University of Alberta

Identification and control of fractional and integer order systems

by

Anuj Narang

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

Doctor of Philosophy

in

Process Control

Department of Chemical and Materials Engineering

Spring 2012
Edmonton, Alberta

Abstract

The main focus of this thesis is on developing parsimonious models using measured process data and subsequent use of these models in the design of controllers for chemical engineering processes, in particular, for processes with large dead times. The application case studies presented and discussed in this thesis include diverse examples such as a classical heat transfer wall problem, a continuous stirred tank heater with transportation delays, an industrial scaled primary separation cell, and froth heaters.

Two different types of processes are discussed in this thesis: 1) processes that can be described by fractional order transfer function models and 2) industrial processes that are modeled using conventional rational (integer) order models, as is the normal practice in industry. For fractional order systems, this thesis proposes a nested loop optimization method where model parameters including time delay are estimated iteratively in the inner loop and the fractional order model is estimated in the non-linear outer loop. The proposed method is applied in simulation on distributed parameter systems such as a classical heat transfer wall problem and on identification data obtained from laboratory experiments of a continuous stirred tank heater (CSTH) with transportation delays and industrial froth heater process. A fractional order PI controller tuning method using Bode's ideal transfer function as the reference system is also developed for fractional and integer order systems. The proposed tuning method is evaluated by simulation on fractional and integer order systems and experimental application on a computer-interfaced pilot scale

CSTH process.

Application examples, related to conventional (integer order) models, discussed in the thesis involve two industrial case studies in the oil sands industry. The first of these is the regulation of the froth bitumen and middlings Interface level in a separation cell process which is part of the oil-sands extraction unit. Internal model control (IMC) and model predictive control (MPC) using linear models are designed, implemented and tested in real time on the industrial separation cell. These controllers yielded better performance over the existing control strategy which uses PID control. The second application is concerned with temperature control of the bitumen froth which is part of the froth treatment unit. Using the linear models obtained from the industrial data, a gain scheduling multivariable MPC is designed, and tested in simulations and compared with the current operation which uses a number of local PID controllers. Results presented in the thesis illustrate the first successful industrial implementation of an MPC controller on a separation cell in the oil sands extractions unit at Suncor Energy Inc. in Fort McMurray, Alberta.

Overall, this thesis presents results on identification and model based control design case studies on fractional order systems, distributed parameter systems and two industrial oil sands processes.

Acknowledgements

I express my most sincere thanks and appreciation to my supervisors, Dr. Sirish. L. Shah and Dr. Tongwen Chen, who enthusiastically encouraged and guided me throughout all the stages of my thesis work. It really means a lot to work with people who are so understanding and supportive. Dr. Shah and Dr. Chen supervised my research for more than four years and with them I learned a lot about everything and it has been a wonderful experience working with them. Their interest in my work, careful reading and corrections, helpful suggestions, helped me immensely throughout the course of my research work. I really enjoyed the independence they gave me to pursue new ideas, learn and teach new subjects. I also want to acknowledge the financial support through the NSERC Strategic Grant and NSERC-Matrikon-Suncor-Icore Industrial research chair program in computer process control that Dr. Chen and Dr. Shah have provided me throughout my graduate studies. I would also like to thank them for providing me the opportunities to present my research at technical conferences and giving me more exposure to the current research fields.

Another person I would like to thank is Artin Afacan. It has been a wonderful experience working with him as a teaching assistant for process data analysis course and writing educational papers. I learned a lot from him about the practical aspects of teaching and enjoyed all the great talks. He always motivated me to do good work.

I would also like to thank Dr. J. Prakash, who visited University of Alberta several times over the last couple of years during the summer term. My numerous discussions with him and his invaluable suggestions always motivated me in my research.

I would also like to thank Dr. Ramesh Kadali at Suncor Energy Inc., for introducing the two industrial case studies to me and for all the useful discussions about the process and how to approach the problem. I am also grateful to Eliyya, Elom and Jamie at Suncor Energy for being so helpful in running the industrial experiments and sharing their expertise.

I would also like to thank Dr. Sachin Patwardhan, he has been and will always be a motivation for me on how to reach top level of excellence in both teaching and research. He is one of the main reason on why I decided to pursue PhD after my masters degree.

I met some really interesting people during the course of my studies and made some close friends. I would like to thank all the CPC group members and Dr. Dave Shook for some great interactions and critical reviews during the group presentations. I sincerely thank all my friends: Chirag Gupta, Aditya Tulsyan, Xing Jin, Nima Danesh, Samidh Pareek, Sandeep Reddy, Arjun V. Shenoy, Yashasvi Purwar, Saneej B.C., Venkat Nadadoor, Fari Kiasi, Iman Izadi, Fan Yang, Yuri Shardt, Yue Yang, Ulagendran Venukrishnan, Siddhartha Kumar, Manjeet Singh, Debanjan Chakrabarti, Abhijit Badwe, Vinay Bavdekar, Swanand Khare, Hector Siegler, Kartik Surisetty, Mohammad Iqbal, Sankar Mahadevan, Phanindra Varma, Naseeb A. Adnan, Salman, Amal Mehrotra, Rahima Bhanji, Prakash Bhalavi, Lalit Kumar, Palash Panja for their valuable suggestions, constant help and support. I wish you all the very best for future endeavors.

Last but not the least, I would like to thank my parents, my sister and my jiju for being so understanding and for their constant support and encouragement throughout my studies.

Contents

List of Tables

List of Figures

List of Symbols

D^{α}	Fractional differentiation or integration of order ι
$\boldsymbol{\Gamma}(.)$	Euler's Gamma function
$\pounds(.)$	Laplace transform operator
$\pounds^{-1}(.)$	Inverse Laplace transform operator
γ or α	Commensurate fractional order
$G(s)$	Transfer function
L	Time delay or dead time
s^{λ}	Fractional order (with λ) operator in Laplace domain
ω	frequency (per sec)
F(s^{λ})	Filter transfer function
y_t	Sampled output(s) at tth sampling instant
u_t	Sampled input(s) at tth sampling instant
$\mathbf{a}_n$ and $\mathbf{b}_m$	Constant model parameters of fractional order transfer function.
$\hat{y}_t$	Predicted value of the model output(s) at tth sampling instant
$\Phi(t)$	Regressor matrix
$\Phi_{IV}(t)$	Instrumental variable matrix
$\hat{\theta}^{(i)}$	Parameter estimates at ith iteration count
$G_{FO}(s)$	Fractional order transfer function

$G_{IO}(s)$	Integer order transfer function
$\bar{\hat{\theta}}$	Sample mean of parameter vector, $\hat{\theta}$
$s(\hat{\theta})$	Sample standard deviation of parameter vector, $\hat{\theta}$
ϱ	Thermal conductivity of the wall
ζ	Thermal diffusivity of the wall
T	Temperature of the wall
S_A	Cross-sectional area of the wall
Ω	Heat flux applied to the wall
$C_{FO}(s)$	Fractional order controller transfer function
K_c	Proportional gain for $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ controller
K_i	Integral gain for $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ controller
K_d	Derivative gain for $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ controller
λ	Integrator order for $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ controller
μ	Differentiation order for $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ controller
$G_{ref}(s)$	Open-loop Bode's ideal transfer function
$G_{refCL}(s)$	Closed-loop transfer function of the reference model
w_y and w_u	Weights or penalties for the terms in cost function for FO-PI tuning method
$S(s)$	Maximum sensitivity function
$T(s)$	Complementary sensitivity function
mc	Momentum factor in the PSO algorithm
w	Inertia factor in the PSO algorithm
D	Diffusion coefficient
c	Concentration of the tracer

$\boldsymbol{\mathsf{Q}}, \boldsymbol{\mathsf{T}}, \boldsymbol{\mathsf{R}}, \boldsymbol{\mathsf{S}}$	Weighting matrix in the MPC objective function
PH	Prediction horizon for MPC
CH	Control horizon for MPC
$\bar{s}$	Standard deviation (around mean) of the process variable
ρ	scheduling variable(s)
$\Sigma(\rho)$	Set of linear plant models
P	Pressure measurements
$\Pi(\rho)$	Set of linear controllers

List of Abbreviations

CMI	Continuous Model Identification
FO	Fractional Order
IO	Integer Order
FC	Fractional Calculus
CFOTDS	Commensurate Fractional Order Time Delay System
SVF	State Variable Filters
RIVC	Refined Instrument Variable for Continuous Systems
IV	Instrument Variable
GL	Gr$\ddot{u}$nwald-Letnikov
RL	Riemann-Liouville
SISO	Single Input Single Output
BIBO	Bounded Input Bounded Output
CFE	Continuous Fraction Expansion
SNR	Signal to Noise Ratio
MC	Monte Carlo
PRBS	Pseudo Random Binary Sequence
CW	Classical Wall
PSO	Particle Swarm Optimization
FO-PI	Fractional Order Proportional Integral

CSTH	Continuous Stirred Tank Heating
PID	Proportional Integral Derivative
FO-PID or $PI^{\lambda}D^{\mu}$	Fractional Proportional Integral Derivative
ITAE	Integral Time Absolute Error
ISE	Integral Square Error
IMC	Internal Model Control
DCS	Distributed Control System
OPC	Object Linking and Embedding for Process Control
DPS	Distributed Parameter System
ODE	Ordinary Differential Equations
PDE	Partial Differential Equations
OE	Output Error
CONTSID	Continuous System Identification
PPE	Percent Prediction Error
FHC	Froth Heater Control
PSV	Primary Separation Vessel
SepCell	Separation Cell
MPC	Model Predictive Control
MV	Manipulating Variable
CV	Controlled Variable
DV	Disturbance Variable
MISO	Multi Input Single Output
MIMO	Multi Input Multi Output
FF	Feed Forward

ARX	Auto Regressive Exogenous
QP	Quadratic Programming
CPO	Controller Performance Optimizer
APC	Advanced Process Control
KPI	Key Performance Index
GS	Gain Scheduling
OC	Operating Condition
LTI	Linear Time Invariant
GPC	Generalized Predictive Control

Chapter 1

Introduction

1.1 Motivation

Over the last few decades, automatic 'process control' systems have emerged as the standard technology in use in all major process industries, be it oil & gas, chemical or manufacturing. These systems are crucial for safe plant operation, improved product quality and production efficiency and also help in reducing environmental footprints. The word process control contains two important key words, 'process' and 'control'. Understanding the process and its behavior is equally important as the control aspect for safe plant operation.

To be able to predict something in the near future has always been part of human thinking and this idea is used in every sphere of human living, science and technology. In this regard, we have developed process understanding based on past experience and present knowledge, to predict something in the near future. Understanding requires complete analysis of the process and models are the most appropriate tools to analyze any system or process. In the rest of the thesis, the terms process and system will be used interchangeably.

Mathematical models of a dynamic process take various forms, such as differential equations, partial differential equations, state-space equations, difference equations, transfer functions, etc. Most of the real processes are modeled by constructing mathematical equations based on physical laws that govern the behavior of the process also known as being modeled using first principles. These include use of physical laws based on mass, energy or momentum balance. The development of first principle models is usually time-consuming and needs a lot of effort, especially for a complex process (Henson, 1998). In practice, a large number of processes are, firstly, poorly understood, and secondly, too complicated to be modeled from first principles. Also, it is not easy to estimate first principle models directly from process data. This is because of identifiability problems which are encountered by over-parametrization of the model. Process or system identification is an important tool for modeling dynamic processes from measured data with a reasonable degree of accuracy in the absence of detailed first principle models.

In the field of mathematical modeling, based on the amount of a-priori information used or available for a process, models can be classified into black box or white box models. In black box models, no a priori information is available for a process while for a white box model, all necessary information is available for model design. Practically all processes have different shades of grey, meaning that they lie somewhere between the black box and white box models. Ljung (1999) in his famous book on system identification quoted that " ...Our acceptance of models should thus be guided by 'usefulness' rather than 'truth'...". Thus, system identification methods that use data from the process are popular in the industry because these models are easy to understand, and often provide quick solutions.

The practical application of a model has to be taken into consideration before their use in industry. The identified models based on the nature of the relationship can be broadly classified as: linear and nonlinear models. It is well known that a majority of the real world processes are nonlinear by nature; however, the objective is always to get a reasonably accurate representation of the true process, which at times can be obtained using localized linear models. Also, it is easier to obtain a representation of a process if the process is assumed to be locally linear. System identification methodology in the area of linear models is well developed (Ljung, 1999; Söderström and Stoica, 1989). For linear time series models, a wide variety of model structures are available for capturing the dynamics of a system with respect to known inputs and unmeasured disturbances (Ljung, 1999). However, handling time delays in modeling has always been challenging. It is not possible to identify an accurate model of the process because of a number of reasons. In practice, the true model order of the process is not always exactly known and also data length for identification is finite and data contains unmeasured disturbances/ noise. In this respect, the stochastic nature of the process makes the identification problem challenging.

For linear models, discrete time models are very well researched. However, most often the model parameters for a discrete model have no physical interpretation compared to the continuous-time models. In contrast, continuous-time model identification (CMI) is increasingly becoming more popular. The parameters of the identified model are estimated in continuous domain and thus each of the model parameters is associated with some physical significance. The physical systems are inherently continuous in nature; the use of continuous-time models in controller design and the strong relation between model parameters and the system properties are the major forces for developments in continuous time identification. Details on the CMI can be found in Sinha (1991); Unbehauen and Rao (1998); Young et al. (2003).

Fractional order systems which are described by continuous fractional order transfer functions have become an increasingly interesting topic of research in the scientific and industrial communities over last decade. However, identifying a given system from data becomes more difficult when the physical systems are characterized by fractional-order models instead of classical integer order models. Again, the idea is simple; here we want to build fractional order models for fractional order systems when it is not possible to use first principle's models. The models are build using CMI identification techniques. CMI for rational (or integer) order models are very well researched and the recent book by Garnier and coworkers (Garnier et al., 2008) presents the preliminary results in this area and discusses some of the existing methods. Unlike CMI for integer order models, for fractional order systems, some work has been carried out in the last few years but the proposed models and algorithms are still in a preliminary stage of establishment. Also, Chen (Chen, 2006) has argued that fractional order control is ubiquitous when the dynamic system is of distributed parameter nature (DPS). Whenever material or energy is physically moved in a process or a plant, there is usually a time delay associated with the movement. Apparent time delays may result in the identification exercise when a higher order process (or DPS) is approximated by a lower order model. Both the equation error based and output error based approaches have been explored in the literature for fractional order models. However, none of the studies discusses methods for identification of fractional order system models with time delays. It is of our belief that fractional calculus and integral calculus will be synonymous in a few years because of the generic nature of fractional calculus. Thus, in the present context, the work in this thesis explores continuous-time models for parsimonious parameter representation of systems, including distributed parameter systems.

The developed model based on the process data could be used for a number of purposes: control, fault detection and isolation, prediction, etc. In this work, the focus is on the control design using the identified model(s) for chemical processes. Even though most industrial processes are nonlinear and highly complex, linear models are often sufficient to approximate a process around a single operating point. A common approach for modeling nonlinear processes is to use multiple linear models which when combined together can cover the entire operating range of the nonlinear process. The assumption here is that the process is locally linear within each of these operating regimes. Also, with so much theory that has been developed in the area of linear control design and the fact that linear models simplify the control design, linear models are easy to work with when the control involves rejection of small disturbances. All simple control schemes such as industrial PID controllers to more advanced predictive controls like model predictive controller (MPC) use linear empirical models of the process for control design. Building data based input-output linear models is one of the most widely used method to design a model based control scheme for industrial processes.

