}_M(1)^{-1}$, it follows that $\eta G_D(1)G_R(1)G_C(1)G_M(1) = 1$. This shows that $C(q^{-1})$ contains a pole at $q = 1$. The presence of integral action in the RTD-A controller guarantees zero offset.

5. For $N = 1$, $\eta_1 = 1$ and $G_C(q^{-1}) = \bar{G}_M(q^{-1})$. Under no plant-plant mismatch, with $\theta_D = 1 - a$, conceptually $G_D(q^{-1})$ can be viewed as the inverse of process delay [141].

$$G_D(q^{-1}) = q^{(m+1)} \quad (2.36)$$

$$= G_M(q^{-1})(1 - G_R(q^{-1})) = G_M(q^{-1}) \frac{(1 - \theta_R)}{(1 - \theta_R q^{-1})} \quad (2.37)$$

Thus, by choosing $\theta_R = 0$, irrespective of the process delay, deadbeat response can be obtained using RTD-A controller. On the other hand, the achievable closed-loop performance for PID controller is limited by the process delay and controller structure [1, 65, 110].

2.4 Tuning Rule for RTD-A Controller

For a given process, the optimal values of the tuning parameters of the RTD-A controller can be found by minimizing an objective function related to the closed-loop performance. For example, Sreenivas *et al.* [151] minimized integral absolute error (IAE) to find the values of θ_A , θ_T , θ_D and θ_R . Such an approach, however, is computationally intensive and is difficult to apply. In this section, we present a semi-analytical tuning rule for the RTD-A controller, which are simple and easy to remember. This rule is derived by relating the different blocks of RTD-A controller with established concepts in process control literature. Although not optimal for a given process, this rule is applicable to a wide class of processes, as shown in the next section.

2.4.1 Overall Aggressiveness Parameter θ_A

Based on (2.13), θ_A is directly related to the prediction horizon N , which determines how far into the future the output will be predicted. Brosilow and Joseph [12] recommended that the prediction horizon and the sampling time be chosen as $N\Delta T = t_{ss}$, where t_{ss} is the time to reach steady state. For an FOPTD model, as used by the RTD-A algorithm, t_{ss} can be approximately taken as $4\tau/\Delta T + m$. Since the output prediction in (2.8) already

accounts for m , a reasonable choice for N seems to be

$$N = 4\tau/\Delta T \quad (2.38)$$

However, for $\Delta T = 0.1\tau$ (recommended value), (2.38) results in $\theta_A = 0.98$ making the controller very conservative. Furthermore, with N being so large, changing θ_T and θ_D has little effect on the setpoint and disturbance responses, respectively. To illustrate this, we note that $\eta G_T(q^{-1})$ can be expressed as

$$\eta G_T(q^{-1}) = \frac{\sum_{i=1}^N \eta_i (1 - \theta_T^{i+1})}{1 - \theta_T q^{-1}} (1 - \theta_T \alpha q^{-1}); \quad \alpha = \frac{\sum_{i=1}^N \eta_i (1 - \theta_T^i)}{\sum_{i=1}^N \eta_i (1 - \theta_T^{i+1})} \quad (2.39)$$

As $N \rightarrow \infty$, $\alpha \rightarrow 1$. Thus, for large values of N , $\eta G_T(q^{-1})$ has a zero at $q \approx \theta_T$, which nearly cancels the pole at $q = \theta_T$ causing the closed-loop setpoint transfer function to be practically independent of θ_T . Similarly, it can be shown that for large N , the zero of $\eta G_D(q^{-1})$ approaches 0.5, irrespective of the values of the parameters of FOPTD model, and thus the closed-loop disturbance response is not significantly affected by different choices of θ_D .

The earlier discussion highlights that N should not be chosen to be very large. On the other hand, for low values of N , *e.g.* $N = 1$, RTD-A results in very aggressive response; see Section 2.3.2 for details. Based on these observations, we recommend the use of moderate value of N , *i.e.* $N = \tau/\Delta T$. Thus, the proposed value for θ_A is

$$\theta_A = 1 - \exp\left(-\frac{\Delta T}{\tau}(N - 1)\right) = 1 - \frac{1}{ae} \quad (2.40)$$

where e denotes Euler's number.

2.4.2 Setpoint Tracking Parameter θ_T

In MPC and MHPC, the following first order setpoint filter is used frequently [79, 132]:

$$y^*(k+i|k) = \gamma y^*(k+i-1) + (1-\gamma)y_{sp}(k+p); \quad i = 1, 2, \dots, N \quad (2.41)$$

where $0 \leq \gamma \leq 1$. Here, γ is defined as [78]

$$\gamma = \exp\left(\frac{-\Delta T}{T_{ref}}\right) \quad (2.42)$$

where T_{ref} is the time constant of the reference trajectory. In the RTD-A algorithm, the role of θ_T is identical to the role of γ in MPC and MHPC. By setting T_{ref} as half of the process time constant *i.e.* $T_{ref} = 0.5\tau$, θ_T is obtained as

$$\theta_T = \exp\left(\frac{-2\Delta T}{\tau}\right) = a^2 \quad (2.43)$$

This choice of θ_T implies that under closed-loop conditions, the process output should reach the setpoint twice as fast in comparison to the open-loop conditions. Note that when ΔT is chosen as 0.1τ (recommended value), $\theta_T = 0.82$. In comparison, Ogunnaike and Mukati [90] suggested choosing θ_T as 0.8. In this sense, the proposed and Ogunnaike's tuning rule lead to similar choices of θ_T .

2.4.3 Disturbance Rejection Parameter θ_D

In the previous section, it was shown that $G_{D,i}(q^{-1})$ has the same structure as the GAP. Under no plant-model mismatch, for step changes in disturbances entering through input channel [141], when θ_D is selected as

$$\theta_D = 1 - \left[\exp\left(-\frac{\Delta T}{\tau}\right) \right] = 1 - a \quad (2.44)$$

$G_D(q^{-1})$ provide perfect predictions of the load disturbance and thus we recommend this choice.

2.4.4 Robustness Parameter θ_R

The tuning of θ_R determines the robustness of the closed loop system. Assuming that

$$G_P(z^{-1}) = \frac{\epsilon b z^{-\bar{m}+1}}{1 - \bar{a} z^{-1}} \quad (2.45)$$

with $m \geq \bar{m}$, the following condition for stability of the closed-loop system is derived; see Section 2.A for proof:

$$\theta_R \geq 1 - \frac{\sum_{i=1}^N (1 - a^i)^2}{\epsilon a^2 \sum_{i=1}^N (1 - a^i)(1 - a^{m+i})} \quad (2.46)$$

Here, ϵ can be seen as the expected gain margin. The condition in (2.46), however, is only necessary for the closed-loop stability and thus is difficult to be used directly. It is noted that for given N and a , the lower bound on θ_R in (2.46) monotonically increases with m for low values of m and remains nearly constant for large values of m . These features are also shown by the following relationship:

$$\theta_R = \frac{m}{1 + m} \quad (2.47)$$

which is an upper bound on the expression on the right hand side of (2.46) for typical values of ϵ, a and N , *i.e.* $\epsilon = 2, a = 0.9048$ and $N = 10$. With the selection of θ_R based on (2.47), the disturbance rejection becomes very slow for very large value of m (when $\theta_R > 0.95$). Thus, the following modified tuning rule is instead proposed:

$$\theta_R = \min \left[\frac{m}{1 + m}, 0.95 \right] \quad (2.48)$$

Note that the minimum value of m is 1 due to zero-order hold and θ_R is always chosen to be larger than 0.5.

The proposed tuning rule for RTD-A controller is summarized in Table 2.2.

Table 2.2: Proposed RTD-A tuning rule

θ_R	θ_T	θ_D	θ_A
$\min \left[\frac{m}{m+1}, 0.95 \right]$	a^2	$1 - a$	$1 - \frac{1}{ae}$

2.5 Evaluation of Proposed Tuning Rule

This section presents several examples to show the effectiveness and reliability of the proposed tuning rule for the RTD-A controller. The output and input performances are evaluated using IAE and total variation (TV), respectively, while the robustness is assessed using gain and delay margins. The closed-loop performance is compared with the RTD-A controller tuned using Ogunnaike's tuning rule [90] and the PI controller.

For the design of RTD-A controller using Ogunnaike's tuning rule, the processes are assumed to have low noise. For processes, whose step response can be closely approximated by the step response of an FOPTD process, λ is chosen as 0.1 [90]. In the case of significant deviations from the step response of an FOPTD process, a higher value of λ is needed to handle the plant-model mismatch, but no guidelines are available for the selection of λ . In this work, we use $\lambda = 0.3$ for non-FOPTD processes. Further, it is noted that Ogunnaike's tuning rule only provides bounds on θ_D and θ_R . To avoid ambiguity, these parameters are set to their lower bounds in this work.

For tuning of PI controllers, numerous methods are available in the literature including Ziegler-Nichols [155], IMC [104], MacLaurin-IMC [73], SIMC [121], and magnitude optimum [138]. Recently, Lin *et al.* [75] evaluated various tuning rules and showed that

SIMC method works best for a wide range of processes. Therefore, SIMC tuning rule are used for PI controller design in this work. For FOPTD model in (2.1), the PI controller parameters based on SIMC tuning rule are [121]

$$K_c = \frac{\tau}{K_p(\tau_c + \alpha')}; \quad \tau_I = \min(\tau, 4(\tau_c + \alpha')) \quad (2.49)$$

where K_c , τ_I , and τ_c denote the controller gain, integral time, and desired closed-loop time constant, respectively, and $\alpha' = \alpha + \Delta T/2$. The recommended value of τ_c is $\tau_c = \alpha'$ [121].

To facilitate the design of RTD-A and PI controllers, an FOPTD model is identified using fraction incomplete method [131]. As recommended by Ogunnaike and Mukati [89], the sampling time ΔT is chosen to be 0.1τ or lower, such that α is an integral multiple of ΔT . The process, FOPTD models and chosen sampling time are shown in Table 2.3, while the controller parameters are shown in Table 2.4. The closed-loop simulation is carried out for 100s, where unit step changes in setpoint and disturbance are introduced at $t = 0$ and $t = 50s$, respectively. Table 2.5, Table 2.6, and Table 2.7 show the performance and robustness comparison of the proposed RTD-A, Ogunnaike's RTD-A, and SIMC-PI controllers.

First order processes. The proposed RTD-A tuning rule is implemented on numerous FOPTD processes, which can be classified into normal, lag dominant and lead dominant processes based on the ratio between α and τ . For the FOPTD process with $\alpha/\tau \approx 1$, such as Case 1, Ogunnaike's tuning rule provide slightly better output performance for both setpoint tracking and disturbance rejection than the proposed RTD-A and SIMC-PI controllers; see Figure 2.5 and Table 2.5. The RTD-A controller tuned using Ogunnaike's tuning rule, however, lead to a very large change in input upon setpoint change, highlighting the aggressive nature of the controller.

For processes with $\alpha/\tau \ll 1$ (lag dominant process), such as Case 2, the performances of all the controllers are comparable for the disturbance rejection. However, for the setpoint

Table 2.3: Case studies and their first order approximation

Case	$G_P(s)$	$G_M(s)$	ΔT
1	$\frac{e^{-5s}}{10s+1}$	$\frac{e^{-5s}}{10s+1}$	1
2	$\frac{e^{-s}}{20s+1}$	$\frac{e^{-s}}{20s+1}$	1
3	$\frac{e^{-10s}}{2s+1}$	$\frac{e^{-10s}}{2s+1}$	0.2
4	$\frac{1}{(20s+1)(s+1)}$	$\frac{e^{-s}}{20s+1}$	1
5	$\frac{1}{(5s+1)^2}$	$\frac{e^{-3s}}{7.27s+1}$	1
6	$\frac{(-s+1)e^{-s}}{(6s+1)(2s+1)^2}$	$\frac{e^{-5.5s}}{6.56s+1}$	0.5
7	$\frac{-2s+1}{(s+1)^3}$	$\frac{e^{-3s}}{1.63s+1}$	0.2
8	$\frac{(15s+1)e^{-4s}}{(8s+1)(5s+1)}$	$\frac{e^{-3.4s}}{1.64s+1}$	0.2
9	$\frac{(2s+1)e^{-5s}}{(10s+1)(0.5s+1)}$	$\frac{e^{-3.3s}}{10s+1}$	0.3
10	$\frac{(2s+1)e^{-2s}}{(10s+1)(0.5s+1)}$	$\frac{e^{-0.3s}}{10s+1}$	0.3

change, the RTD-A controller tuned using the proposed rule provides sluggish performance and the IAE and TV values are significantly larger in comparison with other controllers; see Figure 2.5, Table 2.5, and Table 2.6. The sluggish response is due to the fact that the proposed tuning rule aims at achieving closed-loop setpoint response, which is twice as fast as the open-loop response and for Case 2, τ is large. The performance of RTD-A controller can be improved by using more aggressive settings, *i.e.* by reducing θ_A or θ_T (details not shown).

The conclusions are reversed when $\alpha/\tau \gg 1$ (delay dominant process), such as Case 3. For these processes, both proposed and Ogunnaike's RTD-A controllers show superior performance compared to the SIMC-PI controller. It can be seen from Figure 2.7 that SIMC-PI controller gives significantly more sluggish response for both setpoint and disturbance changes than the RTD-A controllers. While for the lag-dominant process (Case 2), the performance of the RTD-A controller tuned using the proposed rule can be improved by controller re-tuning, reducing τ_c does not significantly improve the closed-loop performance of SIMC-PI controller, as large time delay poses fundamental limitation

Figure 2.5: Closed-loop simulation for Case 1

Figure 2.6: Closed-loop simulation for Case 2

Table 2.4: RTD-A/PI controller parameters

Case	Controller Parameters		
	RTD-A (Proposed)	RTD-A(Ogunnaike) [90]	SIMC-PI [121]
	$\theta_R, \theta_T, \theta_D, \theta_A$	$\theta_R, \theta_T, \theta_D, \theta_A$	K_c, τ_I
1	0.83, 0.82, 0.10, 0.59	0.50, 0.80, 0.50, 0.06	0.91, 10.00
2	0.50, 0.90, 0.05, 0.61	0.50, 0.80, 0.50, 0.01	6.67, 12.00
3	0.95, 0.82, 0.10, 0.59	0.50, 0.80, 0.50, 0.55	0.10, 2.00
4	0.50, 0.90, 0.05, 0.61	0.50, 0.80, 0.50, 0.01	6.67, 12.00
5	0.75, 0.76, 0.13, 0.58	0.50, 0.80, 0.50, 0.21	1.04, 7.27
6	0.92, 0.86, 0.07, 0.60	0.50, 0.80, 0.50, 0.42	0.57, 6.56
7	0.94, 0.78, 0.12, 0.58	0.90, 0.80, 0.10, 0.80	0.26, 1.63
8	0.94, 0.78, 0.11, 0.58	0.90, 0.80, 0.10, 0.86	0.23, 1.64
9	0.92, 0.94, 0.03, 0.62	0.50, 0.80, 0.50, 0.04	1.45, 10.00
10	0.50, 0.94, 0.03, 0.62	0.50, 0.80, 0.50, 0.003	11.10, 3.60

on the achievable performance of PI controllers [1, 65, 110].

Figure 2.7: Closed-loop simulation for Case 3

Higher order processes. For evaluation of the tuning rules for higher order processes, two processes are considered, *i.e.* a second order process with dominant pole (Case 4) and a second order process with repeated pole (Case 5). When the RTD-A controller is tuned using Ogunnaike's rules, the presence of plant-model mismatch causes the closed-loop response to be oscillatory for Case 4. Even though the process can be accurately represented by an FOPTD model, small value of α/τ leads to an aggressive controller.

Table 2.5: IAE of RTD-A and SIMC-PI controllers for setpoint tracking (SP) and disturbance rejection (D)

Case	RTD-A (Proposed)		RTD-A (Ogunnaike)		SIMC-PI	
	SP	D	SP	D	SP	D
1	12.31	8.38	10.37	6.50	13.04	10.85
2	14.27	2.04	6.20	2.05	4.92	1.77
3	12.26	12.90	12.16	10.93	22.44	21.19
4	14.27	2.04	6.20	2.05	4.97	1.77
5	8.75	5.23	8.55	5.09	9.23	6.97
6	10.75	9.30	9.83	7.79	13.79	11.72
7	5.50	6.69	6.97	6.09	8.03	7.96
8	7.62	7.70	7.53	6.61	10.55	11.19
9	10.77	5.22	—	—	9.28	6.81
10	—	—	—	—	—	—

On the other hand, the proposed tuning rule gives smooth closed-loop responses with reasonable gain and delay margins; see Figure 2.8 and Table 2.7. For the disturbance change, the responses of the RTD-A controller tuned using proposed rule and the SIMC-PI controllers are nearly identical, while the former controller shows sluggish response for setpoint change. Similar to Case 2, the performance of the RTD-A controller tuned using proposed rule can be improved easily for setpoint tracking by reducing θ_A or θ_T .

Figure 2.8: Closed-loop simulation for Case 4

Table 2.6: TV of RTD-A and SIMC-PI controllers for setpoint tracking (SP) and disturbance rejection (D)

Case	RTD-A (Proposed)		RTD-A (Ogunnaike)		SIMC-PI	
	SP	D	SP	D	SP	D
1	2.20	1.02	4.86	1.00	1.77	1.08
2	2.10	1.11	13.76	1.00	13.90	1.24
3	11.00	5.00	12.30	5.00	5.31	5.26
4	3.03	1.15	109.66	5.89	12.91	1.25
5	3.50	1.14	5.48	1.11	1.77	1.14
6	5.81	2.12	11.25	2.27	2.63	2.24
7	19.37	8.88	33.52	23.21	5.82	7.16
8	15.89	6.58	13.20	7.81	7.08	8.35
9	27.28	13.07	—	—	11.96	4.21
10	—	—	—	—	—	—

The performance of both RTD-A controllers are comparable for Case 5, where the process has no dominating pole. For this case, the SIMC-PI controller gives similar closed-loop output response as both the RTD-A controllers, although the RTD-A controllers require somewhat larger input variation than the SIMC-PI controller. In the context of GAP, Wellons and Edgar [141] have pointed out that significant error in FOPTD model parameter a can cause the input response to be oscillatory for setpoint changes. Increasing θ_R can reduce the oscillations in the input for RTD-A controllers, although at the expense of sluggish output response [141].

Processes with unstable zeros. The presence of unstable zeros can pose fundamental limitations on the achievable performance of feedback controllers [115, 124]. To evaluate the performance of RTD-A controller for processes with unstable zeros, we consider two processes. For Case 6, the magnitude of the unstable zero is much larger than the magnitude of the dominating pole and thus only a small inverse response is observed in the step response of the process. In this case, both the RTD-A controllers provide faster closed-loop response than the SIMC-PI controller (see Figure 2.10 and Table 2.5) with reasonable gain and delay margins (see Table 2.7).

Figure 2.9: Closed-loop simulation for Case 5

Figure 2.10: Closed-loop simulation for Case 6

Table 2.7: Robustness properties of RTD-A and SIMC-PI controllers

Case	Gain Margin			Delay Margin		
	RTD-A (Proposed)	RTD-A (Ogunnaike)	SIMC-PI	RTD-A (Proposed)	RTD-A (Ogunnaike)	SIMC-PI
1	2.63	2.10	3.13	9	6	11
2	2.80	2.17	2.94	2	2	2
3	2.29	2.03	3.14	67	54	540
4	11.19	2.51	6.87	3	2	3
5	10.40	7.23	13.29	6	6	7
6	3.97	3.72	3.69	21	16	49
7	1.99	1.46	2.41	27	17	146
8	3.53	2.86	2.71	22	15	111
9	1.36	—	2.19	16	—	74
10	—	—	—	—	—	—

For Case 7, a large inverse response occurs in the step response of the process, as the magnitude of the unstable zeros is half of the magnitude of the repeated pole. In this case, Ogunnaike's tuning rule leads to oscillatory closed-loop response (see Figure 2.11) with the gain margin being less than 2 (see Table 2.7). The response of the RTD-A controller tuned using proposed rule is acceptable, especially for the disturbance rejection. An apparent reason for the ability of the proposed RTD-A controller to handle the model mismatch arising due to the presence of unstable zeros is that the inverse response is approximated as time delay, which consequently leads to a more conservative choice of θ_R .

Process with stable zeros. Finally, we consider the effect of stable zeros on the RTD-A controller. Depending on the location of the zeros with respect to the poles, these processes can exhibit overshoot (Case 8) or behave essentially like a first order process (Cases 9 and 10) [107]. The open-loop step response of Case 8 shows a mild overshoot of 20%. Despite the large plant-model mismatch, both RTD-A controllers provide good responses and perform slightly better than the SIMC-PI controller.

For Case 9, the step response of the response of the process can be reasonably well approximated using an FOPTD model. However, for this process, the fraction incomplete

Figure 2.11: Closed-loop simulation for Case 7

Figure 2.12: Closed-loop simulation for Case 8

method underapproximates the delay of the process, which leads to plant-model mismatch at high frequencies giving rise to robustness issues. Ogunnaike's tuning rule results in unstable closed-loop responses. The RTD-A controller tuned using proposed rule shows reasonable output responses, but oscillations with low amplitude are seen in the input with gain margin being less than 2. The performance of the proposed RTD-A controller can be slightly improved by increasing θ_R to 0.9.

Figure 2.13: Closed-loop simulation for Case 9

For Case 10, the process is identical to Case 9, except that the delay is lower. For this process, none of the controllers are able to provide a stable closed-loop response with the use of suggested tuning parameters. We note that instead of using the fraction incomplete method, when the FOPTD model is obtained using the model reduction approach proposed by Skogestad [121], the SIMC-PI controller provides excellent closed-loop responses. It seems that this model order reduction approach [121] is tailored to the SIMC tuning rules, especially for processes with stable zeros. In comparison with the use of FOPTD model obtained using fraction incomplete method, it is expected that the identification of a higher order model using system identification techniques [76] and subsequent model order reduction using a customized method for RTD-A controllers can allow achieving better closed-loop properties.

2.6 Chapter Summary

In this chapter, the recently proposed RTD-A controller is analyzed by developing a block-diagram representation and a semi-analytical tuning rule is proposed. Various numerical examples are used to show that the proposed tuning rule can provide good performance for a wide range of processes. In comparison, RTD-A controllers obtained using Ogunnaike's tuning rule [90] are aggressive, which can lead to instability, even for processes, which can be closely represented by FOPTD model. For most of the numerical examples, the performances of the RTD-A controller tuned using the proposed rules and PI controller tuned using SIMC rule [121] are found to be similar. For the different processes considered in this chapter, the development of simple tuning rules to significantly improve the performance of the RTD-A controller without sacrificing robustness seems difficult. In this sense, the distinct advantage of RTD-A controller over conventional PI controller is the ease with which the controller can be tuned online.

2.A Appendix: Lower Bound on θ_R

The characteristic polynomial of RTD-A controller can be written as

$$P(z^{-1}) = 1 + \eta G_D(z^{-1}) G_R(z^{-1}) G_C(z^{-1}) (G_P(z^{-1}) - G_M(z^{-1})) \quad (2.50)$$

where η , $G_D(z^{-1})$, $G_R(z^{-1})$, $G_C(z^{-1})$, and $G_M(z^{-1})$ are given in (2.9), (2.21), (2.19), (2.28), and (2.2), respectively. For $G_P(z^{-1})$ in (2.45) with $\bar{m} > m$ and $\theta_D = 1 - a$, the expression for $P(z^{-1})$ can be simplified as

$$P(z^{-1}) = (1 - \theta_R z^{-1})(1 - \bar{a} z^{-1}) \left(\sum_{i=1}^N \eta_i^2 (1 - a z^{-1}) + \left(\sum_{i=1}^N a^i \eta_i \right) z^{-1} \right)$$

$$+ \sum_{i=1}^N \eta_i (1 + \beta_i (1 - z^{-1})) (1 - \theta_R) (\epsilon (1 - az^{-1}) z^{-(\bar{m}+1)} - (1 - \bar{a}z^{-1}) z^{-(m+1)}) \quad (2.51)$$

where

$$\beta_i = \frac{a}{1-a} (1 - a^{m+i}); \quad i = 1, 2, \dots, N \quad (2.52)$$

Based on Jury's stability criterion [56, 91], a necessary condition for all the roots of the q^{th} order polynomial $\sum_{i=0}^q c_0 z^i$ to lie inside the unit circle is that $|c_0| < c_q$. For the characteristic polynomial in (2.51),

$$c_0 = -\epsilon a (1 - \theta_R) \sum_{i=1}^N \eta_i \beta_i \quad (2.53)$$

$$c_q = \sum_{i=1}^N \eta_i^2 \quad (2.54)$$

and thus, a necessary condition for closed-loop stability is

$$\theta_R \geq 1 - \frac{\sum_{i=1}^N \eta_i^2}{\epsilon a \sum_{i=1}^N \eta_i \beta_i} \quad (2.55)$$

Now, the lower bound on θ_R in (2.46) follows by substituting for η_i and β_i and simplifying the resulting expression.

Chapter 3

Achievable PID Performance

3.1 Introduction

For stable linear discrete time systems with time delay, which have no finite zeros on or outside the unit circle, minimum variance (MV) benchmark denotes the least achievable output variance [45]. When the controller is restricted to be proportional integral derivative (PID) controller, however, the MV benchmark may not be achievable. In other words, they may not exist a PID controller which can match the output variance indicated by the MV benchmark. Despite the wide usage of PID controllers in process industries, characterization of the limitations arising due to the controller structure has received limited attention.

The key difficulty in characterizing the achievable performance for a restricted structure controller is the non-convexity of the resulting optimization problem. During the past few years, some researchers have suggested the use of gradient based methods [1, 65, 68]

¹Part of this chapter was presented in 14th Regional Symposium of Chemical Engineering, Yogyakarta, Indonesia, 2007 [109].

²The contents of this chapter have been published in Journal of Process Control, 2009 [110].

and stochastic optimization [32] to solve the non-convex optimization problem. These methods do not guarantee the global optimality and hence can only provide upper bounds on the achievable PID performance. Furthermore, for analyzing fundamental limitations arising due to the controller structure, it is more useful to derive a lower bound. Recently, Kariwala [57] proposed an analytical lower bound on the achievable PID performance by considering the first $(2m - 1)$ impulse response coefficients of the closed-loop transfer function between disturbance and output, where m denotes the time delay. Clearly, the lower bound in [57] can be loose in general due to the neglected impulse response coefficients.

In this chapter, we show that the impulse response coefficients of the closed-loop transfer function between disturbance and output can be represented as polynomials in unknown controller parameters. Subsequently, the non-convex optimization problem related to minimization of output variance is solved using sums of squares (SOS) programming, which guarantees a lower bound on the global optimal solution [94]. The SOS programming technique also provides sub-optimal tuning parameters of the PID controller, which can be used to compute a tight upper bound on the achievable PID performance.

The minimization of output variance only may lead to excessive input variation. To overcome this difficulty, we extend the SOS programming approach to find bounds on the weighted sum of the variances of input rate and output. Similar to the linear quadratic gaussian (LQG) benchmark for unrestricted controllers [45], this result can be used to find the trade-off between the variances of input rate and output for PID control. We demonstrate the tightness of proposed bounds using several benchmark examples drawn from literature.

3.2 Sums of Squares Programming

In this section, we provide a brief overview of the SOS programming technique for polynomial minimization; see [94] for further details. A polynomial $r(x)$ is called SOS if it can be decomposed as

$$r(x) = \sum_{i=1}^n f_i^2(x) \quad (3.1)$$

where $f_i(x)$ are polynomials in x . Clearly for $r(x)$ to be SOS, its order should be even. When a polynomial $r(x)$ is SOS, it means that the minimum value of $r(x)$ is greater than or equal to zero over all possible values of x .

The potential of SOS decomposition for solving polynomial minimization problem has been known for long, but its application to polynomials of reasonable dimensions has been limited due to the lack of availability of reliable numerical methods. Recently, Parrilo [94] showed that the decomposition problem can be solved using semi-definite programming (SDP). Based on this approach, a polynomial is nonnegative for all $x \in \mathbb{R}^{n \times 1}$ if there exist a positive semidefinite matrix Q such that

$$r(x) = z(x)^T Q (z(x)) \quad (3.2)$$

In (3.2), $z(x)$ is a vector of monomials of the independent variables of $r(x)$ up to degree $d/2$, where d is the degree of $r(x)$. Now, the problem of minimizing $r(x)$ can be posed as

$$\max_Q p \quad (3.3)$$

$$\text{s.t.} \quad r(x) - p = z(x)^T Q z(x) \quad (3.4)$$

$$Q \geq 0 \quad (3.5)$$

By comparing the coefficients of the polynomials on both sides, (3.4) can be expressed as a

set of linear equality constraints. Hence the optimization problem in (3.3)-(3.5) can be seen as a convex semidefinite program (SDP). To illustrate these concepts further, we provide a simple example.

Example 3.1 *We consider finding the minimum value of the following polynomial*

$$r(x) = x^2 + 4x - 1 \quad (3.6)$$

For this purpose, we define $r'(x) = r(x) - p$. For this problem $z(x) = \begin{bmatrix} 1 & x \end{bmatrix}$ and

$$r'(x) = x^2 + 4x - 1 - p = \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} q_{11} & q_{12} \\ q_{12} & q_{22} \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix} \quad (3.7)$$

By equating the coefficients in (3.7), we obtain the following algebraic equations

$$q_{11} = -(1 + p) \quad (3.8)$$

$$q_{12} = 2 \quad (3.9)$$

$$q_{22} = 1 \quad (3.10)$$

With these equalities, Q is positive semidefinite matrix when $p \leq -5$, implying that the least achievable value of $r(x)$ is -5 .

The p^* obtained by solving the SDP does not necessarily correspond to the minimum value of $r(x)$, but is guaranteed to be its global lower bound [94]. To find x^* , which solves the optimization problem in (3.3)-(3.5), we note that $r(x^*) - p^* = z(x^*)^T Q z(x^*) = 0$. Since $Q \geq 0$, it admits Cholesky decomposition, where $Q = L^T L$ and $L z(x^*) = 0$. As the first element of the vector $z(x^*)$ is unity, $z(x^*)$ and L can be conformably partitioned to get

$$\begin{bmatrix} L_1 & L_2 \end{bmatrix} \begin{bmatrix} 1 & \tilde{z}(x^*)^T \end{bmatrix}^T = 0 \quad \Leftrightarrow \quad L_2 \tilde{z}(x^*) = -L_1 \quad (3.11)$$

Figure 3.1: Block diagram of closed loop system

Due to the numerical errors, it may not be possible to satisfy (3.11) exactly, but a least squares estimate of $\tilde{z}(x^*)$ can be obtained as

$$\tilde{z}(x^*) = -L_2^\dagger L_1 \quad (3.12)$$

Now, $x^* \in \mathbb{R}^{n \times 1}$ can be taken as the first n_x elements of $\tilde{z}(x^*)$ in (3.12). In this work, we use SeDuMi [130] interfaced with MATLAB through Yalmip [77] for solving the SDPs.

3.3 Lower bound on Achievable Performance

We consider the closed-loop system shown in Figure 3.1, where $y(t)$, $u(t)$, and $a(t)$ denote the controlled output, manipulated variable, and disturbance, respectively. For this process,

$$y(t) = g(q^{-1}) u(t) + h(q^{-1}) a(t) \quad (3.13)$$

Here $g(q^{-1})$ and $h(q^{-1})$ are the process and disturbance transfer functions, respectively.

We make the following assumptions:

1. $g(q^{-1})$ and $h(q^{-1})$ are stable, causal and contain no zeros on or outside the unit circle except at infinity (due to time delay).
2. $a(t)$ is a random noise sequence with unit variance.

Note that there is no loss of generality in assuming that $a(t)$ has unit variance, as $h(q^{-1})$ can always be scaled to satisfy this assumption. For notational simplicity, we drop the arguments q^{-1} and t in subsequent discussion.

The closed-loop transfer function between a and y can be expressed as

$$y = s a; \quad s = (1 + g k)^{-1} h \quad (3.14)$$

Our objective is to find the least achievable variance of y , *i.e.*

$$J = \min_k \text{Var}(y) = \min_k \|s\|_2^2 \quad (3.15)$$

where $\|\cdot\|_2$ denotes the $\mathcal{H}_2$ -norm. When no restrictions are imposed on the controller, the least achievable value of $\|s\|_2^2$ is known as the MV benchmark, where [45]

$$J_{mv} = \min_k \|s\|_2^2 = \sum_{i=0}^{d-1} h_i^2 \quad (3.16)$$

In (3.16), h_i is the i^{th} impulse response coefficient of h and d is the time delay associated with g . When the controller is restricted to have PID structure, however, the MV benchmark may not be achievable. The primary difficulty in characterization of achievable PID performance is the non-convexity of the optimization problem in (3.15), when k is restricted to have PID structure. A key observation to overcome this difficulty is that

$$\|s\|_2^2 = \sum_{i=0}^{\infty} s_i^2 \geq \sum_{i=0}^n s_i^2 \quad (3.17)$$

for any finite n and s_i is the i^{th} impulse response coefficient of s . Thus, a lower bound on the achievable PID performance can be found by minimizing $\sum_{i=0}^n s_i^2$. In the following

discussion, we relate s_i with the unknown parameters of the PID controller given as

$$k_{PID} = k_P + \frac{k_I}{\Delta} + k_D \Delta \quad (3.18)$$

where k_P , k_I and k_D are proportional, integral and derivative gains, respectively, and $\Delta = 1 - q^{-1}$. By defining $c_0 = (k_P + k_I + k_D)$, $c_1 = -(k_P + 2k_D)$ and $c_2 = k_D$, the PID controller in (3.18) can be alternately represented as

$$k_{PID} = \frac{1}{\Delta} c; \quad c = c_0 + c_1 q^{-1} + c_2 q^{-2} \quad (3.19)$$

We note that

$$s = (1 + g k)^{-1} h = (1 + \hat{g} c)^{-1} h \quad (3.20)$$

where $\hat{g} = g/\Delta$. When the closed-loop system is stable, (3.20) can be expanded using Taylor series expansion to get

$$s = \left[\sum_{i=0}^{\infty} (-1)^i (\hat{g} c)^i \right] h \quad (3.21)$$

For given n , we define the following $n \times n$ -dimensional Hankel matrices

$$\hat{G}_H = \begin{bmatrix} \hat{g}_0 & 0 & \cdots & 0 \\ \hat{g}_1 & \hat{g}_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{g}_{n-1} & \cdots & \hat{g}_1 & \hat{g}_0 \end{bmatrix}; \quad C_H = \begin{bmatrix} c_0 & 0 & 0 & 0 & \cdots & 0 \\ c_1 & c_0 & 0 & 0 & \cdots & 0 \\ c_2 & c_1 & c_0 & 0 & \cdots & 0 \\ 0 & c_2 & c_1 & c_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & c_2 & c_1 & c_0 \end{bmatrix} \quad (3.22)$$

and the following n -dimensional vectors

$$h_v = \begin{bmatrix} h_0 & h_1 & \cdots & h_{n-1} \end{bmatrix}^T \quad (3.23)$$

$$s_v = \begin{bmatrix} s_0 & s_1 & \cdots & s_{n-1} \end{bmatrix}^T \quad (3.24)$$

Based on (3.21)-(3.24), s_v can be compactly written as

$$s_v = \left[\sum_{i=0}^n (-1)^i (\hat{G}_H C_H)^i \right] h_v = \left[\sum_{i=0}^{\text{round}(n/d)} (-1)^i (\hat{G}_H C_H)^i \right] h_v \quad (3.25)$$

where the equality in (3.25) follows as $(\hat{G}_H C_H)^i = 0$ for $i > \text{round}(n/d)$. Now, a lower bound on the achievable PID performance can be found by solving

$$J_{PID} = \min_{c_0, c_1, c_2} \sum_{i=0}^{\infty} s_i^2 \geq \min_{c_0, c_1, c_2} s_v^T s_v \quad (3.26)$$

for any finite n . For $n = (2d - 1)$, we note from (3.25) that $s_v = (I - \hat{G}_H C_H) h_v$. Then, s_v depends linearly on the unknown controller parameters and a lower bound on the achievable PID performance can be found in a least-squares fashion [57]. The bound proposed in [57] can be loose, however, as the contribution of $s_i, i > (2d - 1)$ towards $\|s\|_2^2$ is not accounted for.

We note that though nonlinear, s_v in (3.25) and thus $s_v^T s_v$ in (3.26) is a polynomial in terms of c_0, c_1 and c_2 for every finite n . Thus, the optimization problem in (3.26) can be solved using SOS programming approach discussed in Section 3.2. In comparison with [57], the use of SOS programming provides a tighter lower bound, when $n > (2d - 1)$.

For any value of the controller parameters

$$\sum_{i=0}^{n+1} s_i^2 \geq \sum_{i=0}^n s_i^2 \quad (3.27)$$

Thus, a tight lower bound on the achievable PID performance can be obtained by solving (3.26) with n being increased sequentially until convergence. An upper bound on the achievable PID performance can be computed by finding c_0, c_1 and c_2 , which solve (3.26), and evaluating $\|s\|_2^2 = \|(1 + \hat{g}c)^{-1}h\|_2^2$ using these parameters. For sufficiently large n , $s_v^T s_v \approx \|(1 + \hat{g}c)^{-1}h\|_2^2$ and thus the lower and upper bounds can be expected to be close to each other.

Case	g	h	Reference
1	$\frac{0.2q^{-5}}{1-0.8q^{-1}}$	$\frac{1}{\Delta(1+0.4q^{-1})}$	[1]
2	$\frac{0.08919q^{-12}}{1-0.8669q^{-1}}$	$\frac{0.08919}{1-0.8669q^{-1}}$	[32]
3	$\frac{0.5108q^{-28}}{1-0.9604q^{-1}}$	$\frac{0.5108}{1-0.9604q^{-1}}$	[32]
4	$\frac{q^{-6}}{1-0.8q^{-1}}$	$\frac{1+0.6q^{-1}}{(1-0.5q^{-1})(1-0.6q^{-1})(1+0.7q^{-1})}$	[65]
5	$\frac{q^{-6}}{1-0.8q^{-1}}$	$\frac{1-0.2q^{-1}}{\Delta(1-0.3q^{-1})(1+0.4q^{-1})(1-0.5q^{-1})}$	[65]
6	$\frac{q^{-6}}{1-0.8q^{-1}}$	$\frac{1+0.6q^{-1}}{\Delta(1-0.5q^{-1})(1-0.6q^{-1})(1+0.7q^{-1})}$	[65]
7	$\frac{0.1q^{-5}}{1-0.8q^{-1}}$	$\frac{0.1}{\Delta(1-0.3q^{-1})(1-0.6q^{-1})}$	[65]
8	$\frac{0.1q^{-3}}{1-0.8q^{-1}}$	$\frac{1}{\Delta}$	[68]
9	$\frac{0.1q^{-6}}{1-0.8q^{-1}}$	$\frac{0.1}{\Delta(1-0.7q^{-1})}$	[68]
10	$\frac{0.1q^{-3}}{1-0.8q^{-1}}$	$\frac{\sqrt{0.001}}{\Delta(1+0.2q^{-1})}$	[129]

Table 3.1: Case studies for performance assessment of PID controllers

Example 3.2

The proposed method is applied to find the lower and upper bounds on the achievable PI(D) performance for numerous case studies taken from literature. These case studies and the corresponding bounds are shown in Table 3.1 and Tables 3.2 - 3.3, respectively. A few salient observations for these case studies are as follows:

1. *In comparison with time delay, the additional limitations on achievable performance*

- imposed by controller structure can be as high as 100%, e.g. for Cases 6 and 7 (PI control).*
2. *For all cases, the lower bound computed using the SOS programming on achievable PI(D) performance is tight, as can be verified through comparison with the upper bound. The maximum relative difference between the lower and upper bounds is about 7%, which is seen for Case 6 (PID control).*
 3. *For Cases 2, 3, and 4, the nearly-optimal controllers do not have integral action, i.e. the integrating poles of the controllers are cancelled by zeros on the unit circle. We recall that a stabilizing controller k must have at least as many poles at $q = 1$ as the disturbance model h . For Cases 2, 3, and 4, h does not have any integrating poles and thus a stable closed-loop system is obtained without having integral action in the controllers.*
 4. *For Case 1, the lower bound on achievable PI performance calculated using the result in [57] is 2.9726, which is close to the MV benchmark implying no significant limitations due to controller structure. In comparison, the SOS programming method correctly identifies the additional limitations on achievable performance due to controller structure.*
 5. *To characterize the limitations due to the PID structure of the controller, Huang [44] suggested using an integrator as the disturbance model. We note that for Cases 4, 5, and 6, though the process model g is the same, the limitations due to controller structure differ significantly, as the disturbance models are different for each case. This observation highlights that the use of an arbitrary disturbance model (e.g. integrator) may lead to incorrect conclusions regarding the limitations on achievable performance due to controller structure.*
 6. *For Cases 1, 3, 7 and 9, the upper bound on the achievable PI performance found*

Case	J_{mv}	LB (Kariwala [57])	UB (Literature)	SOS Programming		
				LB	UB	Controller
1	2.9427	2.9534	3.5961	3.5154	3.5186	$\frac{1.1209-0.9930 q^{-1}}{\Delta}$
2	0.0306	0.0313	0.0314	0.0313	0.0314	0.3092
3	3.0112	3.1392	3.8349	3.1703	3.1706	0.0305
4	3.4004	3.4324	3.4410	3.4408	3.4408	0.0234
5	11.9528	12.6301	17.7460	17.7044	17.7477	$\frac{0.2118-0.1896 q^{-1}}{\Delta}$
6	58.3406	68.8518	123.5400	122.4089	123.6037	$\frac{0.2304-0.2070 q^{-1}}{\Delta}$
7	0.2978	0.3337	0.6511	0.5856	0.5884	$\frac{2.6044-2.3248 q^{-1}}{\Delta}$
8	3.0000	3.0000	3.7044	3.7002	3.7050	$\frac{3.4416-3.0074 q^{-1}}{\Delta}$
9	0.3144	0.3533	0.6028	0.5949	0.5968	$\frac{2.2945-2.0588 q^{-1}}{\Delta}$
10	0.0023	0.0023	-	0.0027	0.0027	$\frac{3.5074-3.1188 q^{-1}}{\Delta}$

Table 3.2: Lower (LB) and upper (UB) bounds on achievable PI control performance

using the SOS programming is smaller than reported in the literature. This is likely due to the use of local optimal values of c_0 , c_1 , and c_2 found using gradient based methods [1, 65, 68] or stochastic optimization [32] for the non-convex optimization problem.

7. For each case, the proposed method takes less than 100 seconds on a PC with Intel® Dual Core™ 2.40GHz, 2GB RAM using MATLAB® 2007a showing computational efficiency, even for processes with large time delay (Case 3).

In summary, these case studies demonstrate that SOS programming can be used to find tight bounds on achievable $PI(D)$ performance efficiently and reliably.

The proposed approach can be extended to minimize weighted sensitivity function. For this purpose, let w be the frequency dependent weight. Let define w_i as the i^{th} impulse response coefficients of w and let w_H be the Hankel marix containing the impulse response

Case	J_{mv}	LB (Kariwala [57])	SOS Programming		
			LB	UB	Controller
1	2.9427	2.9486	3.0730	3.0730	$\frac{2.8327-4.3950q^{-1}+1.7478q^{-2}}{\Delta}$
2	0.0306	0.0310	0.0310	0.0310	$1.7433 - 1.4495q^{-1}$
3	3.0112	3.0383	3.0492	3.0495	$0.4796 - 0.4493q^{-1}$
4	3.4004	3.4025	3.4065	3.4065	$0.1338 - 0.1148q^{-1}$
5	11.9528	11.9606	13.6341	13.8243	$\frac{0.7251-1.2072q^{-1}+0.5200q^{-2}}{\Delta}$
6	58.3406	59.3154	83.5605	89.6983	$\frac{0.8989-1.5413q^{-1}+0.6926q^{-2}}{\Delta}$
7	0.2978	0.2998	0.4278	0.4278	$\frac{8.6911-14.3626q^{-1}+6.2116q^{-2}}{\Delta}$
8	3.0000	3.0000	3.2093	3.2093	$\frac{6.6909-9.5975q^{-1}+3.6106q^{-2}}{\Delta}$
9	0.3144	0.3162	0.4288	0.4288	$\frac{8.7008-14.6895q^{-1}+6.4323q^{-2}}{\Delta}$
10	0.0023	0.0024	0.0024	0.0025	$\frac{5.1681-6.6294q^{-1}+2.0130q^{-2}}{\Delta}$

Table 3.3: Lower (LB) and upper (UB) bounds on achievable PID control performance

coefficients of w . We have

$$w_H = \begin{bmatrix} w_0 & 0 & \cdots & 0 \\ w_1 & w_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ w_{n-1} & \cdots & w_1 & w_0 \end{bmatrix}; \quad W = w_H s_v \quad (3.28)$$

Similar to (3.26), the following identity holds.

$$J_{w, freq} = \min_{c_0, c_1, c_2} \sum_{i=0}^{\infty} W_i^2 \geq \min_{c_0, c_1, c_2} \sum_{i=0}^n W_i^2 \quad (3.29)$$

The optimization problem in (3.29) can be solved using SOS programming as before. Similarly, the parameters which solves the optimization problem in (3.29) can be used as the controller parameter to obtain tight upper bound on the achievable performance.

Example 3.3 *Based on the earlier case studies, it can be noted that the controllers do not have integral action for processes with non-integrating disturbance. One possible way to*

enforce integral action in the controller is by choosing w as an integrator, i.e $w = 1/\Delta$. To illustrate this concept, we revisit Case 2 in Example 3.2.

For this process, we have $J_{w,mv} = \min_k \|ws\|_2^2 = 1.605$, When the controller is restricted to be of PI-type, we find that $J_{w,pi} \geq 4.327$ (lower bound). The following PI controller is designed based on the solution obtained from SOS programming.

$$K_{PI} = \frac{1.520 - 1.428^{-1}}{\Delta} \quad (3.30)$$

which provides $\|ws\|_2^2 = 4.387$ (upper bound) and $\|s\|_2^2 = 0.062$. When PID controller is used, we have $J_{w,pid} \geq 3.051$. The following controller is designed based on the solution obtained from SOS programming.

$$K_{PID} = \frac{8.061 - 14.331q^{-1} + 6.499q^{-2}}{\Delta} \quad (3.31)$$

Using K_{PID} , we have $\|ws\|_2^2 = 3.051$ and $\|s\|_2^2 = 0.063$. For both PI and PID controllers, the lower bound found using SOS programming closely matches the upper bound.