PID controllers are still widely used in the industry because they are easy to implement and perform well for wide class of processes. It is argued that fractional order systems require much more than classical PID controllers to achieve good closed loop performance. A fractional-order PID controller is considered as the generalization of the conventional PID controllers (Podlubny, 1999*a*). Fractional order PID controllers, which provide additional tuning parameters, can provide better closed loop performance and robustness features compared to classical PID controllers. PID controllers in general are not very well suited for control of processes with long dead time characteristics since they can cause stability issues for these closed loop systems. The controller design of these processes is a challenging problem. The most popular and very effective dead-time compensator in use today is the Smith predictor (Smith, 1957). For processes with time delays, the Smith predictor structure and the internal model controller (IMC) structure are equivalent. Even for fractional order PID controllers, modifications in the control scheme such as the Smith predictor, can be made to handle time delays. IMC works well for single input single output (SISO) processes; however, to deal with process interactions and constraints in multivariable processes, more advanced control schemes such as model predictive control have gained significant industrial adoption. In contrast to other controllers, MPC computes optimal control moves by solving an optimization problem over a finite horizon, taking into account dynamic behavior of the process as well as the operational constraints (Qin and Badgwell, 2003; Rawlings, 2000). They also present the ease for continuous operation over long periods of time without expert intervention.

There are a lot of complex and challenging problems found in the process industries, and this in turn has led to significant research activities in the field of model predictive control over past three decades. Given a dynamic model of the system under consideration, the issues such as controller design to ensure stability of MPC schemes (Rawlings and Muske, 1993) and efficient online computations have been very well researched in the process control literature. The key component of any MPC scheme is the dynamic model used for carrying out on-line predictions. Development and maintenance of the dynamic model is of paramount importance for achieving good closed loop performance. Industrial applications of MPC rely mainly on linear empirical models obtained by employing time series analysis (Qin and Badgwell, 2003). Control design based on linearized dynamics need not exhibit good performance or even stabilizing when operating away from the equilibrium. In the case of changing operating conditions for a process, one way to handle the problem of maintaining the stability and/or performance of a linear MPC scheme is by employing multiple model based controller design. Gain-scheduling (GS) control using set of linear models can then be used to cover the entire operating range of the process.

The focus of this thesis is that identification and control go hand in hand and it is equally important to have a high fidelity and parsimonious model (model with fewer parameters), to design a good model based controller. The objective is to build data based models (parsimonious models) for integer and fractional order systems which can represent a process sufficiently well and use the model(s) to design controllers (simple PID, fractional order PID, IMC, MPC etc.) to provide good robustness and closed loop performance.

1.2 Thesis Overview

1.2.1 Thesis Contributions

The research presented in this thesis and the major contributions that distinguishes it from the other existing work are listed below:

1. Formulation of a continuous-time model identification method for fractional order system with time delays. Using the proposed method, time delays are simultaneously estimated along with other model parameters for a commensurate fractional order model with time delays (CFOTDS).

2. Evaluation of the proposed identification method is performed on a simulated heat transfer problem of classical wall, an example of distributed parameter systems. The proposed method is also applied to two other distributed pa-

rameter systems, industrial froth heater and experimental setup of stirred tank heater with transportation delays, and the results are compared with the rational (integer) order models.

3. Formulated a servo control strategy for tuning of fractional order PI (FO-PI or PI$^{\lambda}$) controllers for fractional order and integer order systems using a reference model, whose open-loop transfer function is given by the Bode's ideal transfer function. A global optimization tool, namely particle swarm optimization (PSO) is used for optimization based on minimization of a quadratic cost function satisfying the constraints.

4. Evaluation of the proposed fractional order PI controller tuning method is performed on a simulated heat transfer problem of classical wall, lag and delay dominant processes as well as on a computer-interfaced pilot scale continuous stirred tank heater (CSTH) process. The controller performance is compared with some of the existing methods to tune conventional PI controllers and fractional PI controllers.

5. A case study application of identification and model based predictive control design for an industrial separation cell process: Internal model control and model predictive control using linear models are designed, implemented and tested in real time on the industrial separation cell.

6. A gain scheduling model predictive controller is designed for a non-square multi-variable problem of industrial froth heater process. The process has more controlled variables than manipulating variables; using linear models at different operating conditions, a gain scheduling MPC is designed and tested in simulations to see the benefits of multivariate MPC.

1.2.2 Thesis Outline

This thesis has been prepared as per the guidelines from the Faculty of Graduate Studies and Research (FGSR) at the University of Alberta. The rest of the thesis is organized as follows.

Chapter 2 describes the identification method for fractional order models with time delays and presents Monte Carlo simulations on fractional order systems in the presence of noise as well as simulations to identify a process of thermal diffusion in a wall. These dynamic models using fractional order transfer functions are built using little a priori knowledge about the process.

Chapter 3 describes a tuning method to design fractional order PI controllers for both integer order and fractional order systems using Bode's ideal transfer function as the reference system. The simulation and experimental case study is presented to validate the benefits of designing fractional order PI controllers.

Chapter 4 provides details on distributed parameter systems and using fractional order models as a plausible way to model these infinite dimensional systems.

Chapter 5 presents industrial case studies on modeling and designing model based predictive controllers for a separation cell process. Model based predictive controllers are designed, implemented and tested in real time on the industrial separation cell.

Chapter 6 presents industrial case studies on modeling and design of a gain scheduling MPC for an industrial froth heater process. The benefits of this scheme is discussed using simulations based on models obtained from industrial data.

Chapter 7 draws the conclusion based on the work that has been done in this thesis and provides some perspectives for future research related to the work presented in this thesis.

Bibliography

M. A. Henson. Nonlinear model predictive control: Current status and future directions. *Computers and Chemical Engineering*, vol. 23(2), pp. 187-202, 1998.

L. Ljung. System Identification: Theory for the Users. *Prentice-Hall*, 2nd ed., New Jersey, 1999.

T. Söderström and P. Stoica. System Identification. *Prentice-Hall: Upper Saddle River*, New Jersey, 1989.

N. K. Sinha and G. P. Rao. Identification of Continuous-time Systems: Methodology and Computer Implementation. *Kluwer Academic Publishers*, 1991.

H. Unbehauen and G. P. Rao. A review of identification in continuous-time systems. *Annual Reviews in Control*, vol. 22, pp. 145-171, 1998.

P. C. Young, H. Garnier and A. Jarvis. The identification of continuous-time linear and nonlinear models: A tutorial with environmental applications. *Proceedings of the 13th IFAC Symposium on System Identification*, Rotterdam, Netherlands, pp. 618-628, 2003.

H. Garnier and L. Wang Identification of Continuous-time Models from Sampled Data. *Advances in Industrial Control*, Springer, London, 2008.

Y.Q. Chen. Ubiquitous fractional order controls. *Proc. of The Second IFAC Symposium on Fractional Derivatives and Applications (IFAC FDA06, Plenary Paper)*, pp. 19-31, 2006.

I. Podlubny. Fractional order systems and $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$- controllers. *IEEE Trans. Automat. Control AC*, vol. 44 (1), pp. 208-214, 1999.

O. J. M. Smith. Closer control of loops with dead time. *Chem. Eng. Prog.*, vol. 53(5), 1957.

S.J. Qin and T.A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, vol. 11, pp. 733-764, 2003.

J. B. Rawlings. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, vol. 20, pp. 38-52, 2000.

J. B Rawlings and K. R. Muske. The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, vol. 38(10), pp. 1512-1516, 1993.

Chapter 2

Continuous-time model identification of fractional order models with time delays [1]

Modeling of real physical systems having long memory transients and infinite dimensional structures using fractional order dynamic models has attracted significant interest over the last few years. For this reason, many identification techniques both in the frequency domain and time domain have been developed to model these fractional order systems. However, in many processes time delays are also present and estimation of time delays along with continuous time fractional order model parameters have not been addressed anywhere. This chapter deals with the continuous-time model identification (CMI) of fractional order systems with time delays. In this chapter, a new linear filter is introduced for simultaneous estimation of all model parameters for commensurate fractional order systems with time delays (CFOTDS). The proposed method simultaneously estimates time delays along with other model parameters in an iterative manner by solving simple linear regression equations. For the case when the fractional order is unknown, we also propose a nested loop optimization method where the time delay along with other model parameters are estimated iteratively in the inner loop and the fractional order is estimated in the non-linear outer loop. The applicability of the developed procedure is demonstrated by Monte Carlo simulations on fractional order models in the presence of white noise. The proposed algorithm has also been applied to identify a process of thermal diffusion in a wall in simulation, which are

[1] The work reported in this chapter has been published as: "Continuous-time model identification of fractional order models with time delays, IET Control Theory and Applications, DOI: 10.1049/iet-cta.2010.0718". A condensed version of this chapter was presented at the 15th IFAC Symposium on System Identification (SYSID 2009), July 2009, France (Narang et al., 2009)

characterized by fractional order behavior.

2.1 Introduction

Fractional calculus is a generalization of the traditional integer order integral and differential calculus to non-integer orders. With the growing power of computers, fractional calculus now has become an increasingly interesting topic of research in the scientific and industrial communities. In the last two decades there has been a considerable development in the use of fractional operators in various fields. Before the 20th century, the theory of fractional calculus developed mainly as a pure theoretical field of mathematics useful only for mathematicians. A significant amount of discussions aimed at this subject has been presented by Oldham and Spanier (1974) and Podlubny (1999). However, recently it has been observed that many real-world physical systems are well characterized by fractional-order differential equations rather than using classical integer order models. In particular, materials having long memory and hereditary effects (Bagley and Torvik, 1984) and dynamical processes such as mass diffusion and heat conduction (Jenson and Jeffreys, 1974) in fractal porous media can be more adequately modeled by fractional-order models rather than integer-order models. Some of the other examples of fractal systems include transmission lines, electrochemical processes, dielectric polarization and viscoelastic materials. Diffusive interfaces are particularly characterized by fractional order dynamic behavior, such as it appears in the case of an induction machine, with Foucault currents inside rotor bars (Benchellal et al., 2004) and heat transfer model relating flux and the temperature at the diffusive interface (Cois, 2002; Benchellal et al., 2006; Gabano and Poinot, 2011). The special issue of signal processing (Ortigueira and Tenreiro, 2006) discusses in detail many applications of fractional calculus in different fields.

System identification has become a standard tool for modeling unknown systems. However, identifying a given system from data becomes more difficult when the physical systems are characterized by fractional-order differential equations instead of classical integer order models. Thus, fractional models, using fractional differentiation, have been developed. The identified models for fractional order systems can also be used to design controllers for these systems which may not be, just classical integer order controllers. For integer-order systems, the parameters of the model equation can be optimized directly once the maximum order of the system to be identified is fixed, while for fractional-order systems, identification requires the choice of the fractional powers (orders) of the operators, and also the coefficients of the operators. Thus, loss of integer order significantly complicates the identification process. Time-domain system identification using fractional differentiation models is initiated by Lay (Lay, 1998), Lin (Lin, 1998) and Cois (Cois, 2002), in their PhD thesis work in the late 1990s. The two model identification approaches developed were: Equation error based and output error based approaches, both of which are very well studied in the literature for integer order models. As in the case of continuous model identification for integer order models, fractional differentiation of noisy signals also amplifies noise. Hence, a linear transformation using low-pass filter can be applied to the model equation. As for the integer case, continuous time model identification using linear filter methods have also been proposed for FO models such as fractional integral filter, Poisson's state variable (SVFs) filters (Cois et al., 2001), and Refined Instrumental Variable for Continuous systems (RIVC) (Malti et al., 2008*a*). Using the iterative instrument variable (IV) approach it has been shown in Malti et al. (2008*a*) that the simplified RIVC method is asymptotically unbiased, given the true system is in the right model class. They also proposed an algorithm for optimizing the commensurate fractional order using a gradient based method (Victor et al., 2009). Many of these identification methods have already been applied to model some real physical processes for example see Benchellal et al. (2004); Malti et al. (2009); Sabatier et al. (2006). Use of fractional calculus theory for building parsimonious models can be found in Muddu et al. (2009, 2010). Also, identification methods based on orthogonal basis functions such as fractional Laguerre and Kautz basis functions, have been proposed (Aoun et al., 2007). The recent paper by Malti et al. (2008*b*) discusses briefly all these advances in time-domain system identification using fractional models.

However none of these studies discuss methods for identification of fractional order system models with time delays. Whenever material or energy is physically moved in a process or a plant, there is usually a time delay associated with the movement (Seborg et al., 1989). Time delay is also referred to as dead time, transportation lag or distance-velocity lag. Also, apparent time delays may result due to measurements or actuators in a process or in the identification exercise when a higher order process is approximated by a lower order model. It is reported in Manabe (2003) that due to actuator limitations in some systems such as motion control, the system can be well modeled with a FO open-loop transfer function model with time delay. Malti et al. (2009) noticed the time-lag in flux diffusion while modeling a thermal rod process (a fractional order system) from experimental data. In this work, the focus is to develop a linear filter method based on the equation error (EE) approaches for direct identification of continuous-time transfer function models. In this thesis, we describe a scheme for continuous time identification of commensurate fractional order models with time delays. The proposed algorithm is a extension of the authors' previous work (Narang et al., 2009). We propose two formulations based on the type of input signal excitation. The first formulation is based on step input excitation and the second one applies to more generic (RBS, PRBS, Sinusoidal etc) kind of input signal excitation. In this scheme the delay is estimated simultaneously with all other model parameters. The formulation as developed in Ahmed et al. (2007) for integer order continuous-time systems is extended to identification of fractional system models. Wang and Zhang (2001) proposed a method to estimate time delay along with other parameters for integer order models using a step input. To the best knowledge of the author no formulation for estimating all model parameters including delays has been proposed for fractional models. The formulation is based on the low pass filtering operation where the filter is chosen as the combination of RIVC and a linear integral filter, to make the delay term appear as explicit parameter along with other parameters. The proposed method estimates the time delay along with constant model parameters in an iterative manner by solving simple linear regression equations. In the presence of noise, a modified scheme using instrument variable method where instruments are build based on auxiliary model is developed. For fractional models the aim is also to identify fractional powers along with other model parameters. Here, we also propose a nested loop optimization method where the time delay along with constant model parameters are estimated iteratively in the inner loop and the commensurate order is estimated in the non-linear outer loop for commensurate type transfer function models. The advantage of working with commensurate order models is that all fractional powers in the model are integer multiple of a single fractional order and therefore we need to estimate only one term in the outer loop. The proposed method is generic in the sense that it can also be applied to integer order transfer function models.

The remainder of this chapter is organized as follows. Section 2.2 presents a brief mathematical background of fractional calculus with an introduction to fractional order models. The continuous time model identification algorithm for CFOTDS using step input and any other generic kind of input excitation is presented in Section 2.3. To study the efficacy of the proposed strategy developed in the Section 2.3, different examples of fractional order models in the presence of noise are outlined in Section 2.4. Section 2.5 discusses the results for the proposed algorithm applied to a real process of thermal diffusion in a wall, followed by concluding remarks in Section 2.6.