Compared to the achievable performance without considering frequency dependent weight, the least output variance achieved by forcing the presence of integral action in the controller increases by a factor of two. The increase in the achievable output variance comes as a cost of introducing an integrator in the controller. This issue becomes apparent from the plot of the closed-loop sensitivity function evaluated at different frequencies with and without integrating controller; see Figure 3.2. When the integrator is included in the controller, the sensitivity function exhibits a large peak around $\omega = 0.11$ Hz. Consequently the $\mathcal{H}_2$ -norm of closed-loop system with integrating controller is higher than without integrating controller.

Figure 3.2: Frequency response of closed-loop system with and without weighting factor for Example 3.3

3.4 Weighted Input and Output Performance

Controller tuning through minimization of output variance only (cheap control) may lead to excessive input variation. This difficulty can be overcome by penalizing the rate of input change, *i.e.* minimizing a weighted sum of the variances of output and input rate defined as

$$J_{w,input} = \min_k (\text{Var}(y) + \rho \text{Var}(\Delta u)) \quad (3.32)$$

where ρ is the penalty factor. When there is no restriction on the controller, the optimal solution to the problem in (3.32) can be found using the subspace method [49]. When a PID controller is used, similar to cheap control, *i.e.* $\rho = 0$, the bounds on $J_{w,input}$ found using the subspace method may not be achievable. To find the achievable value of $J_{w,input}$ with PID controller, let $\Delta u = w a = -\Delta k s a$. For given n , define

$$w_v = \begin{bmatrix} w_0 & w_1 & \cdots & w_{n-1} \end{bmatrix}^T \quad (3.33)$$

It follows that $w_v = -C_H s_v$. Then

(a) PI controller

(b) PID controller

 Figure 3.3: Trade-off between $\text{Var}(y)$ and $\text{Var}(\Delta u)$ for Case 1

$$J_{w,PID} = \min_{c_0, c_1, c_2} \sum_{i=0}^{\infty} (s_i^2 + \rho w_i^2) \quad (3.34)$$

$$\geq \min_{c_0, c_1, c_2} \sum_{i=0}^n (s_i^2 + \rho w_i^2) = \min_{c_0, c_1, c_2} (s_v^T s_v + \rho w_v^T w_v) \quad (3.35)$$

for any finite n . As before, the optimization problem in (3.35) can be solved using SOS programming approach discussed in Section 2. The lower and upper bounds on output variance can be computed by evaluating $s_v^T s_v$ and $\|(1 + \hat{g}c)^{-1}h\|_2^2$, respectively, using c_0, c_1 and c_2 , which solve (3.35). The lower and upper bounds on the variance of input rate can be found similarly by evaluating $w_v^T w_v$ and $\|c(1 + \hat{g}c)^{-1}h\|_2^2$, respectively.

Example 3.4 For Case 1 in Table 3.1, we minimize the weighted sum of the variances of output and input rate for different values of ρ taken in the range 0 – 100. The lower and upper bounds on the trade-off curve between $\text{Var}(y)$ and $\text{Var}(\Delta u)$ for PI and PID control are shown in Figure 3.3. The trade-offs curve in Figure 3.3 confirm that the controller

structure gives rise to additional limitations on achievable control performance. These limitations are more prominent for small values of ρ . When ρ is increased, the trade-off curves for different controllers approach each other. This is expected, as when $\rho \rightarrow \infty$, the optimal unrestricted and $PI(D)$ controllers approach zero controllers implying small $\text{Var}(\Delta u)$, but arbitrarily large $\text{Var}(y)$. Similar trends are seen for the other cases given in Table 3.1. Figure 3.3 also demonstrates that SOS programming provides tight lower bounds on the variances of output and input rate, as the lower and upper bounds are almost identical for different values of ρ .

3.5 Chapter Summary

In this chapter, we explored the potential of SOS programming to characterize the lower bound on achievable PID performance, defined in terms of output variance. Numerical examples taken from the literature demonstrate the tightness of the proposed bounds. In other words, there exist PID controllers which provide the performance close to the lower bounds indicated by the proposed approach. The method is also extended to minimize the weighted sum of the variances of input rate and output to obtain the LQG trade-offs curve for PID controllers.

Chapter 4

Achievable Performance of Decentralized Controllers

4.1 Introduction

Similar to the PID structure of controller, as discussed in Chapter 3, the diagonal structure of the decentralized controller can pose limitations on the achievable performance. When a fully centralized controller is used, the least achievable output variance is given by the MV benchmark for multi-input multi-output (MIMO) processes [34, 45]. However, the derivation of MV benchmark for decentralized controllers is difficult. This happens as the optimization problem involving minimization of output variance becomes non-convex, once the diagonal structure is imposed on the controller [105, 127]. With this difficulty, one may alternately look for tight upper and lower bounds on the least achievable output variance using decentralized controllers.

¹Part of this chapter was presented in ADCHEM Conference, Istanbul, Turkey, 2009 [111].

²The contents of this chapter have been submitted to *Journal of Process Control*.

Clearly, any suboptimal tuning strategy for the decentralized controller provides an upper bound on the least achievable output variance. Some approaches for finding upper bounds on least achievable output variance have been reported by utilizing the structure of the optimization problem [62, 152]. Recently, Kariwala [57] addressed the more difficult problem involving derivation of a tight lower bound on the least achievable output variance, where an explicit bound is proposed by considering those impulse response coefficients of the closed-loop transfer function between disturbances and outputs, which depend linearly on the controller parameters. In general, however, the lower bound proposed in [57] can be conservative due to the neglected impulse response coefficients.

In this work, we aim at improving the results in [57] by considering the contribution of the impulse response coefficients, which depend nonlinearly on the controller parameters. We show that though nonlinear, the impulse response coefficients of the closed-loop transfer function between disturbances and outputs can be represented as polynomials in unknown controller parameters. The resulting non-convex optimization problem related to the minimization of output variance is solved using sums of squares (SOS) programming [86, 94, 95] technique, described in Chapter 3.

In practice, the individual sub-controllers of the decentralized controller are often restricted to be of PID type, which can pose additional limitations on the achievable output variance. Various approaches to find an upper bound on the achievable output variance for MIMO processes under decentralized PID control have been proposed [51, 66].

In Chapter 3, an SOS programming approach was presented for finding a tight lower bound on the achievable output variance for SISO processes under PID control, which is extended to decentralized PID controllers in this chapter. Benchmark examples taken from literature are used to demonstrate the usefulness of the SOS programming approach.

Figure 4.1: Block diagram of closed loop system

4.2 Problem Formulation

We consider the closed loop system shown in Figure 4.1. For this system, we denote $G(q^{-1})$ and $H(q^{-1})$ as the process and disturbance models, respectively, such that

$$y(t) = G(q^{-1})u(t) + H(q^{-1})a(t) \quad (4.1)$$

Here, $y(t)$, $u(t)$ and $a(t)$ are controlled outputs, manipulated variables and disturbances, respectively. We make the following simplifying assumptions:

1. $G(q^{-1})$ and $H(q^{-1})$ are stable, causal transfer matrices, contain no zeros on or outside the unit circle except at infinity (due to time delays).
2. $G(q^{-1})$ is square with dimensions $n_y \times n_y$ and the input-output pairings are selected on the diagonal elements of $G(q^{-1})$.
3. $a(t)$ is a random noise sequence with unit variance.
4. $y_{sp}(t)$ is zero for all $t > 0$.

When $H(q^{-1})$ contains zeros outside the unit circle, these zeros can be factored through an all pass factor without affecting the noise spectrum [45]. Further, there is no loss of generality in assuming that the system is affected by noise having unit variance, as the disturbance model can always be scaled to satisfy this assumption. The setpoint $y_{sp}(t)$

is assumed to be zero for simplicity and the results presented in this paper can be easily generalized to the case, where $y_{sp}(t)$ changes with time; see Remark 4.1 for details. For notational simplicity, we drop the arguments q^{-1} and t in the subsequent discussion.

Our objective is to find the least achievable value of $\text{Var}(y)$ with respect to the controller K , *i.e.*

$$J_{\text{decen}} = \min_K \text{Var}(y) = \min_K E[\text{tr}(y y^T)] \quad (4.2)$$

where $E(\cdot)$ and $\text{tr}(\cdot)$ denote the expectation and trace operators, respectively and K is assumed to have a diagonal structure, *i.e.*

$$K = \text{diag}(K_1, K_2, \dots, K_{n_y}) \quad (4.3)$$

J_{decen} in (4.2) is the MV benchmark for decentralized controllers.

Remark 4.1 *The focus of this chapter is on regulatory control and thus y_{sp} is assumed to be zero for all t . When y_{sp} changes with time, the MV benchmark can be defined based on control error, $e = y_{sp} - y$ [45]. In particular, J_{decen} in (4.2) can be defined as $\min_K E[\text{tr}(e e^T)]$ to assess the performance of decentralized controllers.*

4.3 Conventional Approach

Let G be partitioned as:

$$G = D^{-1} \bar{G} \quad (4.4)$$

such that $\bar{G}$ and D^{-1} contain the invertible and non-invertible parts of G , respectively. We consider that $D(q)$ is a unitary interactor matrix, *i.e.* $D^T(q^{-1}) D(q) = I$ [45]. Let

$$q^{-d} D H = F + q^{-d} R \quad (4.5)$$

where d denotes the order of the interactor matrix. Then, the MV benchmark for fully centralized controllers is given as

$$J_{\text{full}} = \min_K E[\text{tr}(y y^T)] = \|F\|_2^2 \quad (4.6)$$

J_{full} in (4.6), however, does not take the diagonal structure of the controller into account and can underestimate the achievable performance of decentralized controllers. In the subsequent discussion, we present a lower bound on the achievable output variance for processes under decentralized control.

4.4 MV Benchmark for Decentralized Controllers

In this chapter, the SOS programming technique described in Chapter 3 is used to characterize lower bound on achievable performance using decentralized control. Although the derivation is similar to that presented in Chapter 3, the detailed derivation is presented for the sake of clarity.

For regulatory control, we have $u = -K y$. Thus, the closed loop transfer function between a and y can be written as

$$y = S a; \quad S = (I + G K)^{-1} H \quad (4.7)$$

Since $E[aa^T] = I$,

$$J_{\text{decen}} = \min_K E[\text{tr}(y y^T)] = \min_K \|S\|_2^2 \quad (4.8)$$

When diagonal structure is imposed on K , the optimization problem in (4.8) becomes

non-convex. A key observation to overcome this difficulty is that

$$\|S\|_2^2 = \sum_{i=0}^{\infty} \text{tr}(S_i^T S_i) \geq \sum_{i=0}^n \text{tr}(S_i^T S_i) \quad (4.9)$$

for any finite n . Thus, a lower bound on J_{decen} can be found by minimizing $\sum_{i=0}^n \text{tr}(S_i^T S_i)$.

Here, S_i represents the i th impulse response coefficient matrix of S .

When the closed loop system is stable, (4.7) can be expanded using Taylor Series expansion to get

$$S = \left[\sum_{i=0}^{\infty} (-1)^i (G K)^i \right] H \quad (4.10)$$

For given n , we define the following $nn_y \times nn_y$ -dimensional Hankel matrices

$$G_H = \begin{bmatrix} G_0 & 0 & \cdots & 0 \\ G_1 & G_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ G_{n-1} & \cdots & G_1 & G_0 \end{bmatrix} \quad (4.11)$$

$$K_H = \begin{bmatrix} K_0 & 0 & \cdots & 0 \\ K_1 & K_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ K_{n-1} & \cdots & K_1 & K_0 \end{bmatrix} \quad (4.12)$$

and the following $nn_y \times nn_d$ -dimensional matrices

$$H_v = \begin{bmatrix} H_0^T & H_1^T & \cdots & H_{n-1}^T \end{bmatrix}^T \quad (4.13)$$

$$S_v = \begin{bmatrix} S_0^T & S_1^T & \cdots & S_{n-1}^T \end{bmatrix}^T \quad (4.14)$$

Based on (4.12) - (4.14), S_v can be compactly written as

$$S_v = \left[\sum_{i=0}^n (-1)^i (G_H K_H)^i \right] H_v \quad (4.15)$$

Now, a lower bound on J_{decen} can be found by solving

$$J_{\text{decen}} = \min_K \sum_{i=0}^{\infty} \text{tr}(S_i^T S_i) \geq \min_K \text{tr}(S_v^T S_v) \quad (4.16)$$

for any finite n . Based on (4.15), we note that S_v and thus $\text{tr}(S_v^T S_v)$ depends polynomially on the controller parameters. Thus, the optimization problem in (4.16) can be seen as finding the global minimal value of a polynomial. For this purpose, we use SOS programming [94] in this work. SOS programming transforms the polynomial minimization problem to a semi-definite program, which is solved using Sedumi [130] interfaced with Matlab through Yalmip [77].

SOS programming does not necessarily provide the minimum value of the polynomial, but guarantees a global lower bound [94]. For any controller K , however, since

$$\sum_{i=0}^{n+1} \text{tr}(S_i^T S_i) \geq \sum_{i=0}^n \text{tr}(S_i^T S_i) \quad (4.17)$$

tight lower bound on J_{decen} can be found by increasing n sequentially until convergence. In comparison with [57], where only the first $(2d - 1)$ impulse response coefficients of S are used to find a lower bound on J_{decen} , the use of SOS programming provides a tighter lower bound, whenever $n > (2d - 1)$.

Example 4.1 We consider the following 2×2 process, where

$$G = \begin{bmatrix} \frac{-0.1q^{-2}}{(1-0.1q^{-1})(1-0.2q^{-1})} & \frac{-0.25q^{-1}(1-0.3q^{-1})}{(1-0.1q^{-1})(1-0.2q^{-1})} \\ \frac{0.5q^{-1}(1+0.9q^{-1})}{(1-0.1q^{-1})(1-0.2q^{-1})} & \frac{-0.1q^{-2}}{(1-0.1q^{-1})(1-0.2q^{-1})} \end{bmatrix} \quad (4.18)$$

and $H = 1/(1 - q^{-1}) I$ [57]. Then, $D = qI$, $F = I$ and the least output variance achievable using full multivariable controllers is $J_{\text{full}} = \|F\|_2^2 = 2$.

By considering the contribution of first $(2d - 1)$ impulse response coefficients of S towards $\|S\|_2^2$, Kariwala [57] found that $J_{\text{decen}} \geq 4$. This lower bound is, however, not very tight due to the neglected impulse response coefficients of S . Using non-convex optimization with multiple randomized initial guesses, Kariwala [57] showed that the exact value of J_{decen} is approximately 8.16, which differs significantly from the reported lower bound.

Using the SOS programming approach, we find that $J_{\text{decen}} \geq 8.023$, which is close to the upper bound found in [57]. This example sufficiently demonstrates the usefulness of the SOS programming technique for finding a tighter lower bound on J_{decen} , as compared to the approach presented in [57]. In addition, for this process, the MV benchmark for full multivariable controller is approximately 4 times lower than the MV benchmark found by accounting for the controller structure, which shows that the controller structure can significantly limit the achievable performance. Thus, the conventional approach for performance assessment of decentralized controllers using MV benchmark for full multivariable controllers may incorrectly classify well-performing controllers as poorly performing.

4.5 MV benchmark for Decentralized PID Controllers

In industrial practice, the individual sub-controllers of the decentralized controller are often fixed to be of PID-type. Clearly, the presence of additional controller structure can further limit the least achievable variance of outputs. Next, we show that the SOS programming approach can be easily extended to derive a tight lower bound on J_{decen} for decentralized PID controllers.

We note that the decentralized PID controller can be expressed as

$$K_{PID} = \frac{1}{\Delta} \sum_{i=0}^2 C_i q^{-i} \quad (4.19)$$

where C_i has diagonal structure and $\Delta = 1 - q^{-1}$ is the integrator. For

$$\hat{G} = \frac{1}{\Delta} G \quad (4.20)$$

let us define the Hankel matrices $\hat{G}_H$ and C_H , which have the same structure as G_H and K_H defined in (4.12). Using similar approach, as used in Section 4.4, it can be shown that for any finite n

$$S_v = \left[\sum_{i=0}^n (-1)^i (\hat{G}_H C_H)^i \right] H_v \quad (4.21)$$

Thus, a lower bound on J_{decen} for decentralized PID controller can be found by minimizing $\text{tr}(S_v^T S_v)$ using SOS programming as before.

Example 4.2 *We revisit Example 4.1. When individual sub-controllers are restricted to be of PI-type, the lower bound on J_{decen} increases to 9.980 (approximately 25% higher than unrestricted decentralized controller). The following sub-optimal controller is designed using trial and error*

$$K_{PI} = \begin{bmatrix} \frac{-0.629+0.474 q^{-1}}{\Delta} & 0 \\ 0 & \frac{-2.844+1.862 q^{-1}}{\Delta} \end{bmatrix} \quad (4.22)$$

which provides $E[\text{tr}(y y^T)] = 10.087$. When PID controllers are used, the lower bound on J_{decen} is 9.250 (approximately 15% higher than unrestricted decentralized controller). The

following sub-optimal controller

$$K_{\text{PID}} = \begin{bmatrix} \frac{-0.884+0.924q^{-1}-0.205q^{-2}}{\Delta} & 0 \\ 0 & \frac{-3.210+2.764q^{-1}-0.869q^{-2}}{\Delta} \end{bmatrix} \quad (4.23)$$

provides $E[\text{tr}(y y^T)] = 9.421$. For both cases (PI and PID control), the lower bounds are close to the upper bounds, which demonstrates that SOS programming technique can be used to find tight bounds on J_{decen} for decentralized PID controllers efficiently and reliably.

Example 4.3 We consider the example of a 4×4 multi-component distillation column [2]. The controlled variables are distillate fraction of 1st component (y_1), bottom fraction of 2nd component (y_2), distillate fraction of 3rd component (y_3), and temperature (y_4). The manipulated variables are reflux ratio (u_1), reboiler duty (u_2), reboiler duty of sidestream (u_3), and sidestream flowrate (u_4), and the disturbance is sidestream feed composition (a). The discrete time transfer function model obtained using a sampling time of 1 minute is given as:

$$G = \begin{bmatrix} \frac{0.0035q^{-2}+0.0097q^{-3}+0.0006q^{-4}}{1-1.857q^{-1}+0.86q^{-2}} & \frac{-0.0032q^{-1}-0.0063q^{-2}-0.0002q^{-3}}{1-1.92q^{-1}+0.92q^{-2}} \\ \frac{-0.0916q^{-5}}{1-0.978q^{-1}} & \frac{0.151q^{-2}+0.0030q^{-3}}{1-0.978q^{-1}} \\ \frac{-0.00486q^{-18}-0.00462q^{-19}}{1-1.852q^{-1}+0.857q^{-2}} & \frac{0.0137q^{-12}+0.0131q^{-13}}{1-1.855q^{-1}+0.86q^{-2}} \\ \frac{-0.003q^{-3}-0.0273q^{-4}-0.006q^{-4}}{1-1.834q^{-1}+0.8377q^{-2}} & \frac{0.083q^{-1}+0.012q^{-2}-0.0788q^{-3}}{1-2.767q^{-1}+2.591q^{-2}-0.823q^{-3}} \\ & \frac{-0.007q^{-1}-0.0046q^{-2}}{1-0.9535q^{-1}} & \frac{-0.0005q^{-6}-0.0005q^{-7}}{1-1.911q^{-1}+0.9131q^{-2}} \\ & \frac{-0.00002q^{-6}-0.00002q^{-7}}{1-1.943q^{-1}+0.943q^{-2}} & \frac{0.006362q^{-3}+0.0252q^{-4}}{1-0.9794q^{-1}} \\ & \frac{0.2402q^{-2}+0.0024q^{-3}}{1-0.947q^{-1}} & \frac{-0.1797q^{-1}-0.1738q^{-2}}{1-0.935q^{-1}} \\ & \frac{-0.0003q^{-1}-0.0003q^{-2}}{1-1.788q^{-1}+0.793q^{-2}} & \frac{0.0012q^{-1}+0.01q^{-2}+0.0023q^{-3}}{1-1.833q^{-1}+0.8356q^{-2}} \end{bmatrix} \quad (4.24)$$

$$H = \begin{bmatrix} \frac{-0.00113q^{-7}-0.0011q^{-8}}{1-1.898q^{-1}+0.901q^{-2}} \\ \frac{-0.02986q^{-6}}{1-0.972q^{-1}} \\ \frac{0.0490q^{-10}}{1-0.959q^{-1}} \\ \frac{-0.016q^{-1}-0.0304q^{-2}-0.0008q^{-3}}{1-1.724q^{-1}+0.778q^{-2}} \end{bmatrix} \quad (4.25)$$

For this process, the interactor matrix is found to be

$$D(q) = \begin{bmatrix} 0 & 0 & 0 & q \\ q^2 & 0 & 0 & 0 \\ 0 & q^2 & 0 & 0 \\ 0 & 0 & q^2 & 0 \end{bmatrix} \quad (4.26)$$

for which $d = 2$. By considering the first 2 impulse response coefficient matrices of $q^{-d}DH$, we find that $J_{full} = 2.56 \times 10^{-3}$ implying that nearly perfect control is possible with the use of a fully centralized controller.

For decentralized control, RGA analysis [11] suggests the use of diagonal pairings. When decentralized PI controller with diagonal pairing is used, the lower bound on the output variance of the processes calculated using SOS programming is 0.1105 showing that with decentralized controllers, the achievable performance is much larger than that can be achieved using fully centralized controllers. The following controller

$$K = \begin{bmatrix} 0.031 & 0 & 0 & 0 \\ 0 & -0.076 & 0 & 0 \\ 0 & 0 & 0.419 & 0 \\ 0 & 0 & 0 & 0.200 \end{bmatrix} \quad (4.27)$$

is designed using trial and error, which provides $E[\text{tr}(y y^T)] = 0.1233$ and can be considered to be an upper bound on the least output variance achievable using

decentralized PI controllers. Based on this upper bound, we note that lower bound obtained using SOS programming approach is tight.

4.6 Chapter Summary

In this chapter, SOS programming technique is proposed to find a lower bound on the least achievable output variance using decentralized controllers. This result is also extended to the case, where the individual controllers are restricted to be PID controllers. Numerical examples show that the lower bound found using the proposed method is tight and the controller structure can significantly limit the achievable performance.

Unlike a PID controller, which has only three parameters, finding lower bound on achievable decentralized performance is more difficult. The problem arises because for decentralized controller the number parameters to be tuned can be arbitrarily large. For example, finding a lower bound for a 2×2 system with PID controller requires solving a polynomial with 70, 210, 495, and 1001 monomials for $n = 2, 3, 4$, and 5, respectively. Consequently, the size of LMI becomes extremely large and the computational time increases significantly. Finding efficient methods for minimizing the polynomial related to achievable decentralized performance is an issue for future research.