2.2 Mathematical background

2.2.1 Definitions and FO models

Fractional calculus is a generalization of integration and differentiation to non-integer orders. The two most popular definitions used to describe fractional differentiation and integration are the Grünwald-Letnikov (GL) discrete form of the definition and the Riemann-Liouville (RL) definition (Oldham and Spanier, 1974). The GL definition for a function $f(t)$ is given as

$$\mathrm{D}^{\alpha} f(t) = \lim_{h \to \infty} \frac{1}{h^{\alpha}} \sum_{i=0}^{\infty} [(-1)^{i} \left(\begin{array}{c} \alpha \\ i \end{array} \right) f(t-ih)], \tag{2.1}$$

where

$$\left(\begin{array}{c} \alpha \\ i \end{array} \right) = \frac{\mathbf{\Gamma}(\alpha+1)}{\mathbf{\Gamma}(i+1)\mathbf{\Gamma}(\alpha-i+1)}, \tag{2.2}$$

and the operator D^{α} defines fractional differentiation or integration depending on the sign of α, $\mathbf{\Gamma}(.)$ being the well known Euler's Gamma function and h is the finite sampling interval. This definition is particularly useful for digital implementation of the fractional operator. The RL definition is given as

$$\mathrm{D}^{\alpha} f(t) = \frac{d^m}{dt^m} \left[\frac{1}{\Gamma(m-\alpha)} \int_0^t \frac{f(\tau)}{(t-\tau)^{\alpha+1-m}} d\tau \right] \tag{2.3}$$

where m is an integer such that $(m-1 < \alpha < m)$ and $t > 0 \quad \forall\ \alpha \in \mathbb{R}_+$. For convenience, the Laplace domain notation is usually used to describe fractional differentiation-integration operation. When the initializations are assumed to be zero,

$$\pounds\{\mathrm{D}^{\alpha} f(t)\} = s^{\alpha} F(s) \qquad (\alpha \in \mathbb{R}) \tag{2.4}$$

The generic single-input single-output (SISO) fractional order system representation in the Laplace domain is given as

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\kappa_0 s^{\beta_0} + \kappa_1 s^{\beta_1} + ... + \kappa_m s^{\beta_m}}{1 + \varpi_1 s^{\alpha_1} + ... + \varpi_n s^{\alpha_n}} \tag{2.5}$$

where $\kappa_0, \kappa_1, ..., \kappa_m$ and $\varpi_1, \varpi_2, ..., \varpi_n$ are constant model parameters or model coefficients, while $\beta_0 < \beta_1 < ..... < \beta_m$ and $\alpha_1 < \alpha_2 < ..... < \alpha_n$ are the fractional powers or fractional orders (real numbers). The transfer function (2.5) is called non-commensurate when β_j, α_i can take any arbitrary values.

The transfer function as given by equation (2.5) can be classified as a commensurate transfer function. A transfer function $G(s)$ is commensurable of order γ if and only if it can be written as $G(s) = F(s^\gamma)$, where $F = T/R$ is a rational function, with T and R as two coprime polynomials. Assuming that $G(s)$ is commensurate transfer function of order γ, then it can be written as

$$G(s) = \frac{\sum_{j=0}^{m} b_j s^{j\gamma}}{1 + \sum_{i=1}^{n} a_i s^{i\gamma}} \tag{2.6}$$

Therefore, for commensurate transfer function all fractional powers are integer multiple of a real number, γ. Commensurate transfer function models represent more generic class of polynomial type transfer functions where $\gamma = 1$ gives standard integer order transfer function models. The analytical expression for fractional order systems in the transfer function form is most commonly (more than 50%) described by commensurate transfer functions. A commensurate transfer function of order γ for fractional-order time delay system is given as

$$G(s) = \frac{\sum_{j=0}^{m} b_j s^{j\gamma}}{1 + \sum_{i=1}^{n} a_i s^{i\gamma}} e^{-Ls} \tag{2.7}$$

where L is the time delay. In this work we will be working only with commensurate transfer function models with delays as described in equation (4.2).

2.2.2 Stability condition

Stability condition for a class of transfer function of the form (2.6) has been established by Matignon (1998). The theorem is as follows:

Stability Theorem: *A commensurable* $\gamma-order\ transfer\ function$ $G(s) = F(s^\gamma) = T(s^\gamma)/R(s^\gamma)$, *where* $T(.)$ *and* $R(.)$ *are two coprime polynomials*, *is* $BIBO$ *stable* if *and only* if

$$0 < \gamma < 2$$

and $\forall\ \sigma \in \mathcal{C}$ *such that* $R(\sigma) = 0$

$$|\arg(\sigma)| > \gamma\frac{\pi}{2}$$

2.2.3 Integer order approximation

The modeling and simulation of fractional order systems are complicated due to their long memory behavior (Oustaloup, 1995) and are based on the approximation (approximating the infinite dimensional nature) of the fractional derivative operator. For digital implementation of the fractional order operator, the key step is numerical evaluation or discretization of this operator. In most of the cases, it is not easy to obtain analytical expressions of the output for a given input excitation for a fractional order transfer function model. Two classes of methods developed over the last few years to approximate the fractional derivative operator can be classified as: direct methods - based on the approximation of a fractional derivative operator by a rational discrete time one, and indirect methods - based on the approximation of a fractional derivative operator by a rational continuous-time one. Power series expansion and continuous fraction expansion (CFE) of the Euler's, Tustin and Al-Alaoui operators give different discrete approximations of the fractional operator. The power series expansion of Euler's operator gives numerical approximation of the GL definition as in equation (2.1). The details for the discretization schemes can be found in (Vinagre et al., 2002; Chen and Moore, 2002). One of the good continuous approximation for this fractional order operator compared to GL definition is the Oustaloup continuous approximation (Oustaloup, 1995) where it makes use of a recursive distribution of poles and zeros. In this work, we will be using the Oustaloup continuous approximation for the simulation of fractional order transfer functions. Additional details on this appear below.

Many real physical systems generally have bandlimited fractional behavior and also due to the practical limitations of input and output signals (Shannons cutoff frequency for the upper band and the spectrum of the input signal for the lower band), fractional operators are usually approximated by high order rational models. As a result, a fractional model and its rational approximation have the same dynamics within a limited frequency band. The Oustaloup approximation of s^λ in the frequency band $[\omega_A, \omega_B]$ has been defined as

$$s^\lambda \to s^\lambda_{[\omega_A,\omega_B]} = C_0 \left(\frac{1+\frac{s}{\omega_A}}{1+\frac{s}{\omega_B}} \right)^\lambda \approx C_0 \prod_{k=1}^{N} \frac{1+\frac{s}{\acute{\omega}_k}}{1+\frac{s}{\omega_k}} \tag{2.8}$$

where

$$\omega_i = \alpha\ \acute{\omega}_i, \quad \acute{\omega}_{i+1} = \eta\ \omega_i \quad \text{and} \quad \lambda = 1 - \frac{\log\ \alpha}{\log\ \alpha\eta} \tag{2.9}$$

where the parameter C_0 is chosen such that the approximation shall have a unit gain at 1 rad/s.

Note that the developed method is independent from the way fractional differentiation and integration are simulated in the time domain. However, approximation error will arise if we don't use a good approximation to simulate the fractional order operator. We have used the Oustaloup approximation in this chapter.

2.3 Problem Statement

2.3.1 Identification formulation

The transfer function for CFOTDS of commensurate order α is given as

$$G(s) = \frac{\sum_{j=0}^{m} b_j s^{j\alpha}}{1 + \sum_{j=1}^{n} a_j s^{j\alpha}} e^{-Ls} \tag{2.10}$$

For integer order models, $\alpha = 1$ and only the model coefficients a_j, b_j and L are estimated. However, here we are interested in estimating α as well. For the present case, initial conditions are assumed zero and the model in the vector form can be represented as

$$\mathbf{a}_n \mathbf{s}^{n\alpha} Y(s) = \mathbf{b}_m \mathbf{s}^{m\alpha} U(s) e^{-Ls} + e(s) \tag{2.11}$$

where

$$\begin{aligned} \mathbf{a}_n &= [a_n \ a_{n-1} \ .... \ a_1 \ 1] \ \in \ R^{1\times(n+1)} && (2.12)\\ \mathbf{b}_m &= [b_m \ b_{m-1} \ .... \ b_1 \ b_0] \ \in \ R^{1\times(m+1)} && (2.13) \end{aligned}$$

$$\begin{aligned} \mathbf{s}^{n\alpha} &= [s^{n\alpha} \ s^{(n-1)\alpha} \ .... \ s^{\alpha} \ s^0]^T \ \in \ R^{(n+1)\times 1} && (2.14)\\ \mathbf{s}^{m\alpha} &= [s^{n\alpha} \ s^{(m-1)\alpha} \ .... \ s^{\alpha} \ s^0]^T \ \in \ R^{(m+1)\times 1} && (2.15) \end{aligned}$$

and $Y(s)$,$U(s)$ and $e(s)$ are the Laplace transforms of output $y(t)$, input $u(t)$ and $e(t)$ respectively. The term $e(t)$ accounts for the noise which is assumed to be uncorrelated with $y(t)$. Note that here we are using an equation error approach for estimating a continuous time model.

Parameter estimation using a filtering approach has been very well established method available in the literature; however, to estimate only the parameters (a_n and b_m) but not the delay. Victor et al. (2009) proposed a continuous-time identification method with optimal fractional differentiation order for fractional order systems. The estimation of delay is mathematically different from the estimation of other parameters because the other parameters appear explicitly in the model while the delay appears implicitly as can be seen in equation (2.10). Next, we devise a linear filter method for estimation of model parameters. To obtain explicit appearance of the delay term in the estimation equation and have it appear as an element in the parameter vector, we introduce a linear filter method with a structure of the filter as a combination of RIVC and a linear integral filter. This structure of a filter has been introduced by Ahmed et al. (2007) for rational order models. This low pass filter not only serves the purpose of removing noise amplification but it also makes the delay term appear as a explicit parameter to be estimated along with the other parameters. The filter transfer function is represented as

$$F(s^{\alpha}) = \frac{1}{sA(s^{\alpha})} \tag{2.16}$$

where $A(s^{\alpha})$ is the denominator of the model equation. Now applying the filtering operation on both sides of equation (2.11) yields

$$\mathbf{a}_n \mathbf{s}^{n\alpha} F(s^{\alpha}) Y(s) = \mathbf{b}_m \mathbf{s}^{m\alpha} F(s^{\alpha}) U(s) e^{-Ls} + F(s^{\alpha}) e(s) \tag{2.17}$$

or

$$\mathbf{a}_n \mathbf{s}^{n\alpha} \frac{1}{sA(s^{\alpha})} Y(s) = \mathbf{b}_m \mathbf{s}^{m\alpha} \frac{1}{sA(s^{\alpha})} U(s) e^{-Ls} + \varsigma(s) \tag{2.18}$$

where $\varsigma(s) = F(s)e(s)$. Here F(s) can be factored as

$$\frac{1}{sA(s^{\alpha})} = \frac{C(s^{\alpha})}{sA(s^{\alpha})} + \frac{1}{s} \tag{2.19}$$

where

$$C(s^{\alpha}) = -(a_n s^{n\alpha} + a_{n-1} s^{(n-1)\alpha} + ... + a_1 s^{\alpha}) \tag{2.20}$$

Also, $\mathbf{a_n} \mathbf{s}^{n\alpha}$ and $\mathbf{b_m} \mathbf{s}^{m\alpha}$ can be factored as

$$\mathbf{a}_n \mathbf{s}^{n\alpha} = (\bar{\mathbf{a}}_n \mathbf{s}^{(n-1)\alpha} s^{\alpha} + 1) \tag{2.21}$$

and

$$\mathbf{b}_m \mathbf{s}^{m\alpha} = (\bar{\mathbf{b}}_m \mathbf{s}^{(m-1)\alpha} s^{\alpha} + b_0) \tag{2.22}$$

where $\bar{\mathbf{a}}_n$ and $\bar{\mathbf{b}}_m$ are the $\mathbf{a}_n$ and $\mathbf{b}_m$ vectors respectively with the last element removed. Now defining the filtered output and input variables as

$$Y_f(s) = \frac{Y(s)}{sA(s^{\alpha})} \quad \text{and} \quad Y_{f_D}(s) = \frac{s^{\alpha} Y(s)}{sA(s^{\alpha})} \tag{2.23}$$

$$U_f(s) = \frac{U(s)}{sA(s^{\alpha})} \quad \text{and} \quad U_{f_D}(s) = \frac{s^{\alpha} U(s)}{sA(s^{\alpha})} \tag{2.24}$$

Thus equation (2.18) becomes

$$\begin{aligned} Y_f(s) = &-\bar{\mathbf{a}}_n \mathbf{s}^{(n-1)\alpha} Y_{f_D}(s) + \bar{\mathbf{b}}_m \mathbf{s}^{(m-1)\alpha} U_{f_D}(s) e^{-Ls} \\ &+ b_0 \left(\frac{C(s^\alpha)}{sA(s^\alpha)} + \frac{1}{s} \right) U(s) e^{-Ls} + \varsigma(s) \end{aligned} \tag{2.25}$$

Defining additional filtered variables as

$$U_{f_C}(s) = C(s^\alpha) U_f(s) = \frac{C(s^\alpha)}{sA(s^\alpha)} U(s) \tag{2.26}$$

$$U_I(s) = \frac{U(s)}{s} \tag{2.27}$$

Then equation (2.25) can be written as

$$\begin{aligned} Y_f(s) = &-\bar{\mathbf{a}}_n \mathbf{s}^{(n-1)\alpha} Y_{f_D}(s) + \bar{\mathbf{b}}_m \mathbf{s}^{(m-1)\alpha} U_{f_D}(s) e^{-Ls} \\ &+ b_0 U_{f_C}(s) e^{-Ls} + b_0 U_I(s) e^{-Ls} + \varsigma(s) \end{aligned} \tag{2.28}$$

Before taking the Laplace inverse on both sides, we define the Laplace inverse for various terms

$$\pounds^{-1}(Y_f(s)) = y_f(t) \tag{2.29}$$

$$\pounds^{-1}(Y_{fD}(s)) = y_{f_D}(t) \tag{2.30}$$

$$\pounds^{-1}(U_{fD}(s)) = u_{f_D}(t) \tag{2.31}$$

$$\pounds^{-1}(\varsigma(s)) = \varsigma(t) \tag{2.32}$$

$$\pounds^{-1}\left(\mathbf{s}^{(n-1)\alpha} Y_{f_D}(s)\right) = y_{f_D}^{(n-1)\alpha}(t) \tag{2.33}$$

$$\pounds^{-1}\left(\mathbf{s}^{(m-1)\alpha} U_{f_D}(s) e^{-Ls}\right) = u_{f_D}^{(m-1)\alpha}(t-L) \tag{2.34}$$

$$\pounds^{-1}(U_{f_C}(s)) = u_{f_C}(t) \tag{2.35}$$

$$\pounds^{-1}(U_{f_C}(s) e^{-Ls}) = u_{f_C}(t-L) \tag{2.36}$$

and

$$\pounds^{-1}(U_I(s)) = \int\limits_0^t u(t) dt \tag{2.37}$$

$$\pounds^{-1}(U_I(s) e^{-Ls}) = u_I(t-L) \tag{2.38}$$

Now, depending upon the type of input signal used for perturbation we propose two different formulations for estimating the model parameters.