Chapter 5

Practical Performance Assessment of Decentralized Controllers

5.1 Introduction

With the increasing competition, process industries continue to strive to operate their plants safely and economically. Ideally, the plant should be operated at the economically optimal operating point, which often lies at intersection of process constraints. In practice, however, a back-off from the constraints is required to accommodate for the variability of the process variables; see Figure 5.1. Clearly, increased economic returns can be obtained by reducing the process variability, *e.g.* using a control system.

The performance of a well-designed control system can degrade over time due to changes in operating conditions and disturbance dynamics. Controller performance assessment is useful for identifying the opportunities for performance improvement of

¹Parts of this chapter were presented in ADCHEM Conference, Istanbul, Turkey, 2009 [111] and CUTSE international Conference, Miri, Malaysia, 2009 [112].

²The contents of this chapter have been submitted to *Journal of Process Control*.

Figure 5.1: Illustration of improved operation through output variance reduction

industrial controllers. Among the various available methods [35, 52, 99], minimum variance (MV) benchmarking is one of the most widely used methods for controller performance assessment. In this approach, the minimum achievable variance of the process outputs is taken as a benchmark, *i.e.* MV benchmark. The controller is deemed to provide satisfactory performance, if the observed output variance is close to the MV benchmark. On the other hand, reduction in output variance is considered to be feasible through controller retuning, when the observed output variance is significantly lower than the MV benchmark.

The origin of MV benchmark can be traced back to Åström [133] and Jenkins *et al.* [10], who demonstrated that the achievable output variance for a single-input single-output (SISO) process under feedback control depends on the first few impulse response coefficients of the disturbance model. Harris [33] showed that with *a priori* knowledge of time delay, MV benchmark can be estimated from closed loop operating data and established it as a tool for performance assessment of SISO systems. Using the concept of interactor matrices [24], Harris *et al.* [34] and Huang *et al.* [46] proposed MV benchmark for multi-input multi-output (MIMO) systems. This methodology has also been extended to different controller architectures including feedback/feedforward controllers [18, 47],

Figure 5.2: Insufficiency of available MV benchmarks for performance assessment of decentralized controllers

cascade controllers [67, 71] and constrained model predictive controllers [37], and process types such as linear time varying processes [43, 148] and nonlinear processes [25, 36]. The success of MV benchmark for controller performance assessment is highlighted by its incorporation into a number of industrial software, which are used routinely in process industries [52, 99].

This chapter focuses on performance assessment of decentralized controllers used for feedback control of linear time invariant processes. Owing to their simplicity in design and maintenance, decentralized controllers are widely used for regulation purposes in process industries. The distinctive feature of decentralized controllers is their diagonal structure. It is well-known that similar to time delay, the diagonal structure of the decentralized controllers can also limit the achievable output variance; see *e.g.* [57]. The available MV benchmark for SISO processes [33] assumes that the loop is operating in isolation from the rest of the process. Although this tool has been used by industrial vendors, it implicitly ignores interactions among the loops of the decentralized controller. On the other hand, the available MV benchmark for MIMO processes [34, 46] considers that the controller is centralized or full multivariable and thus fails to account for the limitations due to controller structure. Thus, both SISO and MIMO MV benchmarks can lead to incorrect conclusions regarding performance assessment of decentralized controllers.

Naturally, the key to deriving an exact MV benchmark for decentralized controllers is the characterization of achievable output variance. This problem is, however, difficult to solve, as the optimization problem involving minimization of output variance becomes

non-convex, once the diagonal structure is imposed on the controller [105, 127]. Some approaches for finding upper [51, 62, 66, 152] and lower bounds [57] on the achievable output variance have been proposed. A tighter lower bound on the achievable output variance can also be found using the approach presented in Chapter 4. These approaches, however, have limited applicability for performance assessment of industrial controllers, as they require full knowledge of the process and/or disturbance models.

This chapter aims to provide a more practical approach for performance assessment of decentralized controllers. In particular, the goal is to present a method, which can be applied using closed-loop operating data without requiring the full process model. Theoretically, the least achievable output variance is seen, where all sub-controllers of the decentralized controller are allowed to be re-tuned simultaneously. In process industries, however, control loops are mostly tuned independently or sequentially. With this view, we focus on a loop by loop analysis, which provides a measure of reducing the variance of i th output through retuning of the i th controller.

In this chapter, we propose an MV benchmark for loop by loop analysis. The proposed result requires a modification of the available MV benchmark for SISO processes to account for the loop interactions. It is further shown that the modified MV benchmark can be directly estimated from closed loop data with *a priori* knowledge of the delays between all the inputs and outputs. An interesting insight is that for processes under decentralized control, the pre-whitening of output data using algorithms such as filtering and correlation (FCOR) algorithm [45] does not necessarily provide the first few impulse response coefficients of the disturbance model, as is the case for SISO processes. Numerous examples taken from literature are used to demonstrate the usefulness of the proposed approach.

5.2 MV Benchmark for Decentralized Controllers

We consider the closed loop system shown in Figure 4.1 and the open loop transfer function in (4.1). Our objective is to find the least achievable variance (MV benchmark) of y_i through retuning of K_i , *i.e.*

$$J_{i,\text{decen}} = \min_{K_i} E[\text{tr}(y_i y_i^T)] \quad (5.1)$$

Based on (5.1), the performance index for the i th output can be defined as

$$\eta_{i,\text{decen}} = \frac{J_{i,\text{decen}}}{E[\text{tr}(y_i y_i^T)]} \quad (5.2)$$

where $E[\text{tr}(y_i y_i^T)]$ is the observed variance of the i th output. Similar to 4.1, performance assessment of each loop on decentralized controllers in the presence of setpoint change can be defined as $\min_K E[\text{tr}(e e^T)]$ and $J_{i,\text{decen}} = \min_{K_i} E[\text{tr}(e_i e_i^T)]$ for $i = 1, 2, \dots, n_y$.

We first consider performance assessment of the decentralized controller for 2×2 systems. The result is generalized to $n_y \times n_y$ systems in Section 5.2.3.

For a 2×2 system, (4.1) can be written as

$$y_1 = G_{11} u_1 + G_{12} u_2 + H_1 a \quad (5.3)$$

$$y_2 = G_{21} u_1 + G_{22} u_2 + H_2 a \quad (5.4)$$

We assume that G_{ii} does not contain unstable zeros. The time delay associated with G_{ij} is denoted as d_{ij} , *i.e.*

$$G_{ij} = q^{-d_{ij}} \bar{G}_{ij} \quad (5.5)$$

where $\bar{G}_{ij}$ denotes the invertible part of G_{ij} . Without loss of generality, we consider that the objective is to characterize the least achievable variance of y_1 .

5.2.1 Conventional Approach

A common approach for loop by loop analysis involves using the MV benchmark for SISO systems. Here, H_1 is decomposed using Diophantine identity as

$$H_1 = F_1 + q^{-d_{11}} R_1 \quad (5.6)$$

Then, the least achievable variance of y_1 is taken as [33, 133]

$$J_{1,iso} = \min_{K_1} E[\text{tr}(y_1 y_1^T)] = \|F_1\|_2^2 \quad (5.7)$$

where $\|\cdot\|_2$ denotes the $\mathcal{H}_2$ -norm. The performance index for individual outputs is defined similar to (5.2). An inherent assumption in the use of (5.7) for loop by loop analysis is that $u_2 = 0$ at all times or in other words, the first loop is being operated in isolation from the rest of the process. We next demonstrate that when the presence of other loops is accounted for, the first d_{11} elements of H_1 are not necessarily feedback invariant and thus the least achievable variance of y_1 can be lower than $J_{1,iso}$ in (5.7).

5.2.2 Modified MV Benchmark

Consider that the second loop is closed with $u_2 = -K_2 y_2$. Under partially closed loop conditions, we have [124]

$$y_1 = P_{11} u_1 + P_{d1} a \quad (5.8)$$

where

$$P_{11} = G_{11} - \frac{G_{12} K_2 G_{21}}{1 + G_{22} K_2} \quad (5.9)$$

$$P_{d1} = H_1 - \frac{G_{12} K_2 H_2}{1 + G_{22} K_2} \quad (5.10)$$

In subsequent analysis, we assume that P_{11} does not contain unstable poles and zeros, which is often satisfied in practice. Some discussion on relaxing this assumption is provided in Section 5.3.

Let $P_{d1,m}$ denote the minimum phase version of P_{d1} obtained by mirroring all unstable zeros of P_{d1} across the unit circle and $\mathcal{A}$ be the all-pass factor containing the unstable zeros of P_{d1} . Then,

$$y_1 = P_{11} u_1 + P_{d1,m} \tilde{a} \quad (5.11)$$

where $\tilde{a} = \mathcal{A}a$ with $\text{Var}(\tilde{a}) = \text{Var}(a) = 1$. In (5.9)-(5.10), since $1/(1+G_{22}K_2)$ and K_2 are rational and invertible, it follows that the delay associated with P_{11} or the effective delay of the first loop is

$$d'_1 = \min(d_{11}, d_{12} + d_{21}) \quad (5.12)$$

Now, let $P_{d1,m}$ be decomposed using Diophantine identity as

$$P_{d1,m} = F'_1 + q^{-d'_1} R'_1 \quad (5.13)$$

Using (5.11) and (5.13), it follows that

$$y_1 = F'_1 \tilde{a} + q^{-d'_1} (\bar{P}_{11} u_1 + R'_1 \tilde{a}) \quad (5.14)$$

where $P_{11} = q^{-d'_1} \bar{P}_{11}$ and $\bar{P}_{11}$ denotes the invertible part of P_{11} . Since the first term in (5.14) cannot be affected by u_1 (invariant of K_1), it follows that

$$J_{1,\text{decen}} = \min_{K_1} E[\text{tr}(y_1 y_1^T)] = \|F'_1\|_2^2 \quad (5.15)$$

Here, $\|F'_1\|_2^2$ represents the least achievable variance of y_1 , when the presence of second loop is accounted for. In comparison with (5.7), we note that

1. For decentralized control, one needs to consider the impulse response coefficients of P_{d1} (disturbance model for partially closed-loop system) and not H_1 (open loop disturbance model).
2. The number of impulse response coefficients to be considered is d'_1 (effective delay) and not d_{11} (open loop delay).

Both these factors can lead to incorrect conclusions regarding performance assessment of decentralized controllers using MV benchmark for SISO processes. Note that in general, $J_{1,\text{decen}}$ may depend on K_2 . This dependance, however, has no bearing on the loop by loop performance assessment, where the objective is to find the least achievable variance of y_i through tuning of K_i .

Figure 5.3: Presence of parallel path (shown with thick line) from u_1 to y_1 for 2×2 systems

Though the result in (5.15) may seem entirely mathematical, a physical reasoning with this result can be associated by considering the block diagram of a 2×2 system shown in Figure 5.3. Here, u_1 can affect y_1 directly through G_{11} , but also through a parallel path involving G_{12} and G_{21} (shown with thick line in Figure 5.3). Thus, $d'_1 = \min(d_{11}, d_{12} + d_{21})$ represents the delay of the fastest path through which u_1 can affect y_1 . In this sense, loop interaction can sometimes be beneficial for reducing output variance; often referred as positive interactions [118].

Example 5.1 *To illustrate the findings of this section, we consider the example of Wood-Berry (WB) distillation column [143]. For this process, the overhead (y_1) and bottom (y_2) product compositions need to be controlled using reflux (u_1) and steam (u_2) flow rates, while the feed flowrate (a) is considered to be the disturbance. The continuous-time model is discretized with a sampling time of $\Delta T = 1$ minute to get*

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{0.744q^{-2}}{1-0.942q^{-1}} & \frac{-0.879q^{-4}}{1-0.954q^{-1}} \\ \frac{0.579q^{-8}}{1-0.912q^{-1}} & \frac{-1.302q^{-4}}{1-0.933q^{-1}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} \frac{0.247q^{-9}}{1-0.935q^{-1}} \\ \frac{0.358q^{-4}}{1-0.927q^{-1}} \end{bmatrix} a \quad (5.16)$$

The relative gain array (RGA) [11] suggests the use of diagonal pairings. The following decentralized PI controller is tuned using internal model control (IMC) method [121]

$$K = \begin{bmatrix} \frac{0.652-0.571q^{-1}}{1-q^{-1}} & 0 \\ 0 & \frac{-0.124+0.115q^{-1}}{1-q^{-1}} \end{bmatrix} \quad (5.17)$$

which provides $\text{Var}(y_1) = 0.122$ and $\text{Var}(y_2) = 0.759$. The least achievable variances of y_1 and y_2 according to the conventional approach discussed in Section 5.2.1 are $J_{1,\text{siso}} = 0.114$ and $J_{2,\text{siso}} = 0.413$, respectively. Accordingly, the performance indices for loops 1 and 2 are $\eta_1 = 0.932$ and $\eta_2 = 0.544$, respectively, which indicate that the variance of y_2 can be reduced significantly by re-tuning K_2 , but tuning K_1 will not reduce variance of y_1 significantly. Using the modified MV benchmark, we next show that this conclusion is not entirely correct.

We have $d_{11} = 2$, $d_{12} = 4$, $d_{21} = 8$ and $d_{22} = 4$. Thus, $d'_1 = \min(d_{11}, d_{12} + d_{21}) = 2$ and $d'_2 = \min(d_{22}, d_{12} + d_{21}) = 4$. Although $P_{d2} \neq H_2$, the first $d'_2 = d_{22}$ impulse response coefficients of P_{d2} and H_2 are the same; see Figure 5.4. Thus, we find that $J_{2,\text{decen}} = J_{2,\text{rmsiso}} = 0.413$ and $\eta_{2,\text{decen}} = \eta_{2,\text{siso}} = 0.544$. P_{d1} contains two unstable zeros, which are reflected across the unit circle to obtain $P_{d1,m}$. From Figure 5.4, we note that the impulse response coefficients of $P_{d1,m}$ decay much faster than the impulse response

Figure 5.4: Impulse response coefficients of disturbance models of open-loop and partially closed-loop systems for Example 5.1

coefficients of H_1 . Based on the first d'_1 impulse response coefficients of $P_{d1,m}$, we find that $J_{1,\text{decen}} = 0.062$ and $\eta_{1,\text{decen}} = 0.507$. Thus, the modified MV benchmark identifies that the variance of y_1 can be reduced by half through tuning of K_1 . This observation is in direct contrast to the findings of the conventional approach, which is based on the MV benchmark for SISO processes. The large difference between the performance indices computed using conventional and modified approaches is seen, as the former approach uses the impulse response coefficients of the open-loop disturbance model (H_1) instead of the disturbance model for the partially-closed loop system (P_{d1}). We point out that for this process, the first d'_1 impulse response coefficients of $P_{d1,m}$ and thus $J_{1,\text{decen}}$ depend on K_2 . For example, when the gain of K_2 is decreased by a factor of 0.25, $J_{1,\text{decen}}$ increases to 0.1. With this controller tuning, the variance of y_1 is 0.151. Thus, we have $\eta_{1,\text{decen}} = 0.665$, which indicates that the variance of y_1 can still be reduced by approximately 50%.

Example 5.1 demonstrates that the MV benchmark for SISO processes can overestimate

the performance index for decentralized controllers, as the change in disturbance model due to loop interactions is not accounted for. Furthermore, for Example 5.1, the effective delays for both loops are the same as open-loop delays, *i.e.* $d'_1 = d_{11}$ and $d'_2 = d_{22}$. This is not true in general, as shown next.

Example 5.2 *We consider the following process model taken from [26]:*

$$G = \begin{bmatrix} \frac{0.0347q^{-6}+0.0287q^{-7}}{1-1.542q^{-1}+0.5674q^{-2}} & \frac{1.106q^{-1}}{1-0.7788q^{-1}} \\ \frac{0.2835q^{-1}}{1-0.7165q^{-1}} & \frac{-0.1951q^{-7}}{1-0.9512q^{-1}} \end{bmatrix} \quad (5.18)$$

where the continuous-time model has been discretized with a sampling time of $\Delta T = 1$ unit. We assume the disturbance model to be an integrator, *i.e.* $H = 1/(1 - q^{-1})I$, where I is an identity matrix. The steady state RGA analysis [11] suggests the use of diagonal pairing, although one may notice that the delay corresponding to the diagonal elements of the process is significantly higher than its off-diagonal elements. As before, the following decentralized PI controller is designed using the IMC method using the continuous-time model and the recommended value of τ_c for both loops [121]:

$$K = \begin{bmatrix} \frac{0.533-0.5q^{-1}}{1-q^{-1}} & 0 \\ 0 & \frac{-0.417+0.396q^{-1}}{1-q^{-1}} \end{bmatrix} \quad (5.19)$$

which provides $\text{Var}(y_1) = 30.713$ and $\text{Var}(y_2) = 4.346$.

As the disturbance model is a pure integrator, the MV benchmark for SISO processes is the same as the time delay. Thus, according to the conventional approach, the least achievable variances of y_1 and y_2 are $J_{1,\text{siso}} = 6$ and $J_{2,\text{siso}} = 7$, respectively. This is clearly incorrect, as the actual variance of the y_2 is less than the indicated least achievable output variance leading to a performance index larger than 1. Next, we show that this discrepancy can be resolved by taking the loop interactions into account.

Using the proposed method, we have $d'_1 = \min(d_{11}, d_{12} + d_{21}) = 2$ and $d'_2 =$

Figure 5.5: Impulse response coefficients of disturbance models of open-loop and partially closed-loop systems for Example 5.2

$\min(d_{22}, d_{12} + d_{21}) = 2$, which shows that the effective delays for the two loops are lower than the corresponding open-loop delays. In addition, the impulse response coefficients of the disturbance models for the partially closed loop system (P_{d1} and P_{d2}) differ from the open loop disturbance models (H_1 and H_2); see Figure 5.5. For Loop 2, the impulse response coefficients of P_{d2} are much smaller than the impulse response coefficient of H_2 indicating the presence of positive interactions, i.e. performance enhancement due to the interactions from other loop.. Based on (5.15), the least achievable variances are calculated to be $J_{1,\text{decen}} = 3.134$ and $J_{2,\text{decen}} = 1.721$, and thus $\eta_{1,\text{decen}} = 0.102$ and $\eta_{2,\text{decen}} = 0.396$. This result indicates that the variance of the second loop can still be reduced by a factor of 2.5 through controller retuning, where as conventional approach significantly overestimates the performance index.

5.2.3 Generalization to $n_y \times n_y$ Systems

To generalize the modified MV benchmark for loop by loop analysis to $n_y \times n_y$ systems, let us define

$$\tilde{G}_{12} = \begin{bmatrix} G_{12} & G_{13} & \cdots & G_{1n_y} \end{bmatrix} \quad (5.20)$$

$$\tilde{G}_{21} = \begin{bmatrix} G_{21} & G_{31} & \cdots & G_{n_y1} \end{bmatrix}^T \quad (5.21)$$

$$\tilde{G}_{22} = \begin{bmatrix} G_{22} & G_{23} & \cdots & G_{2n_y} \\ \vdots & \vdots & \cdots & \vdots \\ G_{n_y2} & G_{n_y3} & \cdots & G_{n_y n_y} \end{bmatrix} \quad (5.22)$$

$$\tilde{H}_1 = \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1n_d} \end{bmatrix} \quad (5.23)$$

$$\tilde{H}_2 = \begin{bmatrix} H_{21} & H_{22} & \cdots & H_{2n_d} \\ \vdots & \vdots & \cdots & \vdots \\ H_{n_y1} & H_{n_y2} & \cdots & H_{n_y n_d} \end{bmatrix} \quad (5.24)$$

$$\tilde{K}_2 = \text{diag}(K_2, K_3, \cdots, K_{n_y}) \quad (5.25)$$

When all other loops are closed, y_1 is still given by (5.8), but the expressions for P_{11} and P_{d1} change as

$$P_{11} = G_{11} - \tilde{G}_{12} \tilde{K}_2 (I + \tilde{G}_{22} \tilde{K}_2)^{-1} \tilde{G}_{21} \quad (5.26)$$

$$P_{d1} = \tilde{H}_1 - \tilde{G}_{12} \tilde{K}_2 (I + \tilde{G}_{22} \tilde{K}_2)^{-1} \tilde{H}_2 \quad (5.27)$$

Thus, $J_{1,\text{decen}}$ can be found by using the Diophantine identity for minimum phase version of P_{d1} . Here, the effective delay d'_1 needs to be calculated as

$$d'_1 = \min(d_{11}, \min_{k,k \neq 1} d_{1k} + \min_{\ell, \ell \neq 1} d_{\ell 1}) \quad (5.28)$$

which represents the delay of the fastest among $(n_y - 1)^2 + 1$ paths through which u_1 can affect y_1 .

Example 5.3 We consider the energy balance control for interacting sidestream column/stripper distillation column example taken from [88]. For this process, the outputs are overhead ethanol mole fraction (y_1), side-stream ethanol mole fraction (y_2), and temperature on the 19th tray (y_3). The available inputs are reflux flowrate (u_1), sidestream product flowrate (u_2), and reboiler steam pressure (u_3), and the disturbances are feed flowrate (a_1) and feed temperature (a_2). The discrete time transfer models of the process and disturbance are obtained with a sampling time of 1 minute [88] and are given as:

$$G = \begin{bmatrix} \frac{0.0383q^{-3}+0.0533q^{-4}}{1-0.8614q^{-1}} & \frac{-0.0343q^{-4}-0.0324q^{-5}}{1-0.8907q^{-1}} & \frac{-5.12 \times 10^{-4}q^{-2}}{1-0.8955q^{-1}} \\ \frac{0.1583q^{-7}+0.1357q^{-8}}{1-0.7351q^{-1}} & \frac{-0.4278q^{-4}}{1-0.8187q^{-1}} & \frac{-1.28 \times 10^{-3}q^{-2}-2.98 \times 10^{-4}q^{-3}}{1-0.8665q^{-1}} \\ \frac{-3.2420q^{-10}-0.7621q^{-11}}{1-0.8845q^{-1}} & \frac{2.4740q^{-10}+1.5760q^{-11}}{1-0.9123q^{-1}} & \frac{0.1238q^{-2}-0.1136q^{-3}}{1-1.7220q^{-1}+0.7333q^{-2}} \end{bmatrix} \quad (5.29)$$

$$H = \begin{bmatrix} \frac{0.0093q^{-13}}{1-0.851q^{-1}} & \frac{-3.73 \times 10^{-4}q^{-3}+3.10 \times 10^{-4}q^{-5}+6.45 \times 10^{-4}q^{-5}}{1-1.814q^{-1}+0.8228q^{-2}} \\ \frac{0.0166q^{-11}+0.0154q^{-12}}{1-0.8651q^{-1}} & \frac{-2.28 \times 10^{-3}q^{-4}+5.45 \times 10^{-4}q^{-5}+1.54 \times 10^{-3}q^{-6}}{1-1.7660q^{-1}+0.7796q^{-2}} \\ \frac{-0.2545q^{-2}-0.3584q^{-3}}{1-0.8812q^{-1}} & \frac{0.0719q^{-3}+0.1011q^{-4}}{1-0.8791q^{-1}} \end{bmatrix} \quad (5.30)$$

The decentralized PI controller is designed using IMC method [121] based on the diagonal elements of the process. For this process, $g_{33}(s)$ has second order and is thus approximated as a first order transfer function using the model reduction techniques discussed in [121] prior to controller design. Furthermore, the use of recommended values of τ_c for the three loops results in a very aggressive controller. Using trial and error, τ_c is chosen as $55\tilde{\theta}_{11}$, $18\tilde{\theta}_{22}$ and $6\tilde{\theta}_{33}$ for loops 1, 2 and 3, respectively, where $\tilde{\theta}_{ii} = \theta_{ii}$ with θ_{ii} being the time delay of the first-order approximation of $g_{ii}(s)$. The resulting decentralized

PI controller is given as:

$$K = \begin{bmatrix} \frac{0.0697-0.0593q^{-1}}{1-q^{-1}} & 0 & 0 \\ 0 & \frac{-0.0372+0.0297q^{-1}}{1-q^{-1}} & 0 \\ 0 & 0 & \frac{1.0340-0.7685q^{-1}}{1-q^{-1}} \end{bmatrix} \quad (5.31)$$

For diagonal pairing, we have $d_1 = d'_1 = 3$; $d_2 = d'_2 = 4$ and $d_3 = d'_3 = 2$. As the expected magnitudes of the two disturbances are 0.2 and 20, respectively [88], H is scaled by $\text{diag}(\sqrt{0.2}, \sqrt{20})$ to satisfy the assumption of disturbances having unit variance. For the process, every element of P_d contains finite zeros outside the unit circle. Thus, prior to the analysis, these unstable zeros are reflected across the unit circle to obtain $P_{d,m}$. The impulse response coefficients of $P_{d,m}$ and H are shown in Figure 5.6 and the MV benchmarks obtained using the conventional and modified approach are shown in Table 5.1.