Using step input signal

In continuous model identification, by using a step input signal we can estimate all the parameters of a higher dimensional model. If we have a step input signal of step size h, then

$$U(s) = \frac{h}{s} \tag{2.39}$$

then equation (2.38) becomes

$$\pounds^{-1}(U_I(s)e^{-Ls}) = \pounds^{-1}\left(\frac{1}{s}\frac{h}{s}e^{-Ls}\right) = h(t-L) \;\; \forall \;\; t > L \tag{2.40}$$

and equation (2.36) can be written as

$$\pounds^{-1}(U_{f_C}(s)e^{-Ls}) = \pounds^{-1}\left(\frac{C(s^\alpha)}{sA(s^\alpha)}\frac{h}{s}\; e^{-Ls}\right) = u_{f_C}(t-L) \tag{2.41}$$

Now taking inverse Laplace transform of equation (2.28) we have

$$\begin{aligned} y_f(t) = -\bar{\mathbf{a}}_n \mathbf{y}_{f_D}^{(n-1)\alpha}(t) + \bar{\mathbf{b}}_m \mathbf{u}_{f_D}^{(m-1)\alpha}(t-L) \\ + b_0 h u_{f_C}(t-L) + b_0 h(t-L) + \varsigma(t) \qquad \text{for } t > L \end{aligned} \tag{2.42}$$

If we define $\mathbf{G}_{f1}$ as

$$\mathbf{G}_{f1} = \begin{bmatrix} \mathbf{u}_{f_D}^{(m-1)\alpha}(t-L) \\ \\ hu_{f_C}(t-L) + ht \end{bmatrix}^T \tag{2.43}$$

then

$$y_f(t) = \left[-\mathbf{y}_{f_D}^{(n-1)\alpha}(t) \qquad \mathbf{G}_{f1} \qquad -h\right] \begin{bmatrix} \bar{\mathbf{a}}_n \\ \mathbf{b}_m \\ b_0 L \end{bmatrix} + \varsigma(t) \tag{2.44}$$

or equivalently

$$\psi(t) = \phi(t)\theta + \varsigma(t) \tag{2.45}$$

where $\theta = \begin{bmatrix} \bar{\mathbf{a}}_\mathbf{n} \\ \mathbf{b}_m \\ b_0 L \end{bmatrix}$. Note that $a_0 = 1$. Similarly, we can consider model (2.45) for all $t = t_k$ where $k = t, t+1, ...N$, such that $t > L$, N being the total number of data points. The stacked terms in this equation then yield the following estimation equation

$$\Psi = \Phi\theta + \Delta \tag{2.46}$$

which is linear in parameter equation and can be solved using linear least squares. However, the delay, L appears as $b_0 L$ but since b_0 is estimated simultaneously in θ, we can estimate delay using this fact. In practice, the selection of the output $y(t)$

after $t > L$ can be made as follows (Bi et al., 1999). When the process output is stationary, the process output will be monitored for a period, the 'listening' period, during which the noise band B_n can be found. Then, $y(t)$ satisfying

$$\arg(y(t)) > 2B_n \tag{2.47}$$

can be treated as the process response after $t > L$, and thus can be used for the model (2.46).

Generic input signal

For any other kind of input signal we use graphical information from the time series input curve. From Fig. 2.1 it can be seen that, for the input signal $U(t)$, the $U_I(t-L)$ term corresponds to area under the $U(t)$ curve over the time instant $(t-L)$. This can be written as the sum of three terms (or representing areas on the curve) as

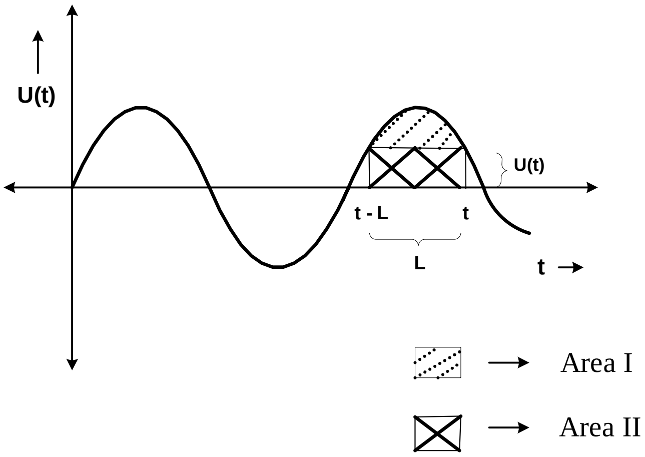


Figure 2.1: Time series process input curve

$$u_I(t-L) = u_I(t) - \overbrace{\int_{t-L}^{t} [u(t_k) - u(t)]dt_k}^{area\ I} - \underbrace{u(t)L}_{area\ II} \tag{2.48}$$

Here, for demonstration purpose we have used a sinusoidal input but the above relationship can be used for any type of input signal excitation. Using the above defined relation (2.48) and taking Laplace inverse of equation (2.28), we have

$$\begin{aligned} y_f(t) &= -\bar{\mathbf{a}}_n \mathbf{y}_{f_D}^{(n-1)\alpha}(t) + \bar{\mathbf{b}}_m \mathbf{u}_{f_D}^{(m-1)\alpha}(t-L) + b_0 u_{f_C}(t-L) \\ &+ b_0 \left[u_I(t) - \int\limits_{t-L}^{t} \left[u(t_k) - u(t)\right] dt_k - u(t)L \right] + \varsigma(t) \end{aligned} \tag{2.49}$$

Now if we again define the augmented $\mathbf{G}_{f2}(t)$ as

$$\mathbf{G}_{f2}(t) \;=\; \left[\begin{array}{c} \mathbf{u}_{f_D}^{(m-1)\alpha}(t-L) \\ u_{f_C}(t-L) + u_I(t) - \int\limits_{t-L}^{t} \left[u(t_k) - u(t)\right] dt_k \end{array} \right] \tag{2.50}$$

then

$$y_f(t) = \left[-\mathbf{y}_{f_D}^{(n-1)\alpha}(t) \quad \mathbf{G}_{f2}(t) \quad -u(t) \right] \left[\begin{array}{c} \bar{\mathbf{a}}_{\mathbf{n}} \\ \mathbf{b}_m \\ b_0 L \end{array} \right] + \varsigma(t) \tag{2.51}$$

The model equation (2.51) can be written as a linear regression equation of the form

$$\psi(t) = \phi(t)\theta + \varsigma(t) \tag{2.52}$$

where

$$\theta = \left[\begin{array}{c} \bar{\mathbf{a}}_{\mathbf{n}} \\ \mathbf{b}_m \\ b_0 L \end{array} \right] \tag{2.53}$$

Similarly, we can consider model (2.52) for all $t = t_k$ where $k = 1, 2, ...N$. The stacked terms in this equation for different times then yield the following estimation equation

$$\Psi = \Phi\theta + \Delta \tag{2.54}$$

Thus, using all the filtered variables and approximating the area for the input curve, we are able to make time delay term appear as an explicit term in the form of a parameter vector in the regression model. Now given any input-output data we can formulate this identification problem as given above and estimate all the parameters using linear least square method.

2.3.2 Parameter estimation

When α is known:

For the case when the commensurate order α is known, we only need to estimate $\mathbf{a}_n$, $\mathbf{b}_n$ and L. Since the filter itself involves the coefficients $\mathbf{a}_n$ and we need L in order to formulate the above linear regression equation, we start with some initial values of $\mathbf{a}_n$ and $\mathbf{L}$, then solving the linear model developed in the previous section using linear least squares we can get a new estimate of the parameter vector θ. This parameter vector also gives us updated estimates of $\mathbf{a}_n$ (note that $a_0 = 1$) and $\mathbf{L}$. The updated values are again used to get the new estimates. The proposed algorithm is similar to the RIVC algorithm except the proposed algorithm formulates an iterative procedure to simultaneously estimate the parameters and the delay, L. However, the delay L appears as $b_0 L$ but since b_0 is estimated separately in θ, we can estimate the delay using this and do it iteratively until until the convergence is achieved for all the parameters. Note that we still have L term coupled with the b_0 term, so any error in estimating one term translates to another.

Instrument variable method

For the cases when the data is corrupted with white noise, the filtering operation converts the white noise signal to colored noise and this algorithm gives biased estimates in the presence of colored noise. Therefore, in order to get unbiased estimates of the parameters, we use the bootstrap instrumental variable (IV) algorithm (Young, 1970) where the instruments are built based on the auxiliary model (using predicted $y (\hat{y})$ instead of measured y values). The instrument variable for the first formulation is then defined as

$$\phi_{IV}(t) = \begin{bmatrix} -\hat{\mathbf{y}}_{f_D}^{(n-1)\alpha}(t) \\ G_{f1}(t) \\ -h \end{bmatrix}^T \tag{2.55}$$

and for the second formulation it is defined as

$$\phi_{IV}(t) = \begin{bmatrix} -\hat{\mathbf{y}}_{f_D}^{(n-1)\alpha}(t) \\ G_{f2}(t) \\ -U(t) \end{bmatrix}^T \tag{2.56}$$

Using this, we can construct the instrumental variable matrix as $\Phi_{IV}(t)$ and we add this IV scheme within the iteration steps of our proposed method thus requiring no additional steps, and the parameter estimation step is then given by

$$\hat{\theta}_{IV}^{(i)} = \left(\Phi_{IV}(\hat{\theta}_{IV}^{(i-1)})^T \Phi(\hat{\theta}_{IV}^{(i-1)})\right)^{-1} \Phi_{IV}(\hat{\theta}_{IV}^{(i)})^T \; \Psi(\hat{\theta}_{IV}^{(i-1)}) \tag{2.57}$$

where (i) gives the iteration count, and $\Phi_{IV}(\hat{\theta}_{IV}^{(i-1)})$, $\Phi(\hat{\theta}_{IV}^{(i-1)})$ and $\Psi(\hat{\theta}_{IV}^{(i-1)})$ constructs Φ_{IV}, Φ and Ψ respectively for the parameter vector $\hat{\theta}_{IV}^{(i-1)}$.

When α is unknown:

For cases when the commensurate order α needs to be estimated along with other parameters, we can get an estimate of α by posing the problem as a nested loop optimization problem. We start with an initial value of α in the outer loop and in the inner loop we iteratively estimate the model parameters ($\mathbf{a}_n$,$\mathbf{b}_m$) and the delay term (L), as discussed in the previous section. Once convergence is achieved in the inner loop for a fixed α, we update α in the outer loop in a non-linear fashion.

2.3.3 Summary of the proposed algorithm

The iterative procedure for the parameter estimation for both the formulations can be summarized as

STEP 1: OUTER LOOP

Initialization 1: Initialize the algorithm with some initial value for α.

STEP 2: INNER LOOP

Initialization 2: Initialize the inner loop with some initial values for $\hat{\mathbf{a}}_n^{(0)}$ and $\hat{L}^{(0)}$.

1. LS Step: i = 1

   Construct Ψ *and* Φ by replacing $\mathbf{a}_n$ and L with the estimates, as $\hat{\mathbf{a}}_n^{(0)}$ and $\hat{L}^{(0)}$ and get new estimates of the parameters as

   $$\hat{\theta}^{(1)} = (\Phi^T\Phi)^{-1}(\Phi^T\Psi)$$

   Get values of $\hat{\mathbf{a}}_n^{(1)}$,$\hat{\mathbf{b}}_m^{(1)}$ and $\hat{L}^{(1)}$ from $\hat{\theta}^{(1)}$.

2. IV Step: i = i+1 to convergence

   Construct Ψ, Φ *and* Φ_{IV} by replacing $\mathbf{a}_n, \mathbf{b}_m$ and L with estimates as $\hat{\mathbf{a}}_n^{(i-1)}$, $\hat{\mathbf{b}}_m^{(i-1)}$ and $\hat{L}^{(i-1)}$ and get new $\hat{\theta}^{(i)}$ estimates as

   $$\hat{\theta}^{(i)} = (\Phi_{IV}^T\Phi)^{-1}(\Phi_{IV}^T\Psi)$$

   Get the values of $\hat{\mathbf{a}}_n^{(i)}$, $\hat{\mathbf{b}}_m^{(i)}$ and $\hat{L}^{(i)}$ from $\hat{\theta}^{(i)}$ and repeat this step till convergence.

STEP 3: Update value of α based on the minimization of the objective function (i.e repeat steps 1 and 2 till this objective function is minimized).

$$\hat{\alpha} = \arg\min_{\alpha}(\Delta^T\Delta)$$

The algorithm stops when

$$\left\|\hat{\alpha}^{(i)} - \hat{\alpha}^{(i-1)}\right\|_2 < \epsilon$$

the norm of the difference in the parameter vector for two consecutive iterations is less than, ϵ which is chosen as equal to 10^{-4}.

For the cases when α is known, we will only have the inner loop where the model parameters $\mathbf{a}_n$,$\mathbf{b}_m$ and L are estimated iteratively. Usually if an integer order approximation is available for a fractional order transfer function model, the values of parameters from these models are used as initial guesses for this algorithm. Also, since we are only dealing with single input single output models, a finite impulse response for the data set can provide a good initial guess for the time delay. The algorithm for the proposed scheme is sketched in Fig. (2.2). The optimization toolbox in MATLAB is used for solving the outer non-linear loop.

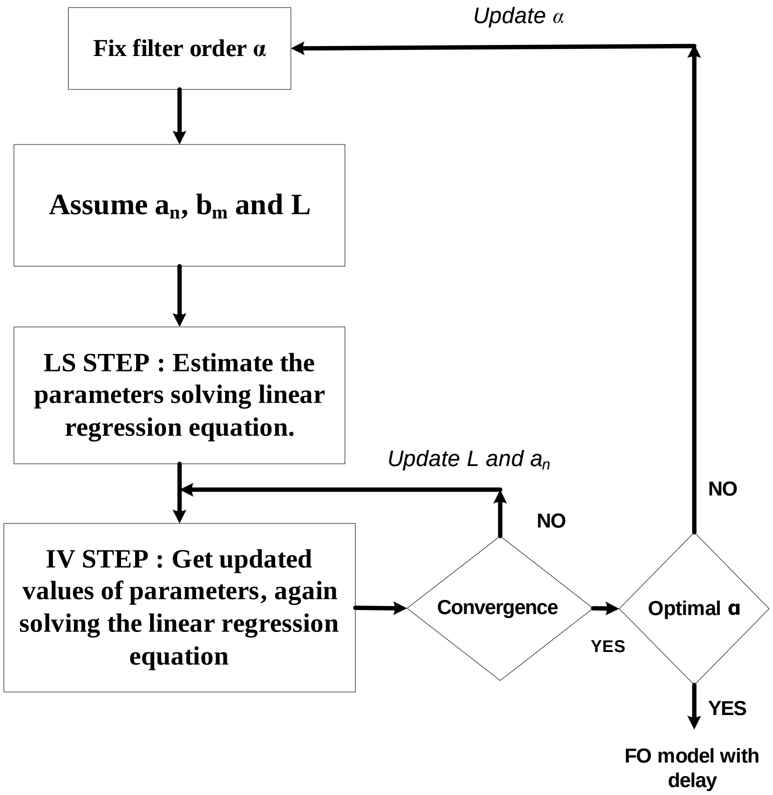

Update α
Fix filter order α
Assume a_n, b_m and L
LS STEP : Estimate the parameters solving linear regression equation.
Update L and a_n
NO
NO
IV STEP : Get updated values of parameters, again solving the linear regression equation
Convergence
YES
Optimal α
YES
FO model with delay

Figure 2.2: Algorithm for estimating parameters for CFOTDS

2.3.4 Convergence issues for the proposed method

The initialization of the inner loop involves choices of $\mathbf{a}_n$,$\mathbf{b}_m$ and L. It is very difficult to prove theoretically the convergence of the proposed algorithm and this is beyond the scope of this work. In practice any initial choice is good except that the filter should not be unstable. As the filter is updated in every step, the final estimate of the parameters is not found to be much sensitive to the initial choice. However, for the outer loop some knowledge on the commensurate fractional order is necessary. This is the limitation of the proposed algorithm that if the outer loop is initialized with a poor initial guess, the convergence of inner loop is not always guaranteed. For the case when the fractional order is known, extensive simulation study shows that the parameter estimates obtained in the inner iterative loop converges to the true parameter values.