Loop	Var(y)	Conventional		Modified	
		$J_{i,\text{siso}}$	$\eta_{i,\text{siso}}$	$J_{i,\text{decen}}$	$\eta_{i,\text{decen}}$
1	1.87×10^{-4}	1.97×10^{-4}	1.0586	1.56×10^{-4}	0.8332
2	2.31×10^{-3}	2.35×10^{-3}	1.0158	1.88×10^{-3}	0.8144
3	1.1887	0.4364	0.3671	0.4898	0.4120

Table 5.1: Performance assessment of sidestream column/stripper distillation column using loop by loop analysis

It can be noted that similar to Example 5.2, for conventional approach, $J_{i,\text{decen}}$ exceeds the observed output variance for Loops 1 and 2, which signifies a limitation of using MV benchmark for performance assessment of decentralized controllers. In comparison, the performance indices obtained using the modified MV benchmark for these loops is closer but still smaller than 1. Furthermore, unlike Example 5.1, $\eta_{3,\text{siso}}$ obtained using the conventional approach is lower than $\eta_{3,\text{decen}}$ obtained using the modified approach. This shows that, in general, the MV benchmark for SISO processes can also underestimate the performance of decentralized controllers.

Figure 5.6: Impulse response coefficients of disturbance models of open-loop and partially closed-loop systems for Example 5.3

5.3 Estimation from Data

A limitation of the proposed approach is the requirement of exact process model, which may not be available in practice. In this section, we show that with the knowledge of time delays between all the inputs and outputs, the modified MV benchmark can be estimated from closed loop data.

When all loops are closed under feedback control, *i.e.* $u = -K y$, we have

$$y = (I + G K)^{-1} H a \quad (5.32)$$

or equivalently

$$y = \begin{bmatrix} 1 + G_{11} K_1 & \tilde{G}_{12} \tilde{K}_2 \\ \tilde{G}_{21} K_1 & I + \tilde{G}_{22} \tilde{K}_2 \end{bmatrix}^{-1} \begin{bmatrix} \tilde{H}_1 \\ \tilde{H}_2 \end{bmatrix} a \quad (5.33)$$

where $\tilde{G}_{12}$, $\tilde{G}_{21}$, $\tilde{G}_{22}$, $\tilde{H}_2$, and $\tilde{K}_2$ are given in (5.20)– (5.25). Using the formula for inversion of partitioned matrices, we have [42]

$$y_1 = X^{-1} \left(\tilde{H}_1 - \tilde{G}_{12} \tilde{K}_2 \left(I + \tilde{G}_{22} \tilde{K}_2 \right)^{-1} \tilde{H}_2 \right) a \quad (5.34)$$

where y_1 is the first element of y and

$$X = 1 + G_{11} K_1 - \tilde{G}_{12} \tilde{K}_2 \left(I + \tilde{G}_{22} \tilde{K}_2 \right)^{-1} \tilde{G}_{21} K_1 \quad (5.35)$$

$$= 1 + \left(G_{11} - \tilde{G}_{12} \tilde{K}_2 \left(I + \tilde{G}_{22} \tilde{K}_2 \right)^{-1} \tilde{G}_{21} \right) K_1 \quad (5.36)$$

Thus, based on (5.26)– (5.27) and (5.36),

$$y_1 = (1 + P_{11} K_1)^{-1} P_{d1} a \quad (5.37)$$

$$= (1 + P_{11} K_1)^{-1} P_{d1,m} \bar{a} \quad (5.38)$$

Let $M_1 = K_1 / (1 + P_{11} K_1)$ (Youla parametrization). Then,

$$y_1 = (1 - P_{11} M_1) P_{d1,m} \bar{a} \quad (5.39)$$

$$= F'_1 \bar{a} - q^{-d'_1} (\bar{P}_{11} M_1 P_{d1,m} - R'_1) \bar{a} \quad (5.40)$$

which shows that the first d'_1 impulse response coefficients of the closed-loop transfer function between y_1 and $\bar{a}$ are the same as open-loop transfer function. Thus, F'_1 and $\eta_{i,\text{decen}}$ can be estimated using closed loop data, *e.g.* using the FCOR algorithm [45], with *a priori* knowledge of the delays of g_{ij} for all $i, j = 1, 2, \dots, n_y$.

It is interesting to note from (5.40) that the estimation of MV benchmark using closed-loop data leads to estimation of impulse response coefficients of P_{d1} and not H_1 , as is traditionally believed. This observation highlights that although the application of MV benchmark derived for SISO controller to decentralized controllers is theoretically not justified, the estimation using closed-loop data fortunately still leads to the correct impulse response coefficients. The issue regarding the number of the impulse response coefficients to be considered for estimation of MV benchmark, however, still remains. To this end, we point out that the inputs and outputs are often paired based on their physical proximity. For such loops, the effective delay is the same as the open-loop delay. When the output of the loop under consideration, however, is not paired with the closest input, the effective delay is different from open-loop delay. Thus a suitable approach for industrial application of the proposed MV benchmark is to identify such non-trivial loops and find the correct effective delay before the estimation of MV benchmark using closed-loop data.

Remark 5.1 *In general, P_{11} can contain finite poles or zeros outside the unit circle, even though all the elements of G are stable and minimum-phase. For example, the unstable poles may arise, when the pairings are selected on negative elements of RGA [11]. Similar to the MV benchmark for SISO systems [45], the modified MV benchmark can be obtained by mirroring the unstable poles and zeros of P_{11} across the unit circle using all-pass factor. In general, however, the locations of these unstable poles and zeros depend on the controller settings for other loops [16] and thus the estimation of modified MV benchmark from operating data can be difficult.*

Example 5.4 *We revisit Example 5.3 with PI controller settings given in (5.31). The*

closed-loop system is simulated, where the 2 disturbances are assumed to be zero mean random noise sequences with variances being 0.2 and 20, respectively. The observed output variances for the 3 loops are 1.90×10^{-4} , 2.31×10^{-3} and 1.2227, respectively. For each loop, the output signals are pre-whitened to obtain the following estimates of F'_i , i.e. $\hat{F}'_i$,

$$\hat{F}'_1 = 0.0090 + 0.0070q^{-1} + 0.0051q^{-2} \quad (5.41)$$

$$\hat{F}'_2 = 0.0165 + 0.0286q^{-1} + 0.0228q^{-2} + 0.0178q^{-3} \quad (5.42)$$

$$\hat{F}'_3 = 0.3773 + 0.5852q^{-1} \quad (5.43)$$

The estimates of the performance indices $\hat{\eta}_i$ are computed as $\hat{\eta}_i = \|\hat{F}'_1\|_2^2 / \text{Var}(y_i)$, which are found to be 0.8343, 0.8285 and 0.3965 for the 3 loops. We note that the estimated values of performance indices are within $\pm 5\%$ of the theoretical values shown in Table 5.1. The same analysis is repeated 100 times, for which $\hat{\eta}_i = 0.8519 \pm 0.0130, 0.8234 \pm 0.0201$ and 0.4178 ± 0.0136 for $i = 1, 2$ and 3, respectively, which shows that the modified MV benchmark can be reliably estimated from the closed loop data.

5.4 Chapter Summary

In this chapter, we proposed a practical approach for performance assessment of decentralized controller using MV benchmark. We have shown that the use of MV benchmark for SISO system (where the interactions are neglected) may give incorrect conclusions regarding the controller performance and the opportunity for variance reduction through re-tuning of the controller. We proposed a modified MV benchmark, where process interactions are explicitly taken into account. Using numerical examples, it is shown that the modified benchmark can be reliably estimated using closed-loop operational data, which is vital for the application of this technique in process industries.

Chapter 6

An Efficient Genetic Algorithm for Input-Output Pairing Selection

6.1 Introduction

The design of a decentralized controller requires selection of pairings among controlled and manipulated variables. The input output (IO) pairing selection has received a lot of interest during the past few decades. The relative gain method [11] is often regarded as the first systematic method for pairing selection. In the last three decades, a number of other methods for deciding upon the IO pairings have also been proposed, such as **effective relative gain array (ERGA) [146, 147], Normalized RGA [39], relative interaction array [154], decomposed relative interaction array [38, 40], μ -interaction measure [26, 27], ν -gap metric [106] and dissipativity [4, 5, 149]**. Comprehensive reviews on the methods available for IO pairing selection are available in [135–137].

A drawback of most of the available pairing selection methods is that they require evaluation of all possible alternatives. We note that as the system dimension increases,

the number of alternatives increases tremendously. For example, a 5×5 system has 120 pairing alternatives, while a 10×10 system has more than 3.5 millions alternatives. For large scale process, IO pairing selection through evaluation of all possible alternatives is an impossible task.

To handle the issue of large number of pairing alternatives, it is necessary to develop systematic methods for efficient selection of optimal pairings. Unfortunately, this issue has only received limited attention until recently. Kookos and Perkins [70] developed a mixed-integer linear programming (MILP) approach to find the optimal pairings based on RGA [11]. In this formulation, decision variables can take the values of 1 and 0, which represent paired and unpaired variables, respectively. This approach, however, cannot handle some other criteria, such as μ -IM [26], as these criteria cannot be cast as MILPs. Naturally, these other criteria can be handled using mixed-integer nonlinear programming (MINLP) techniques, but this approach can usually only provide locally optimal solution and is computationally expensive. Recently, Kariwala and Cao [59] proposed a branch and bound (BAB) algorithm to handle criteria like RGA and μ -IM in a single framework. BAB gains advantage over complete enumeration by pruning the alternatives which cannot lead to the optimal solution, and guarantees global optimality. This approach has been shown to provide optimal pairings for a 20×20 process in one minute, whereas for the same process screening of all possible alternatives would take more than 3 million years [59]. Though effective, the BAB method requires derivation of tight bounds on the selection criteria to be used for pruning, which can be difficult in general, if not impossible.

In this chapter, the potential of genetic algorithm (GA) is explored for efficient selection of pairings. GA is a well-established method used in many fields such as mathematics, engineering, medicine, and political science [23]. In comparison with other algorithms, GA may not lead to the globally optimum solution, however, it can often find practical (nearly optimal) solutions, while requiring low computational cost.

We develop a GA for pairing selection using a single criterion. For this purpose, customized crossover and mutation operators are developed. The proposed approach is illustrated using RGA-number [124] and μ -IM [26] as the selection criteria. The efficiency of the developed method is compared with other available methods, such as MILP approach [70] and BAB [59] using extensive numerical tests.

6.2 Genetic Algorithm for Pairing Selection

6.2.1 General Principle

Genetic algorithm is a search or optimization method based on evolution principles [17, 23]. This is a population-based search method. Each individual in the population carries information contained in genes. Each individual is also assigned a fitness value based on the selection criteria. The algorithm starts with random selection of several individuals to form a population, called initial population. Some of these individuals are chosen as parents. Subsequently, the crossover and mutation operators are applied to the parents to obtain the population for the next generation. The procedure is repeated until a pre-defined termination criteria is achieved. The commonly used termination criteria are the maximum number of generations and the change in the fitness value (average or best). A flow diagram of GA is shown in Figure 6.1.

For parent selection, two algorithms are commonly used, (1) Tournament selection, where two or more individuals compete to be selected as parents based on their constraint satisfaction and fitness values; and (2) Roulette wheel selection, where the probability of each individual to be chosen as parent is proportional to its relative fitness function, defined as its fitness compared to the sum of fitness of the entire population. Despite its potential to provide better constraint handling, tournament selection requires more computational effort

Figure 6.1: Flow diagram of genetic algorithm

as every pair of individuals which compete to be a parent need to be analyzed separately. In comparison, roulette wheel algorithm enables selecting all parents together. Thus, in this work, roulette wheel selection is chosen for parent selection.

The crossover algorithm involves genes exchanges between two individuals (parents) to form new individuals (children). The crossover operator aim at obtaining better individuals who inherit good features from their parents. On the other hand, mutation involves exchanging two randomly selected genes of an individual to form a new individual. The mutation operator is designed to retain the scatterness of the population and escaping the local minimum solution.

Since the crossover and mutation operators are stochastic in nature, the quality of solution may deteriorate in subsequent generations. To avoid this, elitism is used, where the best solution is always retained in the population.

6.2.2 Application to Pairing Selection

Next, the general GA method is customized for pairing selection. A critical step in development of an efficient GA is representation of individuals. We note that the pairing selection problem is similar to the traveling salesman problem (TSP) [72] and can be considered to be a permutation problem. For standardization of notation, we consider an $n \times n$ system. It is considered that $N_n = \{1, 2, \dots, n\}$ denotes the set of first n natural numbers and P_n is any arbitrary permutation of N_n which is considered as a pairing alternative, where the integer represents the input and its order in the permuted set represents the output which is paired with the corresponding input. For example, $\{2, 3, 1\}$ implies that the second input is paired with the first output, third input with the second output, and first input with the third output. This strategy allows for a compact representation of the pairing alternatives and does not produce infeasible pairings.

In this work, the selection, crossover and mutation operators are defined as follows:

- **Selection:** The parents for generating individuals for the next generation are selected based on their fitnesses. The fitness of each individual is normalized based on the average fitness and its standard deviation [81]. as

$$\hat{f}_i = \frac{f_i - \bar{f}}{\sigma} \quad (6.1)$$

where f_i , $\hat{f}_i$, $\bar{f}$, and σ , denote fitness value of the i^{th} individual, its normalized value, mean of fitness values, and the standard deviation, respectively. The normalized fitness dictates the possibility of each individual to be selected as a parent for

generating individuals for subsequent generation.

The possibility of each individual to be selected as a parent in the next generation is calculated as.

$$P(i) = \frac{\hat{f}_i}{\sum_{j=1}^N \hat{f}_j} \quad (6.2)$$

where N denotes the population size. A few best individuals in the current population are copied to the next generation to ensure it contains individual with the same quality as the previous generation. The percentage of best individuals being copied to the next generation is $(1 - \text{crossover probability}) \times 100\%$

- **Crossover:** As previously discussed, the crossover involving gene exchanges between two parents to form two children. We note that all the genes of the children must be unique to have a feasible pairing assignment. A special algorithm, called cycle crossover [17] ensures this. The procedure is described next.

1. The algorithm starts by copying any single randomly selected gene of first and second parent to the first and second children, respectively.
2. Subsequently, it finds the position of the copied genes of second parent in the first parent. Then, the genes of both parents in the particular position is copied to the first and second child.
3. The procedure is repeated until it finds the position of the first parent's gene which was copied first.
4. The empty positions of the first and second children are copied based on the second and first parent.

For illustration of the crossover algorithm, we consider pairing assignment for a 5×5 system. Let the first and second parents be $P_1 = \{1, 2, 3, 4, 5\}$ and $P_2 = \{4, 3, 2, 1, 5\}$, respectively. Let the initial gene to be copied is the first gene.

Thus, the first and second child are $C_1 = \{1, *, *, *, *\}$ and $C_2 = \{4, *, *, *, *\}$, respectively. The copied gene to the second parent (4) is the fourth gene of the first parent. Thus, the fourth gene of each parent is copied to the children to get $C_1 = \{1, *, *, 4, *\}$ and $C_2 = \{4, *, *, 1, *\}$. Note that the fourth gene of the second child (4) is first gene being copied from the first parent. Thus, the cycle is complete, and the remaining gene of first and second child are copied from their second and first parent, respectively. The complete genes of the children are $C_1 = \{1, 3, 2, 4, 5\}$ and $C_2 = \{4, 2, 3, 5, 1\}$.

- **Mutation:** The mutation operator involves exchanging two randomly selected gene of an individual to form a new individual. For example, suppose that initially the individual is $\{1, 2, 3, 4, 5\}$. The mutated individual after swapping the first and third genes are $\{3, 2, 1, 4, 5\}$.

6.3 Input-Output Pairing Selection Method

Many methods are available for pairing selection for stable linear multivariable systems. In this chapter, two methods, namely relative gain array [11] and μ -interaction measure [26] are considered. Both methods are commonly used for practical decentralized control design.

6.3.1 Relative Gain Array

RGA is the most commonly used approach for pairing selection. For an $n \times n$ process, RGA is defined as [122, 123]

$$\Lambda = G(0) \circ G(0)^{-T} \quad (6.3)$$

where $\circ$ denotes element-by-element multiplication and $G(0)$ is the steady-state gain matrix. The pairings are selected based on the following rules:

- Prefer pairings on RGA elements, which are close to 1.
- Avoid pairings on negative RGA elements.

The first rule is based on the interpretation of RGA as an interaction measure, where the proximity of RGA element to 1 indicates weak interaction. The second rule is used to ensure integrity against loop failure [28].

While for a small process (3×3 or 4×4), the best pairing can be obtained from inspection of RGA, this is difficult for large scale processes. To simplify pairing selection based on RGA, Skogestad and Postlethwaite [124] introduced the concept of RGA-number, which is defined as

$$RGA - number = \|\Lambda - I\|_{sum} \quad (6.4)$$

where I is the identity matrix. We note that permuting G results in same permutation of Λ [11]. Thus, the optimum input-output pairings can be found by solving the following optimization problem:

$$\min_{P_n} \|\Lambda_{P_n} - I\|_{sum} \quad (6.5)$$

$$\text{s.t.} \quad [\lambda_{P_n}]_{ii} > 0 \quad i = 1, 2, \dots, n \quad (6.6)$$

where Λ_{P_n} is the RGA matrix, whose columns are permuted by P_n and P_n is a permutation of N_n . Now, the optimal pairings can be found using GA where the fitness is defined in (6.4), while (6.6) denotes the constraint.

In addition, there are several variants of RGA available in the literature, such as ERGA [146, 147], Normalized RGA [39], and decomposed relative interaction array [38, 40]. For these methods, genetic algorithm can be implemented in a similar fashion as used for RGA in this chapter. For example, we can define ERGA-number similar to (6.4) and the pairings can be selected by minimizing ERGA number subject to positive RGA element as

described in (6.5) - (6.6).

6.3.2 μ -Interaction Measure

To illustrate the principle of μ -IM, let $\tilde{G}(s)$ be a diagonal transfer matrix containing the diagonal elements of $G(s)$. Then a decentralized controller $K(s)$ stabilizing $\tilde{G}(s)$ also stabilizes $G(s)$ if [26]

$$\bar{\sigma}(\tilde{T}(s)) < \mu_{\Delta}^{-1}(E(s)) \quad (6.7)$$

where $\bar{\sigma}$ denotes the largest singular value, $\tilde{T}(s)$ is the complimentary sensitivity function of $\tilde{G}(s)$, *i.e.* $\tilde{T}(s) = \tilde{G}K(I + \tilde{G}K)^{-1}$, μ_{Δ} is the structured singular value computed with a diagonal perturbation Δ [124], and $E(s)$ is defined as

$$E(s) = \left(G(s)\tilde{G}(s)^{-1} - I \right) \quad (6.8)$$

According to (6.7), the pairing should be chosen such that $\mu_{\Delta}(E(s))$ is minimized. Minimizing $\mu_{\Delta}(E(s))$ facilitates satisfying (6.7). In addition, $\mu_{\Delta}(E(s))$ is also a measure of diagonal dominance [60, 124].

The optimal pairings based on μ -IM can be selected by solving

$$\min_{P_n} \mu_{\Delta}^{-1}(E(s)) = \min_{P_n} \mu_{\Delta}^{-1}(G_{P_n}(s)\tilde{G}_{P_n}^{-1}(s) - I) \quad (6.9)$$

$$\text{s.t.} \quad [\lambda_{P_n}]_{ii} > 0 \quad i = 1, 2, \dots, n \quad (6.10)$$

where $G_{P_n}(s)$ is the permuted process with column indexed by P_n . The constraint on positive RGA element is still imposed to ensure integrity against loop failure.

6.4 Numerical Tests

The proposed GA for pairing selection is evaluated through numerical examples. The computation is carried out using a desktop PC with Intel®Core™Duo Processor 6600 (2.40GHz, 2GB RAM). For this case, the crossover and mutation probabilities are 80% and 1%, respectively. The result is evaluated based on the computational time and accuracy. The result is compared with MILP method [70], BAB [59], and Bidirectional Branch and bound (B3) [60]. The accuracy is calculated using the following expression:

$$\text{Error} = \frac{P_{opt} - P_{ga}}{P_{opt}} \quad (6.11)$$

where P_{opt} and P_{ga} are the corresponding values of the best pairings found using B3 algorithm [60] and sub-optimal pairings obtained using GA algorithm, respectively.

First, we consider pairing selection based on RGA. In this test, a set of randomly generated matrices, ranging from 2×2 to 70×70 are used. As the number of pairing alternatives increases when the matrix dimension increase, larger population size and generations are necessary. In this case, the population size and number of generations are taken as $\max(10, n)$ and n , respectively, where n denotes the matrix dimension. For each n the average computational time for each algorithm is shown in Figure 6.2. Its relative error in comparison with the optimum solution found using Hungarian algorithm [13] is shown in Figure 6.3.

Some remarks for pairing selection problem using GA based on RGA approach based on Figure 6.2 and Figure 6.3 is described as follow.

1. The computational time for GA is higher for small dimensional matrices. Nevertheless, the increase in computational time is smaller, such that it can handle matrices with higher dimension in a reasonable time.
2. The computational time of GA algorithm is significantly better than B3 and BAB,

Figure 6.2: Computational time for pairing assignment based on RGA

Figure 6.3: Relative error for GA method based on RGA

especially for large scale processes.

3. As expected, GA does not always find the global optimum solution. The relative error is approximately 20% for 2×2 process and tends to decrease when the process dimension increases. For sufficiently large process, the relative error is negligible.

The GA is also used to find pairings based on μ -IM. For pairing assignment based on μ -IM. The population size for each generation is $\max(n, 30)$ while other parameters are the same as used earlier. The average computational time and the relative error are shown in Figure 6.4 and Figure 6.5, respectively. The relative error data is not available for process larger than 15×15 , as the optimal solution can only be obtained using B3 method [60], which takes very long to find pairings for processes larger than 15×15 .