2.4 Simulation Examples

To illustrate the utility of the proposed algorithm, the identification exercise is carried out on some simulation examples. For the formulation using step input type excitation, we present the result for the case when all model parameters including commensurate order are unknown. Using the second formulation, we present the results for the following two cases: when the commensurate order is known and when it is unknown. The identification exercise is carried out using two transfer functions of the form given below:

$$G(s) = \frac{b_0}{a_1 s^{\alpha} + 1}\, e^{-Ls} \tag{2.58}$$

$$G(s) = \frac{b_0}{a_2 s^{2\alpha} + a_1 s^{\alpha} + 1}\, e^{-Ls} \tag{2.59}$$

where α is the commensurate fractional order for this model. A zero initial condition is assumed for all cases. The sampled noise free outputs generated from simulations for a given input excitation are corrupted by discrete-time white noise sequences with a signal to noise ratio (SNR) given as

$$\text{SNR} = \frac{\text{var}(signal)}{\text{var}(noise)} \tag{2.60}$$

In presence of noise, Monte Carlo simulation analysis (number of runs is more for less computational expensive case) is done to evaluate the efficacy of the proposed algorithm. The integral in equation (2.50) is evaluated numerically. A fast sampling rate is chosen to reduce the estimation error because of the approximations involved in using fractional operator and a continuous time model in general assumes that a sufficiently fast sampling rate data is used for parameter estimation.

2.4.1 Example 1

For this case, we considered the following fractional order system described in equation (2.61) where we are also estimating the commensurate order α along with a_1,b_0 and L.

$$G_{FO_1}(s) = \frac{1}{s^{0.5}+1}e^{-0.5s} \tag{2.61}$$

Thus, the true parameters are $b_0 = 1$, $a_1 = 1$, $L = 0.5$ and $\alpha = 0.50$. The integer order model approximation for (2.61) is available for this process and is given as

$$G_{IO_1}(s) = \frac{1}{1.5s+1}e^{-0.1s} \tag{2.62}$$

The fractional order term is approximated with Oustaloup's approximation with N = 15 in the frequency interval $[10^{-3}, 10^{3}]$.

Step input excitation

The sampled data is generated by simulating the system using a unit step input with sampling time of 0.1 sec. Fig. 2.3 shows the response of the process to two successive steps. The output $y(t)$ is corrupted with noise having the following values of SNR: ∞(deterministic case), 10 and 5; and for each SNR, for different noise realizations we perform 200 Monte Carlo (MC) simulations. For each case we estimated the fractional order (α) as well as other model parameters (a_1,b_0,L) simultaneously using the proposed nested loop optimization algorithm. Table 2.1 gives the the average ($\bar{\hat{\theta}}$) and the sample standard deviation ($s(\hat{\theta})$) of each parameter for these MC simulations. We started with an initial guess of $\alpha = 0.4$ for all the cases. As can be seen the estimated parameters including the fractional order α are quite close to the true values, thus indicating that the proposed algorithm gives unbiased estimates of all the parameters in the presence of noise and the uncertainty associated with each parameter is more for lower SNR. However, there are some computational issues with the outer non-linear loop, as for some other guess value of α, the inner loop does not always converge. Therefore, having some process knowledge regarding the fractional order α is important.

Generic input excitation

The input excitation is chosen to be a pseudo random binary sequence (PRBS) generated using the 'idinput' function in MATLAB with levels [-1, 1]. As we don't

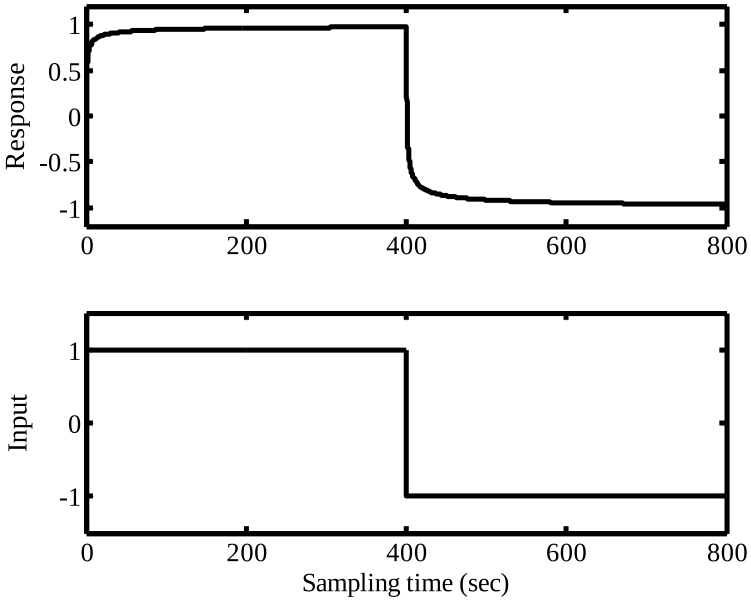


Figure 2.3: Step response for G_{FO_1}

Table 2.1: Step input : Estimated parameters for process $G_{FO_1}(s)$

	$True$	$SNR = \infty$	$SNR = 10$		$SNR = 5$	
			$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
α	0.50	0.500	0.498	0.027	0.492	0.039
a_1	1.00	1.001	1.000	0.025	1.001	0.06
b_0	1.00	1.000	1.003	0.009	1.01	0.026
L	0.50	0.501	0.490	0.023	0.488	0.05

have any rule of thumb available for fractional order processes, here a rule of thumb which is commonly used in many identification techniques is used: the frequency band for PRBS perturbation is used according to the following rule, Frequency Band $\simeq \left[0, \frac{30}{Ts}\right]$, where Ts is the settling time. The frequency band for PRBS excitation is chosen as [0, 0.02] and the sampled data is generated using a sampling time of 0.1 sec. Fig. 2.4 shows the input-output data used for identification for this type of excitation. For this example we performed the identification exercise for these two cases

- When α is known and
- When α is unknown

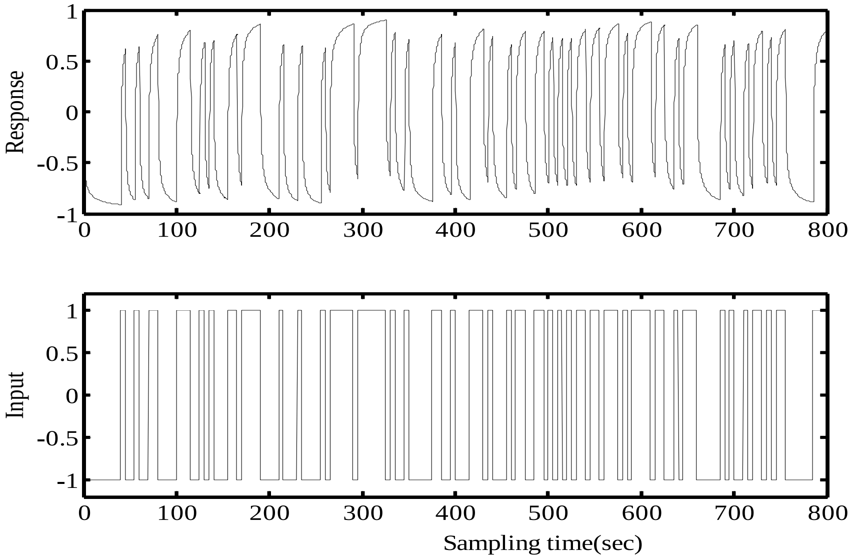


Figure 2.4: Generic input excitation data for G_{FO_1}

The same three values of SNR (∞,10,5) are chosen and for each SNR, Monte Carlo (MC) simulations are performed for different noise realizations. The algorithm is initialized with values of a_1,b_0,L from the integer order model and for the case with unknown α it is initialized with a value of 0.55 and the inner loop with the parameters from the integer order model.

Case 1 - When α is known: For this case we estimated the model parameters (a_1,b_0,L) assuming the commensurate order α is known and equal to 0.5. Table 2.2 gives the average and the sample standard deviation of each parameter for 200 MC simulations. As can be seen, the estimated parameters are quite close to the true values, thus indicating that the proposed algorithm gives unbiased estimates even in the presence of noise.

Table 2.2: PRBS input : Estimated parameters for process $G_{FO_1}(s)$

	$True$	$SNR=\infty$	$SNR=10$		$SNR=5$	
			$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
a_1	1.00	1.001	1.011	0.017	1.007	0.044
b_0	1.00	1.000	1.001	0.006	1.000	0.015
L	0.50	0.500	0.502	0.012	0.490	0.039

Case 2 - When α is unknown: For this case we estimated the fractional order (α) as well as other model parameters (a_1,b_0,L) simultaneously using the proposed nested loop optimization algorithm. Table 2.3 gives the average and the sample standard deviation of each parameter for 100 MC simulations. Some error in estimating fractional order in the outer loop is present because of the approximations involved. Fig. 2.5 presents the ratios of estimated to true parameters along with their confidence intervals (average ± one standard deviation) in a graphical form. As can be seen, the estimated parameters are quite close to the true values, and the scaled confidence intervals include the ratio of one, thus indicating that the proposed algorithm gives unbiased estimates in the presence of noise.

Table 2.3: PRBS input : Estimated parameters for process $G_{FO_1}(s)$

	$True$	$SNR = \infty$	$SNR = 10$		$SNR = 5$	
			$\bar{\hat{\theta}}$	$s(\hat{\theta})$	$\bar{\hat{\theta}}$	$s(\hat{\theta})$
α	0.50	0.499	0.502	0.019	0.509	0.036
a_1	1.00	1.000	1.01	0.013	1.018	0.029
b_0	1.00	1.001	1.003	0.009	0.998	0.018
L	0.50	0.499	0.506	0.01	0.515	0.029

Fig. 2.6 presents the Bode plots for parameter estimates from all the 200 realizations. As can be seen the 200 models fit the Bode diagram of the simulated system (2.61) really well.

2.4.2 Example 2

The process with the following transfer function is used:

$$G_{FO_2}(s) = \frac{1}{8s^{2\times 0.75} + 5s^{0.75} + 1} e^{-4.8s} \tag{2.63}$$

Thus, the true parameters are $b_0 = 1$, $a_1 = 5$, $a_2 = 8$, $L = 4.8$ and $\alpha = 0.75$. The input excitation is chosen to be a pseudo random binary sequence (PRBS) generated using the ‘idinput’ function in MATLAB with levels [-1, 1]. The frequency band for PRBS excitation is chosen as [0, 0.015] and the sampled data is generated using a sampling time of 0.1 sec. Fig. 2.7 shows the input-output data used for identification for this type of excitation.

Table 2.4 gives the average and the sample standard deviation of each parameter for 100 MC simulations. As can be seen, the average ± one standard deviation of

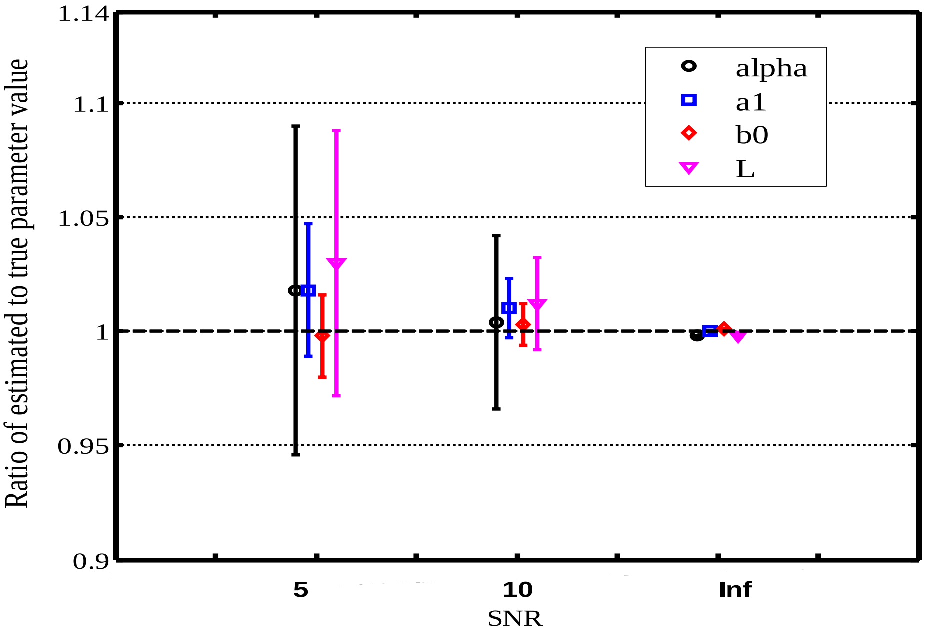


Figure 2.5: Effect of noise level on parameter estimates

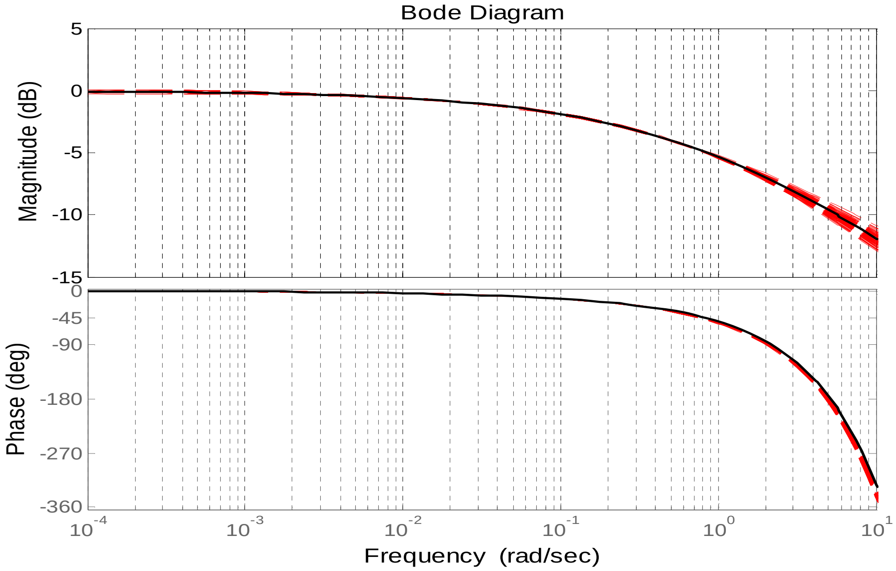


Figure 2.6: Bode plots for 200 MC simulation runs : (-) true process, (- -) 200 FO models

the parameters include the true value of parameters. The algorithm is initialized with the value of α as 0.5 for all cases.

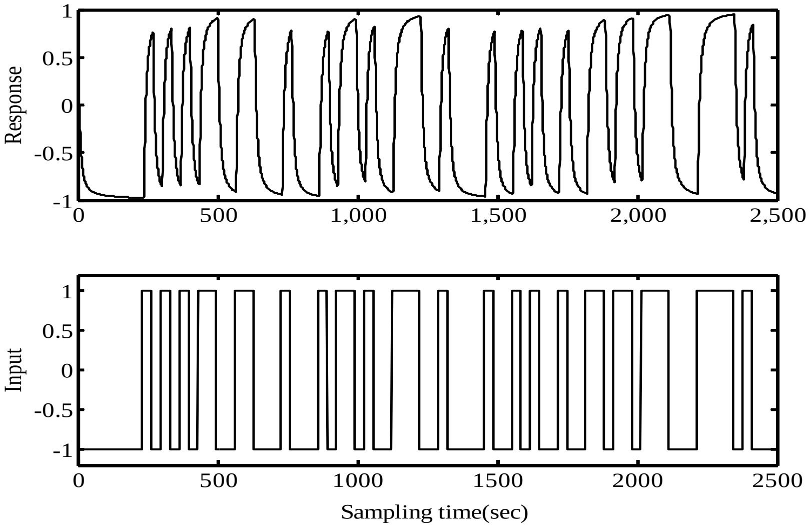


Figure 2.7: Generic input excitation data for G_{FO_2}

Table 2.4: PRBS input : Estimated parameters for process $G_{FO_2}(s)$

	$True$	$SNR = \infty$	$SNR = 20$		$SNR = 10$	
			$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
α	0.75	0.750	0.744	0.022	0.740	0.025
a_1	5.00	5.010	4.896	0.254	4.821	0.511
a_2	8.00	8.004	8.149	0.331	8.110	0.415
b_0	1.00	0.999	1.008	0.002	1.000	0.031
L	4.80	4.793	4.701	0.108	4.689	0.151

2.5 Application to Thermal diffusion in a wall

For many real processes, fractional differentiation appears naturally when the system transients are governed by a diffusion equation, and particularly between the variables governing the functioning of the interface. Benchellal et al. (2006) has shown that the transfer function H(s) relating heat flux and the temperature, on the front face of the heated wall (which is governed by classical heat conduction equation), is a fractional order transfer function with half integer order. This classical wall (CW) problem is considered as a process for this simulation study to illustrate the importance of our proposed algorithm on a real physical system. Fig.2.8 represents the classical wall problem used to analyze heat transfer. The governing equation relating heat flux, $\Omega(x,t)$ and the temperature, $T(x,t)$ for this

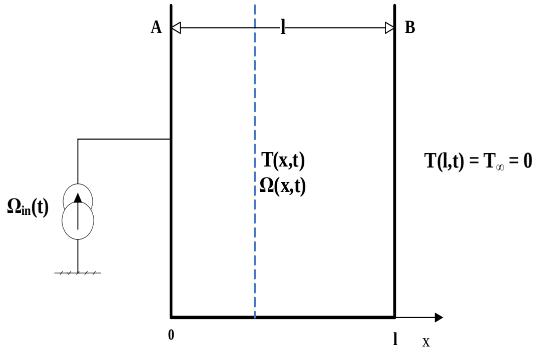


Figure 2.8: Classical wall problem

process as:

$$\frac{\partial T(x,t)}{\partial t} = \zeta \frac{\partial^2 T(x,t)}{\partial x^2} \tag{2.64}$$

$$\Omega(x,t) = -\varrho \frac{\partial T(x,t)}{\partial x} \tag{2.65}$$

where ζ is the thermal diffusivity $(=\frac{\varrho}{\rho c})$, ϱ is the thermal conductivity, ρ is the mass density, and c is the specific heat. The boundary conditions are such that the temperature at face B is kept constant and equal to zero during the overall heating experiment and the external heat is added at face A, that is

$$T(l,t) = 0 \tag{2.66}$$

$$\Omega(0,t) = \Omega_{in}(t) \tag{2.67}$$

where 'l' is the distance between two walls.

2.5.1 Transfer function for the wall problem

If S_A is the cross-sectional area of the wall and we define $y(t) = T(x^*,t)$ and heat input (not flux) as, $u(t) = \Omega_{in}(t)S_A$, then an analytical expression for the transfer function, $H(s)$ relating $Y(s)$ and $U(s)$ can be formulated as

$$H(s) = \frac{Y(s)}{U(s)} = \frac{1}{S_A \varrho \sqrt{\frac{s}{\zeta}}} \frac{\exp\left((2l - x^*)\sqrt{\frac{s}{\zeta}}\right) - \exp\left(x^*\sqrt{\frac{s}{\zeta}}\right)}{1 + \exp\left(2l\sqrt{\frac{s}{\zeta}}\right)} \tag{2.68}$$

The detailed derivation of this is presented in chapter-4. Once the exponential series expansion is done, the transfer function relating temperature and heat input is a commensurate fractional order model with commensurate order 0.5. Thus, it would require an infinite number of terms in both the numerator and denominator to model this process accurately, and working with a reduced model structure will always result in some modeling errors.

Based on the first principle model, a time delay or dead time will not appear in a process transfer function until and unless there is mass or energy flow. Since this process involves energy flow, and depending on the location of the process measurement device (which is represented by x^* here), a dead time may appear in the process. Malti et al. (2009) noticed the time-lag in flux diffusion while modeling thermal rod process from experimental data. Thus, apparent time delay may be present in this process. Using this fact into consideration, and the fact that the process dynamics involve non-integer behavior, it is assumed that this system can be approximated by fractional order model with a time delay using a fewer number of parameters. So, we are trying to model this process as a parsimonious in parameter model using fractional order dynamic model with a delay term.

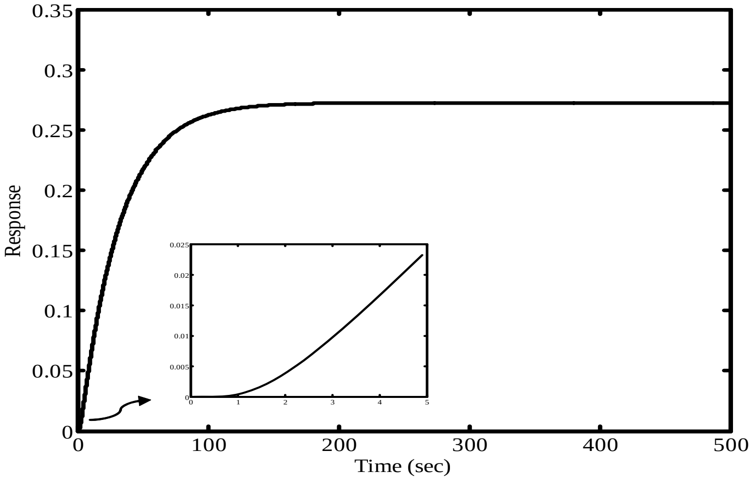


Figure 2.9: Step response of the process

Sampled data is generated by performing numerical simulations using the finite difference method. The following properties of Brass are used for simulations: ρ = 8.522×10^3 kg/ m^3, c = 385 J/kg ^{0}C, λ = 111 W/m ^{0}C. The distance between the walls, l is chosen as 5 cm and surface area of the wall(S_A) is 100 cm^2. Temperature is measured at a distance of 2 cm from the front face of the wall (x^*). 300 discretization points are chosen. For a step input of 10KW in heat input, if we see the zoomed in Fig. 2.9 for the initial time, it shows why we are trying to model this as fractional order model with delay.

2.5.2 Identification results

The sampled data (15000 data points) for the identification exercise is generated using sampling time of 0.1 sec with PRBS type input excitation with levels of [-0.1, 0.1] KW in the frequency band of [0, 0.02]. A Gaussian white noise signal with SNR = 20 is added into the simulated noise free output sequence. This process is known to exhibit fractional order dynamics for frequencies less than 10^3 rad/sec (Benchellal et al., 2006). Here, we use the Oustaloup approximation with N = 15 in the frequency interval $[10^{-3}, 10^3]$ to approximate the fractional differential operator. The overall data is partitioned into two parts : (a) an identification set: First 8000 data points and (b) the validation data set: next 7000 data points. The identification data set is shown in Fig. 2.10. Next, the proposed algorithm is used to fit a fractional order model with time delay to the identification data. The model structure of the commensurate model (4.1) is varied to find a model which gives the best predictions. The estimated continuous time FO model (using average parameter value) along with sampled standard deviation for all parameters

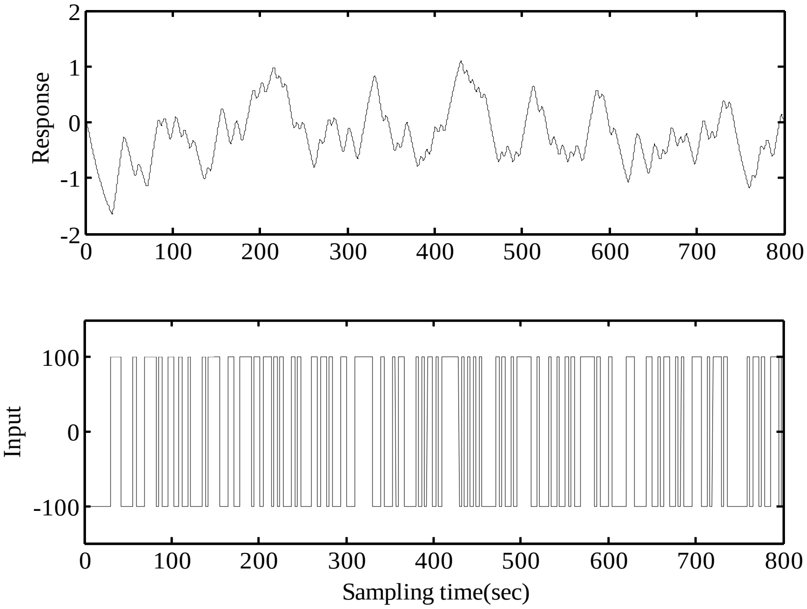


Figure 2.10: Time response for the wall process

using 50 Monte Carlo simulations is given as (4.42)

$$G_{FO_3}(s) = \frac{0.0273(\pm 0.003)}{46.239(\pm 6.211)s^{2\times(0.585\pm 0.035)} + 2.263(\pm 1.265)s^{0.585(\pm 0.035)} + 1} e^{-0.899(\pm 0.058)s} \tag{2.69}$$

It is not possible to show the predictions from all the 50 models, so only predictions from the average model has been presented in Fig. 2.11. It shows the model predictions of $G_{FO_3}(s)$ (infinite step ahead predictions) and the process output for all 50 realizations on the validation data set. As can be seen, FO model ($G_{FO_3}(s)$) fits the measured output quite well. Next we compare the step response (for a step change of 10 KW in heat input) of all the 50 estimated models, to the step response from the actual process as shown in Fig. 2.12. As can be seen from Fig. 2.12, $G_{FO_3}(s)$ estimates both gain and delay very accurately. The Bode plot for the model G_{FO} along with the actual frequency response of the process is shown in Fig. 2.13. It can be seen that the frequency response of the model $G_{FO_3}(s)$ is nearly the same as the true process at low frequencies, however at high frequencies the delay term in the model starts to dominate and there is mismatch between the true and model behavior. The frequency response plot indicates that our proposed modeling scheme is able to capture the deterministic part of the process quite well at low as well as moderate frequency regions. Thus, the proposed algorithm can be used to model the low frequency behavior of this process. This process which

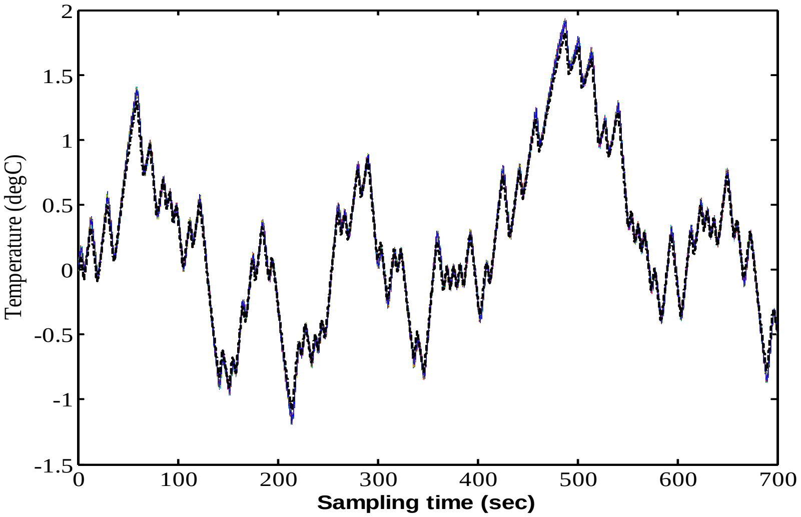


Figure 2.11: Model predictions on the validation set: (- -) black dotted line is model prediction)

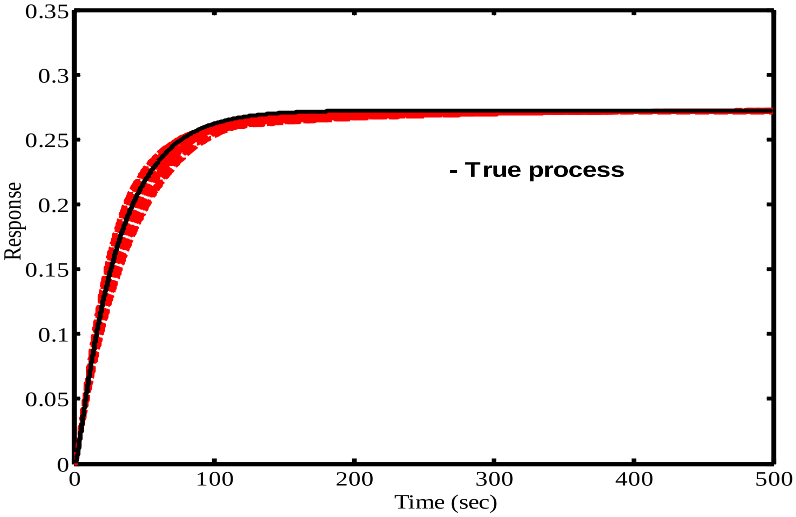


Figure 2.12: Step response of the estimated models(- -) and true process (-)

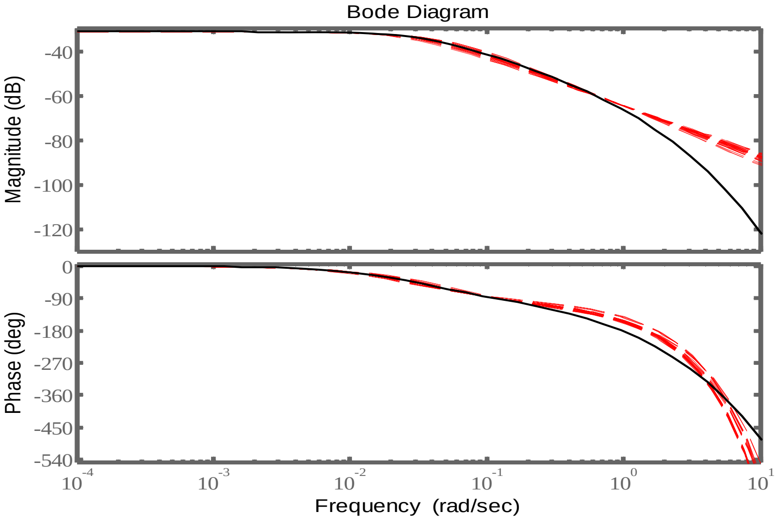


Figure 2.13: Frequency responses of the estimated models (- -) and the true process (-)

is described by half integer order model behavior is discussed here to emphasize the importance of the developed algorithm to model fractional order processes without requiring an integer order approximation of the process. The developed fractional order model can then further be used to design rational or fractional order controllers.