Figure 6.4: Computational time for pairing assignment based on μ -IM

For pairing assignment based on μ -IM, the increase in computational time using GA algorithm are least among other methods. It can find reasonably accurate answer for 15×15 system within reasonable margin of error (with an average of $< 5\%$). Nevertheless, unlike the pairing assignment based on RGA, the relative error tends to increase when the

Figure 6.5: Relative error for GA method based on μ -IM

dimension of the process increases.

6.5 Chapter Summary

In this chapter, we explored the potential of GA for efficient selection of IO pairings. A customized GA is developed and illustrated using RGA and μ -IM as the selection criteria. Based on numerical tests using randomly generated matrices, it is shown that GA can solve large scale process in reasonable amount of time with good accuracy.

6.A Appendix: Illustrative Example

To further illustrate the concept of tailoring GA for pairing selection, let us consider the example of a 4×4 pH neutralization process [107]. It has 24 pairing alternatives and the

RGA is given as:

$$\Lambda = \begin{bmatrix} 0.64 & 0.72 & -0.20 & -0.20 \\ 0.87 & 0.85 & -0.35 & -0.35 \\ -0.18 & -0.21 & 0.70 & 0.70 \\ -0.36 & -0.37 & 0.85 & 0.88 \end{bmatrix} \quad (6.12)$$

Let the population size and crossover probability be 4 and 100%, respectively. In addition, an individual is mutated on every generation. The following pairings are used as the initial population: $P_1 = \{1, 2, 3, 4\}$, $P_2 = \{2, 3, 4, 1\}$, $P_3 = \{3, 4, 1, 2\}$, and $P_4 = \{4, 1, 2, 3\}$. Table 6.1 shows the fitness value and the possibility of each individual to be chosen as parent for the subsequent generation.

Table 6.1: Selection procedure on the first generation.

Individual	f_i	$\hat{f}_i$	$\sum_{j=1}^N f_j$	$P(i)$
$\{1, 2, 3, 4\}$	-2.14	2.21	4.23	0.52
$\{2, 3, 4, 1\}$	1.16	0.89	4.23	0.21
$\{3, 4, 1, 2\}$	4.00	0.00	4.23	0.00
$\{4, 1, 2, 3\}$	0.56	1.13	4.23	0.27

Based on the values of $P(i)$ shown in Table 6.1, we assume that the parents for the second generation consist of P_1 , P_1 , P_2 , and P_4 . As the crossover probability is 100%, no individual from the first generation is being copied to the second generation. Let the children is produced by chromosome exchange between $P_1 - P_2$ and $P_1 - P_4$. By setting the first gene of each individual as the starting point, we have $K_1 = \{1, 4, 3, 2\}$, $K_2 = \{3, 2, 1, 4\}$, $K_3 = \{1, 2, 3, 4\}$, and $K_4 = \{4, 1, 2, 3\}$. In addition, the first and second genes of K_3 are swapped, such that $K_3 = \{2, 1, 3, 4\}$. Subsequent analysis of fitnesses of the second generations are shown in Table 6.2

Based on Table 6.2, better fitness value is observed in the second generation. In this particular generation, the second and third individuals have higher possibilities to be selected as parents compared to other individuals. Thus, we can assume that the set of

Table 6.2: Selection procedure on the second generation.

Individual	f_i	$\hat{f}_i$	$\sum_{j=1}^N f_j$	$P(i)$
$\{1,4,3,2\}$	1.32	0.48	4.00	0.12
$\{3,2,1,4\}$	0.54	0.79	4.00	0.20
$\{2,1,3,4\}$	-2.34	1.94	4.00	0.48
$\{4,1,2,3\}$	0.56	0.79	4.00	0.12

parents for the third generation only consists of K_2 and K_3 . By exchanging chromosome between K_2 and K_3 with the first gene as starting point, we have a new population consisting of K_2 and K_3 .

We note that the fitness of K_3 (1.94) is significantly larger than K_2 (0.79). Thus, K_3 has higher chance to be selected as a parent compared to K_2 . We can assume that eventually the population converges into $K_3 = \{2, 1, 3, 4\}$. The best pairing is found to be $y_2 - u_1$, $y_1 - u_2$, $y_3 - u_3$, and $y_4 - u_4$.

Chapter 7

Decentralized Control of Solid Oxide Fuel Cells

7.1 Introduction

The use of fossil fuels as major energy source has given rise to several concerns, such as the effect of harmful emissions and sustainability. Due to these reasons, a lot of efforts have been placed on finding alternative energy sources. One of the promising environmentally benign alternative for large-scale electricity production involves the use of fuel cells. A fuel cell uses hydrogen as its fuel to produce electricity by electrochemical reactions. A major advantage of fuel cells is that they only produce water as waste.

A fuel cell consists of an electrolyte and two electrodes, *i.e.* cathode and anode, where the reduction and oxidation take place, respectively. The electrolyte acts as a membrane which allows ions to travel between the electrodes and forces electrons to travel via an

¹Part of this chapter was presented in Regional Symposium of Chemical Engineering, Kuala Lumpur, Malaysia, 2008 [113].

²The contents of this chapter have been submitted to *IEEE Transaction on Industrial Informatics*.

external circuit to complete the reaction and hence produce electricity. The electrolyte usually consists of Yttria-stabilized Zirconia, a ceramic material which conducts oxygen ion. Some other metal oxide electrolytes, such as iron (III) oxide, platinum, and calcium oxide, have also been used earlier [150].

One of the most promising types of fuel cell is the solid oxide fuel cell (SOFC) due to its solid-state design, internal reforming of gaseous fuels, and high efficiency [119]. In addition, it produces heat as a by-product which can be used as an energy source in heat exchangers. With these attractive features, several practical applications of SOFC have been reported in the past few years [119].

In order to understand the behavior of SOFC, simplified dynamic models have recently been proposed [8, 14, 30, 53, 74, 92, 153]. These models are derived analytically based on mass and energy balances as well as electron transfer within the fuel cell [9]. More advanced dynamic models include the effects of impedance [97], heat transfer [108], fluid dynamics [97, 98] and polarization [29]. Bhattacharya and Rengaswamy [6] provide a comprehensive survey of the available dynamic models of SOFC.

The operational objectives of SOFC include tracking varying power demand while maintaining some key variables, such as voltage and fuel utilization, within their allowable bounds. Another objective is fast transient response of SOFC in the presence of changes in load. Previous studies have shown that control system design for SOFC is challenging due to its slow response time, tight operating constraints and unavailability of some key measurements [101]. The challenges and demanding operational objectives of SOFC have motivated researchers to propose the use of advanced control strategies like $\mathcal{H}_\infty$ controller [64], model predictive controller [50, 54, 55, 144], data-driven predictive control [139] and nonlinear MPC [7].

In comparison with the aforesaid control strategies, decentralized PID controllers are used more frequently in process industries. A decentralized controller maintains the

simplicity of the control system, is easier to tune and maintain, provides flexibility and can be made fault-tolerant more easily [124]. With these advantages, decentralized PID controllers have been used for control of different configurations of SOFC [7, 29, 126] and hybrid SOFC-Gas turbine system [102].

The aforesaid studies [7, 29, 102, 126] are based on different models of SOFC. An objective of this work is to systematically design a decentralized control system and compare its performance with advanced control strategies. Thus, we use the benchmark non-linear dynamic model of SOFC proposed in [92], which has been earlier used as the basis for the design of $\mathcal{H}_\infty$ controller [64] and data-driven MPC [139]. The design procedure involves selection of controlled variables (CVs), input-output pairing selection, and controller tuning. The tasks of pairing selection and PID controlled tuning are carried using well-established tools like RGA analysis [11] and SIMC-PID tuning method [121], respectively. The choice of appropriate CVs, however, is not straightforward.

Recently, Skogestad [120] introduced the concept of self-optimizing control for selection of CVs as a subset or linear combination of available measurements [3, 31, 58, 61]. In this approach, CVs are selected such that holding the selected CVs at constant setpoints incurs minimal loss (typically measured in terms of process economics), in comparison with the use of an online optimizer. The available methods for self-optimizing control focus on CV selection for the unconstrained problem. On the other hand, a major difficulty in the use of decentralized PID controller for SOFC is the presence of tight operating constraints, which cannot be handled explicitly by the PID controllers. With this observation, we propose a method where CVs are selected such that maintaining the selected CVs at constant setpoints facilitates the constraint satisfaction for all key variables. Using this novel method, two alternate decentralized control structures are proposed, which use a subset and linear combination of the measurements as CVs, respectively. The performance of the proposed control structures are evaluated using rigorous closed-loop simulations,

which show that the designed decentralized controller can closely meet the operational objectives of SOFC, even in presence of variations in the operating point.

Figure 7.1: Dynamic model of solid oxide fuel cells

7.2 Nonlinear Dynamic Model

In this work, we use the dynamic nonlinear model of SOFC proposed in [92]. Although the model is developed using various simplifying assumptions, such as ideal behavior of fuel and oxygen gases, choked orifices of the channels and constant SOFC temperature, it has shown to be suitable for slow dynamic simulation [153]. The schematic of the model is shown in Figure 7.1 and the numerical values for different parameters are given in Table 7.1.

The power supplied by the SOFC is calculated as

$$P_r = V_r I_r \quad (7.1)$$

where V_r and I_r are the output voltage and output current, respectively. It is required that the power supplied by SOFC is kept as close as possible to the power demand during the transient response and is equal to the power demand at steady state. The variables of SOFC can be classified as output variables, manipulated variables (MVs) and disturbance

Table 7.1: Parameters for the dynamic model of SOFC [92, 139]

Parameters	Description	Value
T_0	Absolute temperature	1273 K
F_0	Faraday's constant	96485 C/mol
R_0	Universal gas constant	8.314 J/(mol.K)
E_0	Ideal standard potential	1.18 V
N_0	Number of fuel cells in series	384
K_r	Constant, $N_0/4 F_0$	9.96×10^{-2}
K_{H_2}	Valve molar constant for H_2	8.32×10^{-6} mol/(s.Pa)
K_{H_2O}	Valve molar constant for H_2O	2.77×10^{-6} mol/(s.Pa)
K_{O_2}	Valve molar constant for O_2	2.49×10^{-5} mol/(s.Pa)
τ_{H_2}	Response time for H_2 flow	26.1 s
τ_{H_2O}	Response time for H_2O flow	78.3 s
τ_{O_2}	Response time for O_2 flow	2.91 s
r	Ohmic loss	0.126 Ω
T_e	Electrical response time	0.8 s
T_f	Fuel processor response time	5 s

variables. The output variables include

1. Fuel cell stack output voltage (V_r), which is calculated using the Nernst equation.

$$V_r = N_0 \left[E_0 + \frac{R_0 T_0}{2 F_0} \left(\ln \frac{p_{H_2} \sqrt{p_{O_2}}}{p_{H_2O}} \right) \right] - r I_r \quad (7.2)$$

V_r is directly related with the power supply and thus needs to be maintained at a constant setpoint to track the power demand. The allowable variation of V_r from its nominal value is ± 5 V.

2. Fuel utilization (U_f) is the ratio between the reacted and supplied hydrogen, *i.e.*

$$U_f = \frac{q_{H_2}^r}{q_{H_2}^{in}} \quad (7.3)$$

The fuel utilization has to be maintained between 80% – 90% [92]. Low fuel utilization may cause V_r to rise rapidly in addition to the inefficiency of the reaction,

Table 7.2: Operating conditions and constraint limits for SOFC [139]

Variable	Description	Constraints		
		Nominal	Lower	Upper
q_f^{in}	Fuel flowrate [mol/s]	0.7023	0	1.7023
$q_{O_2}^{in}$	Oxygen flowrate [mol/s]	0.6134	0	1.6134
RH_O	Input flowrate ratio	1.145	0	2
p_{dif}	Pressure difference [kPa]	0	-8	8
U_f	Fuel utilization	0.85	0.8	0.9
V_r	Fuel cell output voltage [V]	333.8	328.8	338.8
I_{in}	Current input [A]	300	270	330

where as higher fuel utilization may cause the cells to suffer from fuel starvation [21].

3. Ratio between hydrogen and oxygen flowrate (RH_O) is defined as,

$$RH_O = \frac{q_{H_2}^{in}}{q_{O_2}^{in}} \quad (7.4)$$

Although this variable seems unimportant, it needs to be maintained within allowable bounds, $0 \leq RH_O \leq 2$. The upper bound on RH_O is useful in keeping the pressure difference between anode and cathode below $4kPa$ under nominal conditions [153].

4. Pressure difference between the hydrogen and oxygen passing through the anode and cathode of the fuel cells (p_{dif}) also needs to be maintained within $\pm 8kPa$ to prevent damage to the electrolyte [153].

The MVs include fuel flowrate (q_f^{in}) and oxygen flowrate ($q_{O_2}^{in}$). These variables have physical constraints corresponding to the fully open and closed positions of the valves. The change in power demand is reflected through a change in current input (I_{in}), which is the main disturbance for SOFC. The current input is considered to be a measured disturbance.

In summary, the operational objectives of SOFC is to track the power demand while maintaining all output variables within their allowable bounds; see Table 7.2 for constraint

limits. This problem is challenging as V_r is the only measured output and all other variables, including unmeasured variables, have tight operating constraints.

7.3 Model Linearization

To facilitate the controller design, the non-linear model is linearized. The linear model can be written as

$$y(s) = G(s) u(s) + H(s) a(s) \quad (7.5)$$

where $G(s)$ and $H(s)$ denote the process and disturbance models and y , u and a are defined as

$$\begin{aligned} y &= \begin{bmatrix} \bar{R}H_{-O} & \bar{U}_f & \bar{p}_{dif} & \bar{V}_r \end{bmatrix}^T \\ u &= \begin{bmatrix} \bar{q}_{O_2}^{in} & \bar{q}_f^{in} \end{bmatrix}^T \\ a &= \bar{I}_{in} \end{aligned}$$

where the overhead bar denotes the deviation variable. Around the nominal operating point, $G(s)$ and $H(s)$ are obtained using Taylor Series expansion. The linearized process and disturbance models are given as

$$G(s) = \begin{bmatrix} -1.87 & \frac{1.63}{5s+1} \\ 0 & \frac{-1.21}{5s+1} \\ \frac{40.16}{2.91s+1} & \frac{-120.19}{(26.1s+1)(5s+1)} \\ \frac{33.47}{2.91s+1} & \frac{201.16}{(26.1s+1)(5s+1)} \end{bmatrix} \quad (7.6)$$

$$H(s) = \begin{bmatrix} 0 \\ \frac{0.0028}{0.8s+1} \\ \frac{0.2(-1.74s+1)}{(26.1s+1)(2.91s+1)(0.8s+1)} \\ \frac{-0.630(1189s^3+731.0s^2+81.86s+1)}{(78.3s+1)(26.1s+1)(2.91s+1)(0.8s+1)} \end{bmatrix} \quad (7.7)$$

The linear model is validated by implementing step changes with magnitudes being 10% of the nominal steady state values of manipulated and disturbance variables. The observed dynamic responses of the non-linear and linearized models are shown in Figure 7.2. Some differences in the dynamic and steady-state behavior are observed, particularly in V_r , which can be expected to be handled easily by a feedback controller. Overall, the model is deemed to be acceptable for controller design.

Figure 7.2: Comparison of dynamics between nonlinear (solid line) and linear (dashed line) models

7.4 Decentralized Controller Design

In this section, we systematically design a decentralized PID controller for SOFC. The design procedure includes (a) selection of controlled variables; (b) input-output pairing selection; and (c) tuning of decentralized PID controller. The methods used for carrying out these tasks and their application for SOFC is discussed next.

7.4.1 Selection of Controlled Variables

The selection of appropriate CVs is arguably the most important task in control system design. As shown in Table 7.2, the SOFC has tight operating constraints, which cannot be handled explicitly by simple PID controllers. We present a novel method for selection of CVs such that maintaining the selected CVs at constant setpoints facilitates constraint satisfaction for all the variables.

Prior to the selection of CVs z , the linearized model in (7.5) is scaled. Let the scaled model be

$$\hat{y}(s) = \hat{G}(s) \hat{u}(s) + \hat{H}(s) \hat{a}(s) \quad (7.8)$$

where $\hat{G}(s)$ and $\hat{H}(s)$ are the scaled process and disturbance transfer function matrices, respectively, defined as

$$\hat{G}(s) = D_y^{-1} G(s) D_u, \quad \hat{H}(s) = D_y^{-1} H(s) D_a \quad (7.9)$$

Here, D_y , D_u and D_d are output, input and disturbance scaling matrices, which are selected such that the allowable magnitudes of the scaled deviation variables, *i.e.* $\hat{y} = D_y^{-1}y$, $\hat{u} = D_u^{-1}u$ and $\hat{a} = D_a^{-1}a$, lie between ± 1 [124]. For SOFC, we have

$$D_y = \text{diag}(0.85, 0.05, 8, 5) \quad (7.10)$$

$$D_u = \text{diag}(0.6134, 0.7023) \quad (7.11)$$

$$D_a = 30 \quad (7.12)$$

For CV selection, let $\hat{y}$ and, $\hat{G}(s)$ and $\hat{H}(s)$ be conformably partitioned as:

$$\hat{y}_1(s) = \hat{G}_1(s) \hat{u}(s) + \hat{H}_1(s) \hat{a}(s) \quad (7.13)$$

$$\hat{y}_2(s) = \hat{G}_2(s) \hat{u}(s) + \hat{H}_2(s) \hat{a}(s) \quad (7.14)$$

We consider that $z = \hat{y}_1$ (selected CV) and the control system maintains $\hat{y}_1$ at 0 at all times (perfect control). We point out that perfect control is not achievable in practice. With an integral action controller, however, this assumption can be closely satisfied on a frequency-by-frequency basis until the bandwidth frequency [124]. With $\hat{y}_1(s)$ being perfectly controlled, for the partially closed-loop system, we have

$$\hat{u}(s) = P_u(s) \hat{a}(s), \quad \hat{y}_2(s) = P_y(s) \hat{a}(s) \quad (7.15)$$

where

$$P_u(s) = -\hat{G}_1^{-1}(s) \hat{H}_1(s) \quad (7.16)$$

$$P_y(s) = \hat{H}_2(s) - \hat{G}_2(s) \hat{G}_1^{-1}(s) \hat{H}_1(s) \quad (7.17)$$

As $\hat{y}_1$ is controlled, the constraints on $\hat{y}_1$ are trivially satisfied. The constraints on $\hat{y}_2$ and $\hat{u}$ are also satisfied, if

$$|P_{y,i}(j\omega)| < 1 \quad \text{and} \quad |P_{u,k}(j\omega)| < 1 \quad \forall \omega \in \mathbb{R} \quad (7.18)$$

where $P_{y,i}(j\omega)$ and $P_{u,k}(j\omega)$ are the i^{th} and k^{th} elements of $P_y(s)$ and $P_u(s)$ evaluated at frequency ω .

SOFC has 2 MVs and thus offset-free control for a maximum of 2 variables can be

achieved. As V_r is closely related to power demand, it is selected as a CV. The control of V_r only is sufficient if the constraints on all variables are satisfied by keeping V_r constant. The magnitudes of $P_{y,i}(j\omega)$ and $P_{u,k}(j\omega)$, when $q_{O_2}^{in}$ is used as MV for control of V_r are shown in Figure 7.3. It can be noted that $P_{y,i}(j\omega)$ for p_{dif} and U_f exceeds 1 at low frequencies. When V_r is controlled using q_f^{in} , $P_{y,i}(j\omega)$ for U_f exceeds one at high frequencies (details not shown). Thus, it can be concluded that constraint satisfaction cannot be achieved using V_r as the only CV and use of another CV is necessary. Two alternate strategies for selection of additional CV are presented next.

Figure 7.3: Magnitudes of $|P_u(j\omega)|$ and $|P_y(j\omega)|$ with control of V_r using $q_{O_2}^{in}$

7.4.1.1 Single Variable as CV

In addition to V_r , either RH_O , p_{dif} or U_f can be selected as a CV. Among these alternatives, selection of only U_f as additional CV leads to satisfaction of (7.18); see Figure 7.4. Thus, we have

$$z = \begin{bmatrix} V_r & U_f \end{bmatrix}^T \quad (7.19)$$

Although this analysis shows that all constraints are likely to be satisfied, when V_r and U_f are used as CVs, U_f cannot be readily measured online. This difficulty can be

overcome using the predicted value of U_f obtained using linear model for implementation of decentralized PID controller; see Section 7.4.3 for details.

Figure 7.4: Magnitudes of $|P_u(j\omega)|$ and $|P_y(j\omega)|$ with control of V_r and U_f

7.4.1.2 Measurement Combination as CV

Alternately, the CVs can be selected as a linear combination of measurements, which can be easily measured online. For SOFC, these measurements are given as

$$y_m = \begin{bmatrix} \hat{V}_r & \hat{q}_{O_2}^{in} & \hat{q}_f^{in} & \hat{I}_{in} \end{bmatrix}^T \quad (7.20)$$

We consider that

$$y_m(s) = G_m(s)\hat{u} + H_m(s)\hat{a} \quad (7.21)$$

where

$$G_m(s) = \begin{bmatrix} \hat{G}_{41}(s) & \hat{G}_{42}(s) \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad H_m = \begin{bmatrix} \hat{H}_4(s) \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (7.22)$$

The CVs z are selected as

$$z = M y_m \quad (7.23)$$

where $M \in \mathbb{R}^{2 \times 4}$ is the combination matrix. As discussed earlier, V_r should be chosen as a CV for SOFC. Thus, M is parameterized as

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ M_1 & M_2 & M_3 & M_4 \end{bmatrix} \quad (7.24)$$

With perfect control of z , we have

$$\hat{u}(s) = P_u(s)\hat{a}(s), \quad \hat{y}(s) = P_y(s)\hat{a}(s) \quad (7.25)$$

where

$$P_u(s) = -(MG_m(s))^{-1}MH_m(s) \quad (7.26)$$

$$P_y(s) = \hat{H}(s) - \hat{G}(s)(MG_m(s))^{-1}MH_m(s) \quad (7.27)$$

As before, all the constraints are satisfied, if (7.18) holds. Now, the optimal M can be found by solving

$$\min_M \max_{\omega \in \mathbb{R}} J = (\| [P_y(j\omega)^T P_u(j\omega)^T]^T \|_1 - 1) \quad (7.28)$$

$$s.t. \quad |P_{y,i}(j\omega)| < 1, |P_{u,k}(j\omega)| < 1 \quad (7.29)$$

where $\|\cdot\|$ is the matrix 1-norm and the objective function in (7.28) maximizes the distance between the constraint limits and the variables. An analytical solution of this optimization problem is difficult. By discretizing ω at 100 points taken in the range 0.001 – 10 Hz and using Matlab routine *fmincon*, we find that the minimal value of J is -0.1102 . For

self-optimizing control, it is noted in [3, 58, 61] that the use of combinations of a few measurements as CVs often provide similar performance as the case where combinations of all available measurements are used. For SOFC, when a linear combination of $\hat{q}_f^{in}$ and $\hat{q}_{O_2}^{in}$ is used, the the minimal value of J is -0.1092 obtained by setting $M_1 = M_4 = 0$ in (7.24), which is comparable to the minimal value of J seen with the use of combinations of all the available measurements. Thus, to simplify the control structure, the CVs are chosen as

$$z = \begin{bmatrix} V_r \\ -\hat{q}_{O_2}^{in} + 1.12 \hat{q}_f^{in} \end{bmatrix} = \begin{bmatrix} V_r \\ -\bar{q}_{O_2}^{in} + 0.98 \bar{q}_f^{in} \end{bmatrix} \quad (7.30)$$

where the equivalence follows based on the scaling on inputs in (7.11). The magnitudes of $P_{y,i}$ and $P_{u,k}$ with the control of these CVs is shown in Figure 7.5.