2.6 Conclusion

A continuous-time identification method for commensurate fractional order models with time delay is developed in this chapter. The proposed method works with any kind of input signal excitation. It is based on a linear filter method where the filter is chosen as a combination of RIVC and a linear integral filter. Using this kind of filter, we can make the delay term appear as explicit parameter similar to other constant model parameters and can form a linear regression model to estimate the parameters in an iterative manner. For the case when the commensurate order α is unknown, a nested loop optimization method is developed to estimate the time delay along with constant model parameters in an iterative way in the inner loop and the fractional order in the outer loop. The applicability of the developed procedure is demonstrated on a CFOTDS for the cases when α is known and when it is unknown. In the presence of noise, Monte Carlo simulation analysis for different noise realizations has been carried out to demonstrate that the proposed algorithm gives unbiased estimates even in the presence of noise. The proposed algorithm is also applied on a fractional order system of classical wall heat transfer problem which is described by fractional behavior. Future work proposed is to extend this algorithm for non-commensurate models.

Bibliography

S. Ahmed, B. Huang and S.L. Shah. Novel identification method from step response. *Control Engineering Practice*, vol. 15, pp. 545-556, 2007.

M. Aoun, R. Malti, F. Levron and A. Oustaloup. Synthesis of fractional Laguerre basis for system approximation. *Automatica*, vol. 43, pp. 1640-1648, 2007.

R.L. Bagley and P. Torvik. On the appearance of the fractional derivative in the behavior of real materials. *J. Appl. Mech.*, vol. 51 (1), pp. 294-298, 1984.

Q. Bi, W.J. Cai, E.L. Lee, Q.G Wang, C.C. Hang and Y. Zhang. Robust identification of first-order plus dead-time model from step response. *Control Engineering Practice*, vol. 7, pp. 71-77, 1999.

A. Benchellal, S. Bachir, T. Poinot and J. C. Trigeassou. Identification of a non-integer model of induction machines *IFAC Workshop on Fractional Differentiation and its Applications (FDA'04)*, Bordeaux, France, pp. 400-407, 2004.

A. Benchellal, T. Poinot and C. Trigeassou Approximation and identification of diffusive interfaces by fractional systems. *Signal Processing*, vol. 86 (10), pp. 2712-2727, 2006.

Y.Q. Chen and K.L. Moore. Discretization schemes for fractional-Order differentiators and integrators. *IEEE Transactions on Circuits and Systems - I : Fundamental Theory and Applications*, vol. 49 (3), pp. 363-367, 2002.

O. Cois. Systèmes linéaires non entiers et identification par modèle non entier: application en thermique. *Ph.D. thesis, Université Bordeaux*, Talence, France, 2002.

O. Cois, A. Oustaloup, T. Poinot and J.L. Battaglia. Fractional state variable filter for system identification by fractional model. *IEEE Sixth European Control Conference (ECC'2001)*, Portugal, 2001.

J.D. Gabano and T. Poinot. Fractional identification algorithms applied to thermal parameter estimation *15th IFAC Symposium on System Identification*, Saint-Malo, France, pp. 1316-1321, 2009

V.G. Jenson and G.V. Jeffreys. Mathematical Methods in Chemical Engineering. *2nd ed. New York: Academic*, New York, 1977.

L. Le Lay. Identifiction fréquentielle et temporelle par modèle non entier, Ph.D. thesis. *Université Bordeaux*, Talence, France, 1998.

J. Lin. Modélisation et identification de systèmes d'ordre non entier, Ph.D. thesis. *Université de Poitiers*, Poitiers, France 2001.

S. Manabe. Early development of fractional order control. *Proc. of ASME 2003 Design Eng. Tech. Conf.*, Chicago, 2003.

R. Malti, S. Victor, A. Oustaloup and H. Garnier. An optimal instrumental variable method for continuous-time fractional model identification. *Proceedings of the 17th IFAC World Congress*, Korea, pp. 14379-14384, 2008a.

R. Malti, S. Victor and A. Oustaloup. Advances in system identification using fractional models. *Journal of Computational and Nonlinear Dynamics*, vol. 3(2), 2008b.

R. Malti, J. Sabatier and H.Akçay. Thermal modeling and identification of an aluminum rod using fractional calculus. *15th IFAC Symposium on System Identification*, Saint-Malo, France, pp. 958-963, 2009.

D. Matignon. Stability properties for generalized fractional differential systems. *ESAIM Proceedings Systèmes Différentiels Fractionnaires Modèles, Méthodes et Applications*, vol. 5, pp. 145-158, 1998.

M. Muddu, A. Narang and Sachin C. Patwardhan. Development of ARX models for predictive control using fractional order and orthonormal basis filter parametrization. *Ind. Eng. Chem. Res.*, vol. 48(19), pp. 8966-8979, 2009.

M. Muddu, A. Narang and Sachin C. Patwardhan. Reparametrized ARX models for predictive control of staged and packed bed distillation columns. *Control Engineering Practice*, vol. 18(2), pp. 114-130, 2010.

A. Narang, Sirish L. Shah and Tongwen Chen. Continuous time model identification of fractional order models with time delays. *15th IFAC Symposium on System Identification*, Saint-Malo, France, 2009.

K.B. Oldham and J. Spanier. The Fractional Calculus. *Academic Press*, San Diego 1974.

M. Ortigueira and J. Tenreiro. Signal Processing Special Issue: Fractional Calculus Applications in Signals and Systems. *Signal Processing*, vol. 86, 2006.

A. Oustaloup. La Dérivation Non Entière. *Hermés*, Paris, 1995.

I. Podlubny. Fractional Differential Equations. *Academic Press*, San Diego 1999.

D. Seborg, T. Edgar and D. Mellichamp. Process Dynamics and Control. *John Wiley & Sons*, New York, 1989.

J.Sabatier, M. Aoun, A. Oustaloup, G. Grégoire, F. Ragot, and P. Roy. Fractional system identification for lead acid battery state of charge estimation. *Signal Processing*, vol. 86, pp. 26452657, 2006.

S. Victor, R. Malti and A. Oustaloup. Instrumental variable method with optimal fractional differentiation order for continuous-time system identification. *15th IFAC Symposium on System Identification*, Saint-Malo, France, 2009.

B.M. Vinagre,I. Podlubny, A. Hernandez and V. Feliu. Some approximations of fractional order operators used in control theory and applications. *41st IEEE Conference on Decision and Control*, 2002.

Q.G. Wang and Y. Zhang. Robust identification of continuous systems with dead-time from step responses. *Automatica*, vol. 37, pp. 377-390, 2001.

P.C. Young. An instrumental variable method for real time identification of noisy process. *Automatica*, vol. 6, pp. 271-287, 1970.

Chapter 3

Tuning of fractional PI controller for processes with and without time delays using particle swarm optimization [1]

In this chapter, a servo control strategy for tuning of fractional order PI (FO-PI or PI$^\lambda$) controllers is proposed for fractional order and integer order systems. The proposed strategy is based on a reference model, whose open-loop transfer function is chosen to be Bode's ideal transfer function. Bode's ideal transfer function is a fractional order system and provides an infinite gain margin and a constant phase margin. To come up with satisfactory tuning parameters of the controller, an iterative optimization method using particle swarm optimization (PSO) has been used, based on minimization of a quadratic cost function with constraints. This cost function is defined as weighted sum of squares of control input moves and sum of square of error between the time response of the reference model and the closed loop system with the FO-PI controller. The resulting closed-loop system is shown to exhibit features of robustness to process gain variations and the step responses exhibit iso-damping property as would be expected from a constant phase margin closed loop process. The applicability of the proposed algorithm is demonstrated by application to simulation examples as well as on a computer-interfaced pilot scale continuous stirred tank heater (CSTH) process.

[1] The full version of this chapter has been submitted to Journal of Process Control for possible publication. A condensed version of this chapter was presented at the 2010 American Control Conference, ACC2010, June 2010, USA (Narang et al., 2010). The first part of this chapter was presented as a poster presentation at ICORE Summit 2009, Banff, Canada.

3.1 Introduction

Recently, there has been an increasing amount of interest in applying fractional calculus to model and control physical systems. Several applications of fractional calculus can be found in Axtell and Bise (1990); Shantanu (2008); Monje et al. (2010). In the area of automatic control systems, the application of the FC can be found in Podlubny (1999*b*). ASME, IEEE and IFAC had organized International symposiums and workshops over the last couple of years to promote and exchange interest in the fractional calculus and its applications.

PID controllers are the most commonly used control algorithms in industry. Since Ziegler Nichols (Ziegler and Nichols, 1942) proposed their tuning method for designing PID controllers there has been significant development in the area of tuning and designing PID controllers. So, a search for new algorithms for better design of these controllers has never been an unreasonable demand for the research community. Bode (Bode, 1945) while working on a problem to design feedback amplifiers mentioned the use of fractional integro differential operator in a feedback loop. The purpose was to obtain the performance from the feedback loop such that it is invariant to changes in the amplifier gain. Bode presented an elegant solution to this robust design problem, which he called the Ideal cutoff characteristic, nowadays known as ideal loop transfer function, whose Nyquist plot is a straight line through the origin giving a phase margin invariant to gain changes. However, the idea of fractional-order algorithms for the control of dynamic systems was first introduced by Oustaloup (1995) where he also demonstrated the superior performance of the CRONE (French abbreviation for Commande Robuste d'Ordre Non Entier) method over the classical PID controller. Later, Podlubny (1999*a*) proposed a generalization of the PID controller, namely the fractional order PID (FO-PID or $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$) controller, involving an integrator of order λ and differentiator of order μ (the orders λ and μ may assume real non-integer values). The author also demonstrated better performance of this type of controllers, in comparison with the classical PID controllers, when used for the control of fractional-order systems.

In the last decade many tuning rules have been proposed for designing FO-PID controllers. Some of these techniques are based on an extension of the classical PID control theory. Several analytical ways to tune such controllers have been proposed in Caponetto et al. (2002, 2004). The recent book by Monje et al. (2010) describes fundamentals, applications and challenges in fractional order modeling and control design in sufficient detail. The tuning of fractional order controllers in frequency domain has been studied by Vinagre et al. (2002). Monje et al. (2004*a*) proposed the technique to tune fractional PID controllers by requiring the closed loop system to satisfy certain conditions on the phase margin, gain crossover frequency and sensitivity functions. The tuning parameters for the FO-PID is then obtained by solving the linear numerical optimization problem. In another paper by Monje et al. (2004*b*), the authors proposed a scheme to tune FO-PI controllers in order to fulfill three different robustness design specifications for the compensated system; an optimization method based on a nonlinear minimization function subject to nonlinear constraints is used to tune the controller. Chen et al. (2008) proposed a generalized MIGO (M_s constrained integral gain optimization) based controller tuning method called F-MIGO, to handle the FO-PI case when the fractional integral order for the controller is known. Also, Valério and Costa (2006) proposed two sets of tuning rules for FO-PID with the proposed rules bearing similarities to the rules proposed by Ziegler and Nichols for integer PID controllers, and made use of the plant time response data. Leu et al. (2002) designed an optimal fractional order PID controller based on specified gain and phase margins with a minimum integral squared error criterion. Luo et al. (2009) proposed FO-PI tuning rule for a class of fractional order systems. Cao and Cao (2006) designed fractional order controller by minimizing weighted combination of ITAE and control input using particle swarm optimization tool. Barbosa et al. (2004) proposed algorithm to tune classical PID controllers using Bode's Ideal transfer function. Recently, Padula and Visioli (2011) proposed a tuning rule to optimally design PID and fractional order PID controllers for first order plus time delay models. The experimental validation of some of these techniques include work by Luo et al. (2010, 2011). Most of these proposed tuning algorithms have been applied to many physical processes. A fractional-order control strategy known as fractional sliding mode control has also been successfully applied in the control of a power electronic buck converter (Calderon et al., 2006). Li et al. (2010) proposed a FO-PD motion controller for dc motor position servo system. Delavari et al. (2010) applied a fractional order sliding mode control to a level control in a nonlinear coupled tank. Robust control of irrigation canals has been applied by Feliu et al. (2009) and fractional control of thermal systems has been applied by Sabatier and Oustaloup (2002); Vinagre et al. (2001); PetrPetráš and Vinagre (2002). Monje et al. (2008) paper summarizes many of the recent advances and applications for tuning of fractional order controllers.

In this chapter we propose a tuning strategy to design fractional order controllers, in particular, fractional PI controllers for processes to enhance robustness and closed loop system performance. It is intuitively true, as also argued in Podlubny (1999*b*), that the fractional order models require much more than classical PID controllers to achieve good closed loop performance. A tuning strategy for fractional order PI controllers is proposed for fractional order as well a classical integer order system models. This chapter extends the authors previous work (Narang et al., 2010). The proposed strategy is based on a reference model, whose open-loop transfer function is given by Bode's ideal transfer function. To obtain the tuning parameters of the controller, an iterative optimization method is used, based on minimization of a quadratic cost function. This cost function is defined as weighted sum of squares of control input moves and sum of the square of error between the time responses of the reference model and the fractional system with the FO-PI controller. Thus, we want to match the closed loop response to that of a reference system and also penalize the control input moves. This work extends the tuning strategy proposed by Barbosa et al. (2004) where the authors designed classical integer order PID controllers for integer order models. The resultant closed-loop systems (with the FO-PI controllers) exhibit the features of robustness to gain variations and the step responses exhibit the iso-damping property. The proposed algorithm is also extended to handle time delays in the system.

This chapter is organized as follows. Section 3.2 presents a brief theory of fractional calculus with an introduction to fractional order PI controllers. The proposed tuning strategy for designing FO-PI controllers based on Bode's ideal transfer function is presented in Section 3.3. To study the efficacy of the proposed strategy developed in Section 3.3, some examples of fractional order system models are studied in Section 3.4 to illustrate its applicability. Section 3.5 presents simulation study on lag/delay dominant processes. Section 3.6 presents simulation study on the thermal diffusion in a wall. Section 3.7 presents the experimental results for the designed FO-PI controller on continuous stirred tank heater (CSTH) setup. Concluding remarks appear in Section 3.8.

3.2 Fractional order PI controller

In Chapter 2 we introduced the generic transfer function form of fractional order systems (see equation 2.5). The same type of models are used in this chapter for designing fractional order PI controllers.

A fractional-order PI$^{\lambda}$D$^{\mu}$ controller is considered as the generalization of the conventional PID controllers involving an integrator of order λ and a differentiator of order μ, where both λ and μ can assume non-integer values. The structure of a PI$^\lambda$D$^\mu$ controller with the transfer function, $C(s)$ is given as

$$C(s) = K_c + \frac{K_i}{s^\lambda} + K_d s^\mu \tag{3.1}$$

where K_c, K_i and K_d are the proportional gain, integral gain and derivative gain, respectively, of the fractional order controller. All classical types of PID controllers are special cases of such FO-PID controllers. The main advantage of using a fractional-order PID controller for a linear control system is that we have two additional degrees of freedom in the controller design. Using these additional parameters of the integral and differential orders, it is expected that the use of FO-PID controllers can enhance the feedback control loop performance compared to integer-order controllers, however it also means that the tuning of the controller can be much more complex. In this work, we study the problem of designing a fractional order proportional-integral controller (FO-PI) ($\mu = 0$) of the form,

$$C(s) = K_c + \frac{K_i}{s^\lambda} \tag{3.2}$$

as a result now we have three controller tuning parameters compared to two, for classical PI controllers. The role of $1/s^\lambda$ is to provide integral action to ensure there is no offset in the closed loop system.