Figure 7.5: Magnitudes of $|P_u(j\omega)|$ and $|P_y(j\omega)|$ with control of V_r and a linear combination of MVs

7.4.2 Pairing Selection

For input-output pairing selection, we use RGA, which is defined as [11]:

$$\Lambda = G_z(0) \circ G_z^{-T}(0) \quad (7.31)$$

where $G_z(0)$ denotes the steady-state gain matrix between CVs and MVs, and $\circ$ is the element-by-element product. In this method, pairings are selected on RGA elements which are close to 1, while pairings on negative RGA elements are avoided to ensure integrity against loop failure [11, 63].

When U_f is selected as CV, $q_{O_2}^{in}$ has no effect on U_f ; see Figure 7.1 and (7.6). In this case, $G_z(0)$ is triangular and $\Lambda = I$, which implies that the preferred pairings are $\bar{V}_r - \bar{q}_{O_2}^{in}$ and $\bar{U}_f - \bar{q}_f^{in}$. When the measurement combinations are used as CV, we have

$$G_z(0) = \begin{bmatrix} 33.47 & 201.16 \\ -1 & 0.98 \end{bmatrix}, \quad \Lambda = \begin{bmatrix} 0.14 & 0.86 \\ 0.86 & 0.14 \end{bmatrix} \quad (7.32)$$

which recommends the use of off-diagonal pairings, *i.e.* $\bar{V}_r - \bar{q}_f^{in}$ and $(-\bar{q}_{O_2}^{in} + 0.98\bar{q}_f^{in}) - \bar{q}_{O_2}^{in}$.

7.4.3 PID Controller Tuning

We use the recently proposed SIMC rules for the tuning of PID controllers [121]. In this method, for

$$g(s) = \frac{K_p}{(\tau_1 s + 1)(\tau_2 s + 1)} e^{-\theta s}; \quad \tau_1 \geq \tau_2 \quad (7.33)$$

the tuning parameters are chosen as:

$$K_c = \frac{1}{K_p} \frac{\tau_1}{\tau_c + \tilde{\theta}}; \quad \tau_I = \min(\tau_1, 4(\tau_c + \tilde{\theta})); \quad \tau_D = \tau_2 \quad (7.34)$$

where K_c , τ_I , τ_D , and τ_c denote the controller gain, integral time, derivative time and desired closed-loop time constant, respectively, and $\tilde{\theta} = \theta + \Delta T/2$ with ΔT being the sampling time. The recommended value for τ_c is $\tilde{\theta}$ [121]. For SOFC, ΔT is taken as 1 s, which is the same as used for implementation of data-driven MPC in [139]. The appropriate value of τ_c is chosen based on closed-loop simulations using linear model in order to satisfy constraints.

7.4.3.1 Single Variable as CV

In this case, U_f is affected by only q_f^{in} and thus Loop 2 ($\bar{U}_f - \bar{q}_f^{in}$) is closed first. The controller parameters for the second loop are $K_{c,2} = -4.13$ and $\tau_{I,2} = 4$. The discrete form of PID controller is given as

$$K_2 = \frac{-4.13 + 3.10 q^{-1}}{1 - q^{-1}} \quad (7.35)$$

When Loop 2 is closed, the transfer function between $\bar{V}_r$ and $\bar{q}_{O_2}^{in}$ remains unchanged, as the model is triangular. The use of $\tau_c = \tilde{\theta}$ for Loop 1 result in constraint violation on RH_O , U_f and V_r . Although increasing τ_c allows satisfying the constraint on RH_O , the constraints on U_f and V_r are still violated due to the use of large sampling time, as discussed further in Section 7.5. Thus, to satisfy constraint on RH_O , we set $\tau_c = 3\tilde{\theta}$, which provides $K_{c,1} = 0.043$ and $\tau_{I,1} = 2.91$. The discrete form of this controller is

$$K_1 = \frac{0.044 - 0.029 q^{-1}}{1 - q^{-1}} \quad (7.36)$$

As V_r is the only available online measurement, the predicted value of U_f , *i.e.* U_f^P , is calculated based on (7.6) and fed back to the controller. The schematic diagram of the closed-loop system is shown in Figure 7.6, where the superscript s denotes the setpoint of the corresponding variables. This control strategy is referred to as linear prediction (LP) strategy in the subsequent discussion.

7.4.3.2 Measurement Combination as CV

We note that keeping the second CV, consisting of $\bar{q}_f^{in}$ and $\bar{q}_{O_2}^{in}$, constant at zero is equivalent to setting

$$\bar{q}_{O_2}^{in} = 0.98 \bar{q}_f^{in} \quad (7.37)$$

Figure 7.6: Schematic diagram of closed-loop system using linear prediction of U_f

Thus, we set $K_1 = 0.98$ to achieve perfect control of $q_{O_2}^{in}$. With K_1 in place, the model between $\bar{V}_r$ and $\bar{q}_f^{in}$ can be taken as $G_{41}(s) + 0.98G_{42}(s)$, where $G_{41}(s)$ and $G_{42}(s)$ are the open-loop models between $\bar{V}_r - \bar{q}_f^{in}$ and $\bar{V}_r - \bar{q}_{O_2}^{in}$, respectively. Subsequently, the following first order model is obtained for the design of PI controller by using 63.2% method

$$\bar{V}_r(s) = \frac{230.6}{27.5s + 1} \bar{q}_f^{in}(s) \quad (7.38)$$

The PI controller tuned using $\tau_c = \tilde{\theta}$ leads to constraint violation for several variables. As before, detuning the controller by setting $\tau_c = 4\tilde{\theta}$ allows constraint satisfaction for all variables except U_f and V_r . For this choice of τ_c , we have $K_{c,2} = 0.048$ and $\tau_{I,2} = 10$, for which the discretized PI controller is given as

$$K_2 = \frac{0.048 - 0.043q^{-1}}{1 - q^{-1}} \quad (7.39)$$

The schematic diagram of this control strategy is shown in Figure 7.7. This control strategy is referred to as measurement combination (MC) strategy in the subsequent discussion.

Figure 7.7: Schematic diagram of closed-loop system using measurement combination as CV

7.5 Closed-loop Simulations

In this section, we implement the decentralized PI control strategies on the dynamic model of SOFC by performing closed-loop simulations. The performance is evaluated using integral absolute error (IAE) for the outputs and total variation (TV) for the inputs.

At nominal conditions, the power demand is varied between $0.9 \times 10^5 W - 1.1 \times 10^5 W$. Due to power demand variation, multiple step changes occur in the disturbance I_{in} . The input and output signals obtained using closed-loop simulations are shown in Figure 7.8. It can be seen that both the proposed strategies track the power demand closely with no steady-state offset. The dynamic responses for V_r and P_r obtained using the two strategies are nearly identical. In comparison with the LP strategy, however, the MC strategy requires less aggressive changes in the inputs. Both the proposed strategies lead to constraint violation during the transient response, especially right after the changes in I_{in} occur. The LP strategy handles the constraint on U_f better, which is expected as U_f is used as a CV for the LP strategy. The issue of constraint violation is further discussed later in this section.

The same decentralized controllers are also implemented on SOFC being operated at different operating conditions for robustness evaluation. Similar to [139], the following operating conditions are considered:

1. The nominal power demand is $1.15 \times 10^5 W$ with $V_r = 328.4 V$ and $I_{in} = 350 A$.

Figure 7.8: Closed-loop responses for nominal conditions

The nominal values of $q_{O_2}^{in}$ and q_f^{in} are 0.7163 mol/s and 0.8202 mol/s , respectively.

The power demand is varied between $1.05 \times 10^5 \text{ W} - 1.15 \times 10^5 \text{ W}$.

2. The nominal power demand is $0.84 \times 10^5 \text{ W}$ with $V_r = 335.9 \text{ V}$ and $I_{in} = 250 \text{ A}$.

The nominal values of $q_{O_2}^{in}$ and q_f^{in} are 0.5117 mol/s and 0.5859 mol/s , respectively.

The power demand is varied between $0.84 \times 10^5 \text{ W} - 0.94 \times 10^5 \text{ W}$.

For both the robustness tests, V_r needs to be maintained within $\pm 5 \text{ V}$ of its nominal value, while constraints on other variables remain unchanged. The closed loop responses of the input and output signals for the 1st and 2nd robustness test are shown in Figure 7.9 and Figure 7.10, respectively. These responses show that despite the variation in operating conditions both the proposed strategies are able to track the power demand with no steady-state offset. Similar to the nominal conditions, the constraints on U_f and V_r are violated for a short duration after the large change in the disturbance occurs with the extent of constraint

Figure 7.9: Closed-loop responses for 1st robustness test Figure 7.10: Closed-loop responses for 2nd robustness test

Table 7.3: Integral Absolute Error (IAE) and Total Variation (TV) for SOFC

	Robustness Tests								
	Nominal Case			1 st test			2 nd test		
	PID		MPC	PID		MPC	PID		MPC
	LP	MC	[139]	LP	MC	[139]	LP	MC	[139]
Integral Absolute Error:									
RH_O	26.32	6.89	30.97	37.42	8.15	31.22	54.02	10.76	45.32
U_f	0.44	3.51	4.43	0.90	5.31	4.89	1.16	6.13	8.00
p_{dif}	725.31	205.33	784.70	971.06	422.32	865.60	945.17	396.63	795.70
V_r	51.64	75.38	303.70	73.68	102.53	359.30	55.16	86.01	364.90
P_r	71571	78457	NA	100405	110251	NA	92279	99856	NA
Total Variation:									
$q_{O_2}^{in}$	1.01	0.94	NA	1.40	1.39	NA	1.29	1.28	NA
q_f^{in}	1.73	0.96	NA	1.73	1.42	NA	1.73	1.30	NA

LP = Linear prediction; MC = Measurement combination; NA = Data not available

violation being larger for the 2nd robustness test.

The violation of constraints on V_r and U_f can lead to thermal stress and redox [7]. We point out that the use of data-driven MPC in [139] also results in violation on constraints on these variables. Furthermore, the decentralized controllers are designed using the same sampling time ($\Delta T = 1s$) as used for the data-driven MPC in [139]. The constraint handling ability of the decentralized control strategies can be significantly improved by using a smaller ΔT . For example, by re-tuning the PI controllers with $\Delta T = 0.1s$, constraint violation on V_r and U_f can be eliminated for the LP strategy and significantly reduced for the MC strategy for all the operating conditions.

For comparing the performances of the two decentralized control strategies further, the IAE and TV values are shown in Table 7.3. It can be noted that the IAE of power demand for the LP strategy is approximately 10% lower than the MC strategy. The IAE of RH_O and p_{dif} , and TV of q_f^{in} are, however, much lower for the MC strategy. Overall, the performances of both these strategies are deemed comparable.

The IAE values for both the proposed strategies are also compared with the

corresponding values reported in [139] for data-driven MPC. It can be noted that the decentralized control strategies provide lower IAEs than data-driven MPC. The IAE for P_r is not reported in [139]. However, as mentioned earlier, P_r is closely related with V_r . Thus, based on Table 7.3, the performance of the proposed decentralized control strategies for tracking varying power demand can be deemed to be significantly better than the data-driven MPC. Although this conclusion may seem surprising, a similar observation regarding superior performance of decentralized PID controller over MPC was earlier reported by Ricker [103]. An apparent reason for the poor performance of the MPC designed in [139] is the use of a non-square controller with more CVs than MVs, which distributes the error among different CVs including V_r . Furthermore, the performance requirements of tracking power demand and satisfaction of constraints on RH_2O and U_f are conflicting, as they require fast and slow manipulation of MVs, respectively. In such cases, selection of appropriate tuning parameters of MPC can be challenging [103].

7.6 Chapter Summary

In this work, decentralized control strategies are designed based on a benchmark model [92] for SOFC. The proposed control strategies meet the operational objectives of the SOFC closely. In terms of tracking the power demand, the performance of proposed strategies is shown to be superior than the data-driven model predictive controller (MPC) [139]. The performance advantages of the decentralized control strategies can be attributed to the systematic selection of controlled variables (CVs) to facilitate constraint satisfaction for all the key variables. This important issue has received limited attention for design of centralized controllers like MPC. In summary, this work demonstrates that a systematically designed decentralized controller can match or even outperform the performance of advanced control strategies, such as MPC, while maintaining the simplicity of the controller structure.

7.A Appendix: Linearization of SOFC Model

Let A_0 be steady state value of A , and $\bar{A}$ denote the deviation variable of A , *i.e.* $\bar{A} = A - A_0$. The steady state values of input and output variables are given in Table 7.2. The following linearized models are found based on the non-linear model in Figure 7.1.

1. Ratio between input hydrogen and oxygen (RH_O)

$$\begin{aligned}
 RH_O &= \frac{q_{H_2}^{in}}{q_{O_2}^{in}} \\
 &= -\frac{q_{f,0}^{in}}{(q_{O_2,0}^{in})^2} \bar{q}_{O_2}^{in} + \frac{1}{q_{O_2,0}^{in}} \frac{1}{1 + T_e s} \bar{q}_f^{in} \\
 &= -1.87 \bar{q}_{O_2}^{in} + \frac{1.63}{1 + 5s} \bar{q}_f^{in}
 \end{aligned}$$

2. Fuel utilization (U_f)

$$\begin{aligned}
 U_f &= \frac{q_{H_2}^r}{q_{H_2}^{in}} \\
 &= -\frac{q_{H_2,0}^r}{(q_{H_2,0}^{in})^2} \bar{q}_{H_2}^{in} + \frac{1}{q_{H_2,0}^{in}} \bar{I}^{in} \\
 &= -\frac{2K_r I_{in,0}}{(q_{f,0}^{in})^2} \bar{q}_f^{in} + \frac{2K_r}{q_{f,0}^{in}} \frac{1}{1 + T_e s} \bar{I}^{in} \\
 &= \frac{-1.21}{1 + 5s} \bar{q}_f^{in} + \frac{0.0028}{1 + 0.8s} \bar{I}^{in}
 \end{aligned}$$

3. Pressure difference between anode and cathode (p_{dif})

$$\begin{aligned}
 p_{dif} &= P_{O_2} - P_{H_2} \\
 &= \frac{1/K_{O_2}}{1 + \tau_{O_2} s} \left[\bar{q}_{O_2}^{in} - \frac{K_r}{1 + T_e s} \bar{I}^{in} \right] - \\
 &\quad \frac{1/K_{H_2}}{1 + \tau_{H_2} s} \left[\frac{1}{1 + T_f s} \bar{q}_f^{in} - \frac{2K_r}{1 + T_e s} \bar{I}^{in} \right] \\
 &= \frac{1/K_{O_2}}{1 + \tau_{O_2} s} \bar{q}_{O_2}^{in} - \frac{1/K_{H_2}}{1 + \tau_{H_2} s} \frac{1}{1 + T_f s} \bar{q}_f^{in} +
 \end{aligned}$$

$$\begin{aligned}
& \left[-\frac{1/K_{O_2}}{1 + \tau_{O_2}s} \frac{K_r}{1 + T_es} + \frac{1/K_{H_2}}{1 + \tau_{H_2}s} \frac{2K_r}{1 + T_es} \right] \bar{I}^{in} \\
= & \frac{40.16}{1 + 2.91s} \bar{q}_{O_2}^{in} - \frac{120.19}{(1 + 26.1s)(1 + 5s)} \bar{q}_f^{in} + \\
& \frac{0.2(1 - 1.74s)}{(1.26.1s)(1 + 2.91s)(1 + 0.8s)} \bar{I}^{in}
\end{aligned}$$

4. Output voltage (V_r)

$$\begin{aligned}
V_r &= NE + \frac{NRT}{2F} \ln \frac{P_{H_2} \sqrt{P_{O_2}}}{P_{H_2O}} - rI_r \\
&= \frac{NRT}{2F} \left[\frac{1}{2P_{O_2,0}} \frac{1/K_{O_2}}{1 + \tau_{O_2}s} \bar{q}_{O_2}^{in} + \frac{1}{P_{H_2,0}} \frac{1/K_{H_2}}{1 + \tau_{H_2}s} \frac{1}{1 + T_fs} \bar{q}_f^{in} \right] + \\
&\quad \frac{NRT}{2F} \left[-\frac{1}{P_{H_2,0}} \frac{2K_r}{1 + T_es} \frac{1/K_{H_2}}{1 + \tau_{H_2}s} - \frac{1}{2P_{O_2,0}} \frac{K_r}{1 + T_es} \frac{1/K_{O_2}}{1 + \tau_{O_2}s} - \right. \\
&\quad \left. \frac{1}{P_{H_2O,0}} \frac{2K_r}{1 + T_es} \frac{1/K_{H_2O}}{1 + \tau_{H_2O}s} \right] \frac{r}{1 + T_es} \bar{I}^{in} \\
&= \frac{33.39}{1 + 2.91s} \bar{q}_{O_2}^{in} + \frac{199.64}{(1 + 26.1s)(1 + 5s)} \bar{q}_f^{in} - \\
&\quad \frac{0.627(1195s^3 + 732.9s^2 + 81.83s + 1)}{(1 + 78.3s)(1 + 26.1s)(1 + 2.91s)(1 + 0.8s)} \bar{I}^{in}
\end{aligned}$$

Chapter 8

Conclusions and Future Work

8.1 Conclusions

In this thesis, several tools have been developed for design, analysis and assessment of structured controllers. The main contributions of this thesis are:

- Block diagram representation of the newly developed RTD-A controller [90] to gain a better understanding of the advantages and limitations of the algorithm.
- Semi-analytical tuning rule for RTD-A controller, which are simple and applicable to a wide range of processes including processes with inverse response and delay-dominant processes.
- Sums of squares (SOS) programming technique for finding a tight lower bound on the least achievable output variance for single-input single-output (SISO) processes under PID and multi-input multi-output (MIMO) processes under decentralized (PID) control.
- Practical minimum variance (MV) benchmark for performance assessment of decentralized controllers on a loop by loop basis, where the benchmark can be readily

estimated using closed-loop operational data.

- Genetic algorithm (GA) for efficient selection of nearly-optimal input-output pairings for decentralized control using relative gain array [11] and μ -interaction measure [26] as the selection criteria.
- Frequency response analysis based method for systematic selection of controlled variables for processes with tight operating constraints.
- Case study on solid oxide fuel cells to demonstrate that a systematically designed decentralized controller can meet the operational objectives of processes, which are difficult to control, and can even outperform advanced controllers such as model predictive control (MPC).

It is worth pointing out that some of the developed tools have wider applicability than considered in this thesis. For example, the SOS programming technique can be readily extended for characterizing fundamental limitations on achievable performance for fixed-order controllers. Similarly, GA can be used for selecting pairings based on almost all of the available selection criteria.

8.2 Directions for Future Work

The work presented in this thesis has highlighted a number of open issues, which include:

- The current RTD-A controller is only applicable to stable processes. The algorithm needs to be extended for integrating and unstable processes. In addition, extension of RTD-A controllers for multivariable processes is another potential future direction for research.
- Though useful, SOS programming is computationally costly and may encounter numerical problems. A better algorithm to find lower bound of a high-order

polynomial would allow finding lower bounds on achievable performance of structured controllers more efficiently and reliably.

- In practice, a process often has constraints on its inputs and outputs. The use of SOS programming technique to find lower bounds on achievable performance for such constrained processes operating with structured controllers is difficult and alternate techniques need to be developed.
- A lower bound estimation on the achievable output variance with simultaneous tuning of decentralized controllers need to be sought for development of a practical approach for simultaneous performance assessment of decentralized controllers.
- A performance assessment tool that account for the effect of tuning an individual loop on the other loops of the decentralized controller due to interactions and extensions of the results in Chapter 5 for block decentralized controllers should be pursued.
- GA is general technique for obtaining practical solutions for engineering problems. The potential of GA for solving other problems in control structure design, such as controlled variable selection based on self-optimizing control, should be explored.

Bibliography

- [1] P. Agrawal and S. Lakshminarayanan. Tuning proportional-integral-derivative controllers using achievable performance indices. *Ind. Eng. Chem. Res.*, 42(22):5576–5582, 2003.
- [2] I. A. Alatiqi and W. L. Luyben. Control of a complex sidestream column/stripper distillation configuration. *Ind. Eng. Chem. Process Des. Dev.*, 25(3):762–767, 1986.
- [3] V. Alstad, S. Skogestad, and E. S. Hori. Optimal measurement combinations as controlled variables. *J. Proc. Control*, 19(1):138–148, 2009.
- [4] J. Bao and P. L. Lee. *Process Control: The Passive Systems Approach*. Springer, London, England, 2007.
- [5] J. Bao, P. L. Lee, F. Wang, W. Zhou, and Y. Samyudia. A new approach to decentralised process control using passivity and sector stability conditions. *Chem. Eng. Comm.*, 182(1):213–217, 2000.
- [6] D. Bhattacharyya and R. Rengaswamy. A review of solid oxide fuel cell (SOFC) dynamic model. *Ind. Eng. Chem. Res.*, 48(13):6068–6086, 2009.
- [7] D. Bhattacharyya and R. Rengaswamy. System identification and nonlinear model predictive control of a solid oxide fuel cell. *Ind. Eng. Chem. Res.*, 49(10):4800–4808, 2010.

- [8] D. Bhattacharyya, R. Rengaswamy, and C. Y. Finnerty. Isothermal models for anode-supported tubular solid oxide fuel cells. *Chem. Eng. Sci.*, 62(16):4250–4267, 2007.
- [9] R. Bove and S. Ubertini. *Modeling Solid Oxide Fuel Cells: Methods, Procedures, and Techniques*. Springer-Netherlands, Dordrecht, Netherlands, 2008.
- [10] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time series analysis: Forecasting and control*. Holden-Day, 19702.
- [11] E. H. Bristol. On a new measure of interaction for multivariable process control. *IEEE Trans. Automat. Contr.*, 11(1):133–134, 1966.
- [12] C. Brosilow and B. Joseph. *Techniques of Model-Based controls*. Prentice Hall, New Jersey, 2002.
- [13] Y. Cao. Personal communication, 2010.
- [14] A. Chaisantikulwat, C. Diaz-Goano, and E. S. Meadows. Dynamic modelling and control of planar anode-supported solid oxide fuel cell. *Comput. Chem. Eng.*, 32:2365–2381, 2008.
- [15] D. Chen and D. E. Seborg. PI/PID controller design based on direct synthesis and disturbance rejection. *Ind. Eng. Chem. Res.*, 41(19):4807–4822, 2002.
- [16] H. Cui and E. W. Jacobsen. Performance limitations in decentralized control. *J. Process Contr.*, 12(4):485–494, 2002.
- [17] K. Deb. *Multi-objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Chichester, 2001.
- [18] L. Desborough and T. Harris. Performance assessment measures for univariate feedback control. *Can. J. Chem Eng.*, 70(6):1186–1186, 1993.

- [19] L. Desborough and R. Miller. Increasing customer value of industrial control performance monitoring - honeywell's experience. In *Proc. Chemical Process Control - VI*, Tuscon, AZ, 2001.
- [20] J. L. Garriga and M. Soroush. Model predictive control tuning methods: A review. *Ind. Eng. Chem. Res.*, 49(8):3505–3515, 2010.
- [21] R. Gaynor, F. Mueller, F. Jabbari, and J. Brouwer. On control concepts to prevent fuel starvation in solid oxide fuel cells. *J. Power Sources*, 180:330–342, 2008.
- [22] L. Giovanini. Predictive feedback control: An alternative to proportional–integral–derivative control. *Proc. Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 223(7):901–917, 2009.
- [23] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine learning*. Pearson Education, Singapore, 1989.
- [24] G. C. Goodwin and K. Sin. *Adaptive Filtering Prediction and Control*. Prentice-Hall, Englewood Cliffs, 1984.
- [25] M. J. Grimble. Non-linear generalized minimum variance feedback, feedforward and tracking control. *Automatica*, 41(6):957–969, 2005.
- [26] P. Grosdidier and M. Morari. Interaction measures for sytems under decentralized controf. *Automatica*, 22(3):309–319, 1986.
- [27] P. Grosdidier and M. Morari. The μ interaction measures. *Ind. Eng. Chem. Res.*, 26(6):1193–1202, 1987.
- [28] P. Grosdidier, M. Morari, and B. R. Holt. Closed-loop properties from steady-state gain information. *Ind. Eng. Chem. Fundam.*, 24:221–235, 1985.

- [29] S. A. Hajimolana and M. Soroush. Dynamics and control of a tubular solid-oxide fuel cells. *Ind. Eng. Chem. Res.*, 48:6112–6125, 2009.
- [30] D. J. Halt and R. G. Colclaser. Transient modeling and simulation of tubular solid oxide fuel cells. *IEEE T, Energy Conver.*, 14:749–753, 1999.
- [31] I. J. Halvorsen, S. Skogestad, J. C. Morud, and V. Alstad. Optimal selection of controlled variables. *Ind. Eng. Chem. Res.*, 42:3273–3284, 2003.
- [32] J. Hanna, S.R. Upreti, A. Lohi, and F. Ein-Mozaffari. Constrained minimum variance control using hybrid genetic algorithm An industrial experience,. *J. Process Contr.*, 18(1):36–44, 2008.
- [33] T. J. Harris. Assessment of control loop performance. *Can. J. Chem. Eng.*, 67(5):856–861, 1989.
- [34] T. J. Harris, F. Boudreau, and J. F. Macgregor. Performance assessment of multivariable feedback controllers. *Automatica*, 32(11):1505–1518, 1996.
- [35] T. J. Harris, C. T. Seppala, and L. D. Desborough. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *J. Proc. Control*, 9(7):1–17, 1999.
- [36] T. J. Harris and W. Yu. Controller assessment for a class of non-linear systems. *J. Process Contr.*, 17(7):607–619, 2007.
- [37] C. A. Harrison and S. J. Qin. Minimum variance performance map for constrained model predictive control. *J. Proc. Control*, 19:1199–1204, 2009.
- [38] M. J. He and W. J. Cai. New criterion for control-loop configuration of multivariable processes. *Ind. Eng. Chem. Res.*, 43(22):7057–7064, 2004.

- [39] M. J. He, W. J. Cai, W. Ni, and L. H. Xie. RNGA based control system configuration for multivariable processes. *J. Process Contr.*, 6(19):1036–1042, 2009.
- [40] M. J. He, W. J. Cai, and B. F. Wu. Control structure selection based on relative interaction decomposition. *Int. J. of Control*, 79(10):1285–1296, 2006.
- [41] M. J. He, W. J. Cai, and B. F. Wu. Design of decentralized IMC-PID controller based on dRI analysis. *AIChE J.*, 52(11):3852–3863, 2006.
- [42] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, England, 1985.
- [43] B. Huang. Minimum variance control and performance assessment of time-variant processes. *J. Process Contr.*, 12(6):707–719, 2002.
- [44] B. Huang. A pragmatic approach towards assessment of control loop performance. *Int. J. Adapt. Control Signal Process.*, 17(7–9):589–608, 2003.
- [45] B. Huang and S. L. Shah. *Performance Assessment of Control Loops: Theory and Applications*. Springer-Verlag, London, England, 1999.
- [46] B. Huang, S. L. Shah, and E. K. Kwok. Good, bad or optimal? Performance assessment of multivariate processes. *Automatica*, 33(6):1175–1183, 1997.
- [47] B. Huang, S. L. Shah, and R. Miller. Feedforward plus feedback controller performance assessment of MIMO systems. *IEEE Transactions on Control Systems Technology*, 8(3):580–587, 2000.
- [48] H. P. Huang, C. L. Chen, Y. C. Chao, and P. L. Chen. A modified smith predictor with an approximate inverse of dead time. *AIChE J.*, 36(7):1025–1031, 1990.
- [49] H. P. Huang, M. Ohshima, and I. Hashimoto. Dynamic interaction and multiloop control system design. *J. Process Contr.*, 4(1):15–27, 1994.

- [50] H. Huo, X. Zhu, W. Hu, H. Tu, J. Li, and Yang J. Nonlinear model predictive control of SOFC based on a Hammerstein model. *J. Power Sources*, 185:338–344, 2008.
- [51] M. Jain and S. Lakshminarayanan. Estimating performance enhancement with alternate control strategies for multiloop control system. *Chem. Eng. Sci.*, 62(17):4644–4658, 2007.
- [52] M. Jelali. An overview of control performance assessment technology and industrial applications. *Control Eng. Pract.*, 14(5):441–466, 2006.
- [53] F. Jurado. Modeling SOFC plants on the distribution system using identification algorithm. *J. Power Sources*, 129(2):205–215, 2006.
- [54] F. Jurado. Predictive control of solid oxide fuel cells using fuzzy Hammerstein models. *J. Power Sources*, 158(1):245–253, 2006.
- [55] F. Jurado and M. Ortega. Model based predictive control of fuel cells. *Electr. Pow. Compo. Sys.*, 34(5):587–602, 2006.
- [56] E. I. Jury. *Theory and Application of the z-transform Method*. John Wiley & Sons, London, 1964.
- [57] V. Kariwala. Fundamental limitation on achievable decentralized performance. *Automatica*, 43(10):1849–1854, 2007.
- [58] V. Kariwala. Optimal measurement combination for local self-optimizing control. *Ind. Eng. Chem. Res.*, 46(11):3629–3634, 2007.
- [59] V. Kariwala and Y. Cao. Efficient branch and bound methods for pairing selection. In *Proceedings of IFAC World Congress*, Seoul, Republic of Korea, 2008.
- [60] V. Kariwala and Y. Cao. Branch and bound method for multiobjective pairing selection. *Automatica*, 46(5):932–936, 2010.

- [61] V. Kariwala, Y. Cao, and S. Janardhanan. Local self-optimizing control with average loss minimization. *Ind. Eng. Chem. Res.*, 47(4), 2008.
- [62] V. Kariwala, J. F. Forbes, and E. S. Meadows. Minimum variance benchmark for decentralized controllers. In *Proc. American Control Conference*, Portland, OR, USA, 2005.
- [63] V. Kariwala and M. Hovd. Relative gain array: Common misconceptions and clarifications. In *Proceedings of 7th Symposium on Chemical Process Control*, Lake Louise, Canada, 2006.
- [64] V. Knyazkin, L. Soder, and C. Canizares. Control challenges of fuel cell driven distributed generation. Available at [http://geri.uwaterloo.ca/pdf/modelling\\$_{dg}.pdf](http://geri.uwaterloo.ca/pdf/modelling$_{dg}.pdf).
- [65] B. S. Ko and T. F. Edgar. Assessment of achievable PI control performance for linear processes with dead time. In *Proc. American Control Conference*, Philadelphia, United States, 1998.
- [66] B. S. Ko and T. F. Edgar. Multiloop PID control performance monitoring. In *Proc. The 8th Congress of Asian Pacific Confederation of Chemical Engineering*, Seoul, Korea, August 1998.
- [67] B. S. Ko and T. F. Edgar. Performance assessment of cascade control loops. *AIChE J.*, 46(2):281–291, 2000.
- [68] B. S. Ko and T. F. Edgar. PID control performance assessment: The single-loop case. *AIChE J.*, 50(6):1211–1218, 2004.
- [69] H. Koivo and J. Tanntu. Tuning of PID controllers: Survey of SISO and MIMO techniques. In *Proc. IFAC Intelligent Tuning and Adaptive Control Symposium*, Singapore, 1991.

- [70] I. K. Kookos and J. D. Perkins. Heuristic-based mathematical programming framework for control structure selection. *Ind. Eng. Chem. Res.*, 40(9):2079–2088, 2001.
- [71] S. Lakshminarayanan, M. W. Hermanto, D. B. Goradia, and G. P. Rangaiah. Estimation of variance reduction opportunities through cascade control. *Can J. Chem. Eng.*, 84(6):704–714, 2006.
- [72] E. L Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. *The Traveling Salesman Problem: A Case Study in Local Optimization*, pages 215–310. John Wiley & Sons, London, England, 1985.
- [73] Y. Lee, S. Park, M. Lee, and C. Brosilow. PID controller tuning for desired closed-loop responses for SI/SO systems. *AIChE J.*, 44(1):106–115, 1998.
- [74] Y. H. Li and S. S. Choi. An analysis of the control and operation of a solid oxide fuel-cell power plant in an isolated system. *IEEE T. Energy Conver.*, 20(2):381–387, 2005.
- [75] M. G. Lin, S. Lakshminarayanan, and G. P. Rangaiah. A comparative study of recent/popular PID tuning rules for stable, first-order plus dead time, single-input single-output processes. *Ind. Eng. Chem. Res.*, 47(2), 2008.
- [76] L. Ljung. *System Identification: Theory for the user*. Prentice Hall, Upple Saddle River, 1999.
- [77] J. Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proc. of the CACSD Conference*, Taipei, Taiwan, 2006.
- [78] A. L. Maalouf. The effect of the time-constant to the reference trajectory in predictive control on closed-loop stability. In *Proceedings of American control conference*, Minneapolis, United States, 2006.

- [79] J. M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, Harlow, England, 2001.
- [80] J. Mikles and M. Fikar. *Process Modelling, Identification, and Control*. Springer, New York, 2007.
- [81] M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, 1998.
- [82] C. F. Moore, C. L. Smith, and P. W. Murrill. Improved algorithm for direct digital control. *Instrum. Contr. Syst.*, 43(1):70–74, 1970.
- [83] M. Morari and C. E. Garcia. Internal model control 1. A unifying review and some new results. *Ind. Eng. Chem. Process Des. Dev.*, 21(2):308–323, 1982.
- [84] M. Morari and J. H. Lee. Model predictive control: past, present and future. *Comput. Chem. Eng.*, 23(4–5):667–682, 1999.
- [85] Z. F. Ng, S. S. How, A. Y. Sendjaja, and V. Kariwala. Analysis and tuning of RTD-A controllers. In *Proceedings of PSE Asia Conference*, Singapore, 2010.
- [86] J. Nie, J. Demmel, and B. Sturmfels. Minimizing polynomials via sum of squares over the gradient ideal. *Mathematical programming*, 106(3):587–606, 2006.
- [87] J. E. Normey-Rico and E. F. Camacho. *Control of Dead-time Processes*. Springer, London, 2007.
- [88] B. A. Ogunnaike, J. P. Lemaire, Morari M., and W. H. Ray. Advanced multivariable control of a pilot-plant distillation column. *AIChE J.*, 29(4):632–640, 1983.
- [89] B. A. Ogunnaike and K. Mukati. An alternative structure for next generation regulatory controllers, Part I: Basic theory for design, development, and implementation. *J. Process Contr.*, 16(5):499–509, 2006.

- [90] B. A. Ogunnaike, K. Mukati, and M. Rasch. An alternative structure for next generation regulatory controllers, Part II: Stability analysis, tuning rules, and experimental validation. *J. Process Contr.*, 19(2):272–287, 2009.
- [91] B. A. Ogunnaike and W. H. Ray. *Process Dynamics, Modeling, and Control*. Oxford University Press, New York, 1994.
- [92] J. Padullés, G. W. Ault, and J. R. McDonald. An integrated SOFC plant dynamic model for power system simulation. *J. Power Sources*, 86(1-2):495–500, 2000.
- [93] G. Pannocchia, N. Laachi, and J. B. Rawlings. A candidate to replace PID control: SISO-constrained LQ control. *AIChE J.*, 51(4):1178–1189, 2005.
- [94] P. A. Parrilo. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, Pasadena, United States, 2000.
- [95] P. Parrilo and B. Sturmfels. *Algorithmic and Quantitative Real Algebraic Geometry*, pages 83–99. American Mathematical Society, Providence, United States, 2003.
- [96] S. C. Patwardhan. Workshop on model predictive control, 2007. Institute of Chemical and Engineering Sciences, Singapore.
- [97] Y. Qi, B. Huang, and K. Chuang. Dynamic modeling of solid oxide fuel cell: The effect of diffusion and inherent impedance. *J. Power Sources*, 180(1-2):32–47, 2005.
- [98] Y. Qi, B. Huang, and J. Luo. 1-D dynamic modeling of SOFC with analytical solution for reacting gas flow problem. *AIChE J.*, 54:1537–1553, 2008.
- [99] S. J. Qin. Controller performance monitoring - A review and assessment. *Comput. Chem. Eng.*, 23:173–186, 1998.

- [100] S. J. Qin. A survey of industrial model predictive control technology. *Contr. Eng. Pract.*, 11(7):733–764, 2003.
- [101] K. Rambabu. *Control Linear Modeling and Nonlinear State Estimation Applied to SOFC-GT Power Systems*. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2007.
- [102] K. Rambabu, L. Imsland, B. A. Foss, C. Stiller, B. Thorud, and O. Bolland. Modeling and control of a SOFC-GT-based autonomous power system. *Energy*, 32(4):406–417, 2007.
- [103] N. L. Ricker. Decentralized control of the Tennessee Eastman challenge process. *J. Process Contr*, 6(4):205–222, 1996.
- [104] D. E. Rivera, M. Morari, and S. Skogestad. Internal model control. 4. PID controller design. *Ind. Eng. Chem. Process Des. Dev.*, 1(25):252–265, 1986.
- [105] M. Rotkowitz and S. Lall. A characterization of convex problems in decentralized control. *IEEE T. Automat. Contr.*, 51(2):274–286, 2006.
- [106] Y. Samyudia, P. L. Lee, I. T. Cameron, and Green M. A new approach to decentralised control design. *Chem. Eng. Sci.*, 50(11):1695–1706, 1995.
- [107] D. E. Seborg and T. F. Edgar. *Process Dynamics and Control*. John Wiley & Sons, Mew York, 1989.
- [108] K. Sedghisigarchi and A. Feliachi. Dynamic and transient analysis of power distribution systems with fuel cells – Part I: Fuel-cell dynamic model. *IEEE T. Energy Convers.*, 19(2):423–428, 2004.
- [109] A. Y. Sendjaja and V. Kariwala. On achievable PID performance. In *Proceedings of Regional Symposium of Chemical Engineering*, Yogyakarta, Indonesia, 2007.

- [110] A. Y. Sendjaja and V. Kariwala. Achievable PID performance using sums of squares programming. *J. Proc. Contr.*, 19(6):1061–1065, 2009.
- [111] A. Y. Sendjaja and V. Kariwala. Performance assessment of decentralized controllers. In *Proceedings of 9th International Symposium on Advanced control of Chemical Processes*, Istanbul, Turkey, 2009.
- [112] A. Y. Sendjaja and V. Kariwala. A practical approach for performance assessment of decentralized controllers. In *Proceedings of 2nd CUTSE International Conference*, Miri, Malaysia, 2009.
- [113] A. Y. Sendjaja, W. M. K. Lesley, and V. Kariwala. Decentralized control of solid oxide fuel cells. In *Proceedings of Regional Symposium of Chemical Engineering*, Kuala Lumpur, Malaysia, 2008.
- [114] A. Y. Sendjaja, Z. F. Ng, S. S. How, and V. Kariwala. Analysis and tuning of RTD-A controllers. *Ind. Eng. Chem. Res.*, 50(6):3415–3425, 2011.
- [115] M. M. Seron, J. H. Braslavsky, and G. C. Goodwin. *Fundamental Limitations in Filtering and Control*. Springer Verlag, London, 1997.
- [116] M. Shamsuzzoha, S. Skogestad, and I. J. Halvorsen. On-line PI controller tuning using closed-loop setpoint response for stable and integrating processes. In *Proceedings of 9th International Symposium on DYCOPS*, Leuven, Belgium, 2010.
- [117] R. Shridhar and D. J. Cooper. A tuning strategy for unconstrained SISO model predictive control. *Ind. Eng. Chem. Res.*, 36(3):729–746, 1997.
- [118] D. D. Siljak. *Decentralized control of complex systems*. Academic Press, 1991.
- [119] S.C. Singhal and K. Kendall. *High-temperature Solid Oxide Fuel Cells: Fundamentals, Design and Applications*. Elsevier, Oxford, England, 2003.

-
- [120] S. Skogestad. Plantwide control: The search for the self-optimizing control structure. *J. Proc. Control*, 10(5):487–507, 2000.
- [121] S. Skogestad. Simple analytic rules for model reduction and PID controller tuning. *J. Proces Contr.*, 11(6):291–309, 2003.
- [122] S. Skogestad and M Morari. Implications of large RGA elements on control performance. *Ind. Eng. Chem. Res.*, 26(11):2323–2330, 1987.
- [123] S. Skogestad and M. Morari. Variable selection for decentralized control. *Model. Ident. Control*, 13(2):113–125, 1992.
- [124] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. John Wiley & Sons, Chichester, England, 2nd edition, 2005.
- [125] O. J. M. Smith. Closer control of loops with dead time. *Chem. Eng. Prog.*, 53(5):217–219, 1957.
- [126] M. Sorrentino, C. Pianese, and Y. Guezennet. A hierarchical modeling approach to the simulation and control of planar solid oxide fuel cells. *J. Power Sources*, 180(1):380–392, 2008.
- [127] D. D. Surlas and V. Manousiouthakis. Best achievable decentralized performance. *IEEE T. Automat. Contr.*, 40(11):1858–1871, 1995.
- [128] Y. Sreenivas, T. W. Yeng, G. P. Rangaiah, and S. Lakshminarayanan. A comprehensive evaluation of PID, cascade, model-predictive, and RTD-A controllers for regulation of hypnosis. *Ind. Eng. Chem. Res.*, 48(12):5719–5730, 2009.
- [129] N. Stanfelj, T. E. Marlin, and J. F. MacGregor. Monitoring and diagnosing process

- control performance: the single loop case. *Ind. Eng. Chem. Res.*, 32(2):301–314, 1993.
- [130] J. F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optim. Method Softw.*, 11(1):625–653, 1999.
- [131] K. R. Sundaresan and P. R. Krishnaswamy. Estimation of time delay time constant parameters in time, frequency, and laplace domain. *Can. J. Chem. Eng.*, 56(2):257–262, 1978.
- [132] P. Tatjewski. *Advanced Control of Industrial Processes: Structures and algorithms*. Springer, London, 2006.
- [133] K. J. Åström. *Introduction to Stochastic Control Theory*. Academic Press, London, England, 1970.
- [134] K. J. Åström and T. Hägglund. *PID controllers: theory, design, and tuning*. Instrumentation Society of America, Research Triangle Park, United States, 1995.
- [135] M. Van de wal. *Selection of inputs and outputs for control*. PhD thesis, Eindhoven University of Technology, 1998.
- [136] M. Van de Wal and B. De Jager. Control structure design: A survey. In *Proceedings of American Control Conference*, Seattle, United States, 1995.
- [137] M. Van de Wal and B. de Jager. A review of methods for input/output selection. *Automatica*, 37(4):487–510, 2001.
- [138] D. Vrancic, S. Strmcnik, J. Kocijan, and P. B. de Moura Oliveira. Improving disturbance rejection of PID controllers by means of the magnitude optimum method. *ISA T.*, 49(1):47–56, 2010.

- [139] X. Wang, B. Huang, and T. Chen. Data-driven predictive control for solid oxide fuel cells. *J. Process Contr.*, 17(2):103–114, 2006.
- [140] Y. G. Wang and H. H. Shao. Optimal tuning for pi controller. *Automatica*, 36(1):147–152, 2000.
- [141] M. C. Wellons and T. F. Edgar. The generalized analytical predictor. *Ind. Eng. Chem. Res.*, 26(8):1523–1536, 1987.
- [142] S. K. P. Wong and D. E. Seborg. A theoretical analysis of Smith and analytical predictors. *AIChE J.*, 32(10):1597–1605, 1986.
- [143] R. K. Wood and M. W. Berry. Terminal composition control of a binary distillation column. *Chem. Eng. Sci.*, 28(9):1707–1717, 1973.
- [144] X. Wu, X. Zhu, G. Cao, and H. Tu. Predictive control of SOFC based on a GA-RBF neural network model. *J. Power Sources*, 179:232–239, 2008.
- [145] Q. Xiong and W. J. Cai. Effective transfer function method for decentralized control system design of multi-input multi-output processes. *J. Process Contr.*, 16(8):773–784, 2006.
- [146] Q. Xiong, W. J. Cai, and M. J. He. A practical loop pairing criterion for multivariable processes. *J. Process Contr.*, 15(7):741–747, 2005.
- [147] Q. Xiong, W. J. Cai, M. J. He, and M. He. Decentralized control system design for multivariable processes - A novel method based on effective relative gain array. *Ind. Eng. Chem. Res.*, 45(8):2769–2776, 2006.
- [148] F. Xu and B. Huang. Performance monitoring of SISO control loops subject to LTV disturbance dynamics: An improved LTI benchmark. *J. Process Contr.*, 16(6):567–579, 2006.

- [149] S. Xu and J. Bao. Interaction analysis for decentralized control based on dissipativity. *Asia Pac. J. Chem. Eng.*, 3(6):656–666, 2008.
- [150] O. Yamamoto. Solid oxide fuel cells: Fundamental aspects and prospects. *Electrochim. Acta*, 45:2423–2435, 2000.
- [151] S. Yelneedi, S. Lakshminarayanan, and G. P. Rangaiah. A comparative study of three advanced controllers for the regulation of hypnosis. *J. Process Contr.*, 19:1458–1469, 2009.
- [152] J. I. Yuz and G. C. Goodwin. Loop performance assessment for decentralized control of stable linear systems. *Eur. J. Control*, 9(1):116–130, 2003.
- [153] Y. Zhu and K. Tomsovic. Development of models for analyzing the load-following performance of microturbines and fuel cells. *Electric Power Syst. Res.*, 62(1):1–11, 2002.
- [154] Z. X. Zhu. Variable pairing selection based on individual and overall interaction measures. *Ind. Eng. Chem. Res.*, 35(11):4091–4099, 1996.
- [155] J. G. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *Trans ASME*, 64:759–768, 1942.