For digital implementation of the fractional order operator, the key step is numerical evaluation or discretization of the operator. Again, we will be using Oustaloup continuous approximation (equation 2.8) for the simulation of fractional order controllers as well as the closed loop systems. The bigger N, the better the approximation of the differentiator s^λ in its frequency band: low values result in simpler approximations, but also cause the appearance of a ripple in both gain and phase behaviors and ripples may be eliminated by increasing N, but the approximation will be computationally heavier. In order to implement the controller in real practice, discretization has to be done and sampling is an issue; however, it is not explored in this work. Note that the proposed method discussed in the next section is independent of the way in which fractional differentiation and integration are simulated in the time domain. Also, the focus of this work is not on which approximation is better but rather on developing an algorithm for tuning fractional order PI controllers.

3.3 FO-PI tuning formulation

3.3.1 Bode's ideal transfer function and design of a FO-PI controller

Bode in his study on design of feedback amplifiers (Bode, 1945), suggested an ideal shape of the open-loop transfer function of the form:

$$G_{ref}(s) \;=\; \frac{K_r}{s^{\gamma}} \quad (1 < \; \gamma < \; 2) \tag{3.3}$$

The purpose was to obtain the performance from the feedback loop such that it is invariant to changes in amplifier gain. This open loop transfer function with gain K_r and fractional order γ shows very interesting properties as listed in Table 3.1.

Table 3.1: Properties of open loop Bode's ideal transfer function

Property	**Relation**
Magnitude curve	Constant slope of -20γ dB/dec
Gain cross-over frequency, ω_c	$(K_r)^{\frac{1}{\gamma}}$
Phase angle curve	Horizontal line at $-\frac{\gamma\pi}{2}$
Nyquist curve	Straight line at argument $-\frac{\gamma\pi}{2}$

The Nyquist plot for Bode's ideal transfer function is shown in Fig. 3.1. As can be seen from the figure that this open loop transfer function gives a constant phase plot at all frequencies. Thus, this closed loop system is robust to process gain variations.

Closed loop TF:

$$G_{refCL}(s) = \frac{1}{(\frac{1}{K_r})s^{\gamma} + 1} \tag{3.4}$$

If we consider a feedback system with Bode's ideal transfer function inserted in the forward path, then based on the frequency domain analysis, this closed loop system exhibits important properties such as infinite gain margin and constant phase margin (dependent only on γ). These properties are listed in Table 3.2. Also, the magnitude of overshoot for the closed loop system is independent of process gain and thus, the step response of the closed loop system exhibits the iso-damping property. As for this reference system, the order γ and the gain K_r establishes the overshoot and the speed of the output response, respectively, as can be seen from the Fig. 3.2. Therefore, these two parameters are used to decide the desired characteristics of the closed loop response.

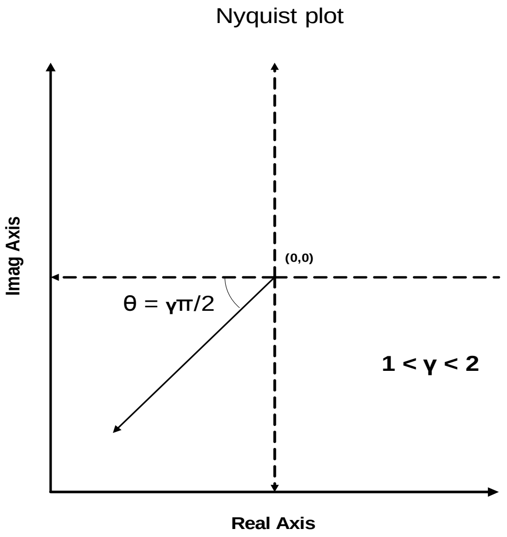


Figure 3.1: Nyquist plot for Bode's ideal transfer function

Table 3.2: Properties of feedback system with Bode's ideal transfer function

Property	**Relation**
Gain margin	∞
Phase margin	$\pi(1 - \frac{\gamma}{2})$
Overshoot	$\simeq 0.8(\gamma - 1)(\gamma - 0.75)$

The motivation behind exploring Bode's ideal transfer function is to exploit these useful properties of Bode's ideal transfer function to tune a fractional order controller which makes the closed loop system robust to process gain variation and the closed loop step response shows an iso-damping property. Consider if $C(s)$ is a controller transfer function and $G(s)$ is the plant transfer function; then to utilize these important properties of Bode's ideal transfer function we would like to have the open loop transfer function ($G_{OL}(s) = G(s)C(s)$) close to Bode's ideal transfer function G_{ref}. Doing so will ensure that the closed loop response of this system will also behave like the closed loop response of the reference system giving us the important properties of the reference system. The idea behind using Bode's ideal transfer function can be demonstrated with the following examples. If $G_{OL}(s)$ is close to G_{ref} then

$$G(s)C(s) \sim G_{ref}(s) \tag{3.5}$$

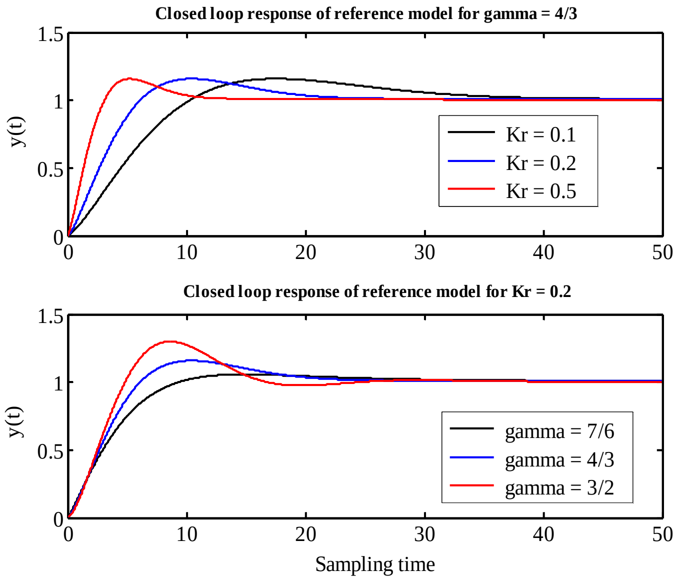


Figure 3.2: Step response for closed loop reference model

For a system with the transfer function model

$$G(s) = \frac{K}{\tau s + 1} \tag{3.6}$$

then

$$C(s) \sim \frac{K_r}{K} \left(\frac{\tau}{s^{\gamma-1}} + \frac{1}{s^{\gamma}} \right) \ \forall \ \gamma > 1 \tag{3.7}$$

which is a FO-PI controller with two fractional integrators. Now, for the system represented by the transfer function model

$$G(s) = \frac{K}{a_2.s^{\beta} + a_1.s^{\alpha} + a_0} \tag{3.8}$$

then

$$C(s) \sim \frac{K_r}{K} \left(\frac{a_2}{s^{\gamma-\beta}} + \frac{a_1}{s^{\gamma-\alpha}} + \frac{a_0}{s^{\gamma}} \right), \ \forall \ \ \gamma > \beta > \alpha \tag{3.9}$$

which is again a FO-PI controller with three fractional integrators. The purpose of demonstrating these examples is that we will always get a fractional order controller (assuming no plant model mismatch) if we want the open loop transfer function close to open loop Bode's ideal transfer function. However, the controller structure changes for different process models. To overcome this disadvantage, in our study we looked at designing a fractional order PI controller such that we match the closed loop response with the reference system or

$$G_{CL} = \frac{G(s)C(s)}{1 + G(s)C(s)} \sim G_{refCL}(s) \tag{3.10}$$

The most common control structure often found in the process industry is the PI type. It is a common practice in industry to turn off the derivative gain. We are interested in a tuning strategy in a set point tracking scenario, so we considered FO-PI controller for this study. For our case if we fix the structure of the controller as the FO-PI controller with the transfer function given by

$$C(s) = K_c + \frac{K_i}{s^\lambda} \tag{3.11}$$

then we are interested in finding controller parameters such that we get a closed loop response which resembles the closed loop response of the reference system. Thus, fixing γ and K_r fixes the gain and phase margins for the reference system and if the closed loop system (with FO-PI controller) closely resembles the reference system, then it also implicitly implies fixing the phase and gain margins of the required closed loop system.

Here we extend the tuning strategy proposed by Barbosa et al. (2004) for tuning classical integer order PID controllers to designing FO-PI controllers for fractional as well as integer order models. We are interested in a tuning strategy for a set point tracking scenario. For the step change in the set point, the tuning parameters of the FO-PI controller (K_c, K_i, λ) are obtained by minimizing a quadratic cost function. This cost function is defined as weighted sum of squares of control input moves and sum of square of error between the time responses of the reference model and the process with the FO-PI controller. An iterative non-linear optimization algorithm is used to simultaneously estimate all parameters of the FO-PI controller. The feedback loop of Fig. 3.3 shows the details of this proposed scheme. The first step involves fixing the parameters of the reference model. The controller design is an iterative process, so the next step is the initialization of FO-PI controller parameters. This initialization can be done by designing classical PI controller using any standard tuning rule. Then, the step response of the reference model (Eq. (3.4)) is computed along with the closed-loop response and controller output (with FO-PI controller) for the process model. These values are used to compute the quadratic cost function and gradient based optimization technique can be used to minimize this cost function. Alternatively, we can define the search space for the tuning parameters and use global optimization tool like particle swarm optimization (PSO) to solve for the controller parameters and can even use this in combination with the optimization toolbox in MATLAB. The controller parameters that minimize this cost function give the required FO-PI controller settings. The algorithm has also been discussed in detail in Section 3.3 while the some details on PSO is discussed in Section 3.4.

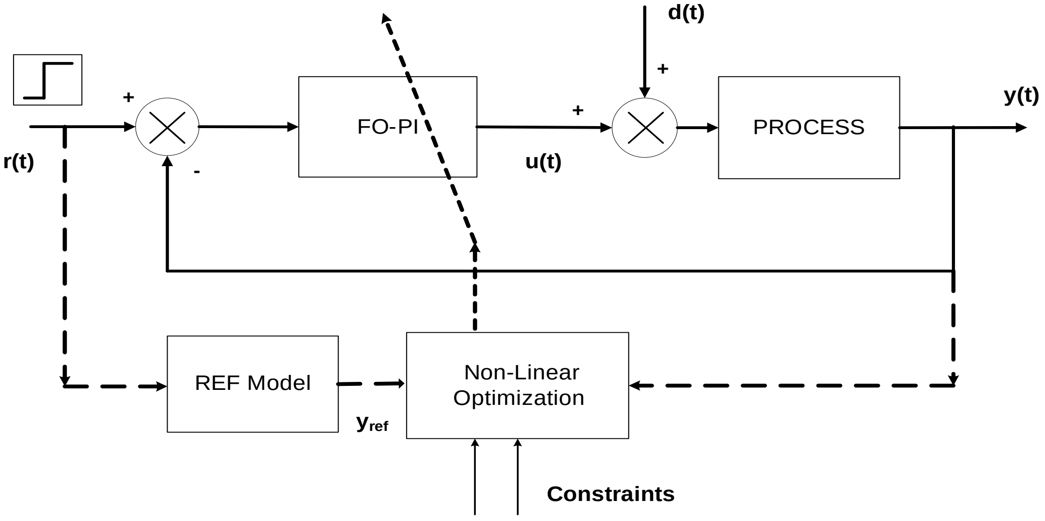


Figure 3.3: Structure for FO-PI controller tuning

Time delays as transportation lags or apparent time delays may be present in a process due to actuator limitations or process measurements. Presence of a time delay in a process limits the performance of a conventional feedback control system. We need to modify the control structure to tune the controller for models with time delays. We can either change the structure of the reference model or use smith predictor technique to overcome time delays that characterizes these systems. Thus, the two methods are:

1. Based on the modified reference model

2. Using the Smith predictor

If the time delays are known with a good degree of precision, we can incorporate this knowledge into our reference system and modify the structure of G_{refCL} as

$$G_{refCL1}(s) = \frac{1}{(\frac{1}{K_r})s^{\gamma} + 1} e^{-Ls} \tag{3.12}$$

Then the parameters are tuned according to this modified reference model (eq. (3.12)).

In the context of the closed-loop control of time-delay systems, Smith (1957) proposed a control scheme called the Smith predictor that leads to the improved closed loop performance compared to the conventional controller. Fig. 3.4 shows a configuration for the Smith predictor. Here, $G_0(s)$ is the nominal model of the time-delay system G(s) and $P_0(s)$ is the time-delay free model. If $G_0(s)$ models the plant G(s) perfectly, the closed-loop stability depends on the controller C(s) and on the delay-free model $P_0(s)$ only, and any closed-loop dynamics can be obtained. The closed loop transfer function relating response y(t) with the set point r(t) and input disturbance d(t) is given as

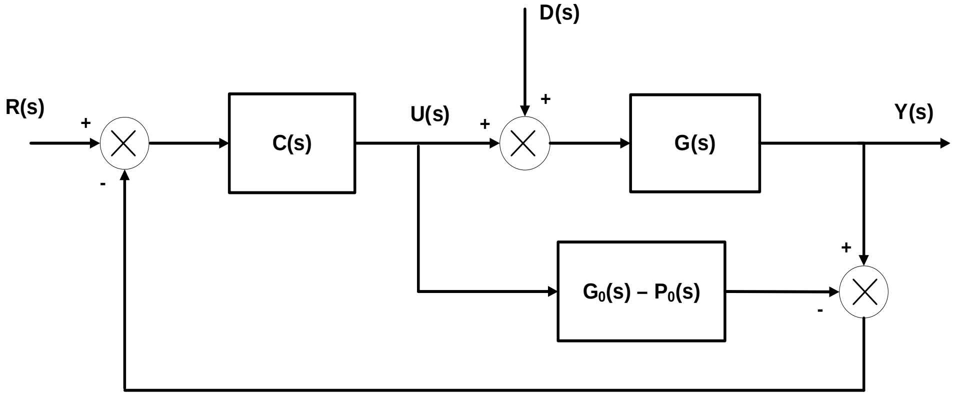


Figure 3.4: Smith Predictor formulation

$$y(t) = \frac{C(s)G(s)}{1 + C(s)\,[G(s) + P_0(s) - G_0(s)]}\,r(t) + \frac{G(s)\,[1 + C(s)\;(\;P_0(s) - G_0(s)\;)]}{1 + C(s)\,[G(s) + P_0(s) - G_0(s)]}\,d(t) \tag{3.13}$$

It is not really important to choose an accurate high-order model G_0 for the control of an uncertain plant G. The Smith predictor technique provides a time-delay compensation to eliminate the actual delayed output and makes it possible to design the primary controller (C(s)) assuming no time-delay in the control loop. Smith predictor has been shown to provide closed loop performance improvement over conventional feedback control if the model errors are not too large. Using Smith predictor in the proposed formulation, for systems with time delays, FO-PI controller is tuned using the delay free part of the system model while it is implemented using the Smith predictor framework (see Fig. 3.4). When there is a mismatch between true and estimated time delay, the closed loop performance using Smith predictor deteriorates.

3.3.2 Imposing constraints

Additional constraints can also be introduced to the above optimization problem. The constraints could be limits on the sensitivity function (S) and the complimentary sensitivity function (T):

$$S = \frac{1}{1 + C(s)G(s)} \tag{3.14}$$

$$T = \frac{C(s)G(s)}{1 + C(s)G(s)} \tag{3.15}$$

In order to reject high-frequency noise, the closed loop transfer function (T) must have a small magnitude at high frequencies; thus it is required that at some specified frequency ω_h or for a range of frequencies $\omega_i < \omega_h$, its magnitude be less than some specified gain:

$$\left| \frac{C(j\omega_h)G(j\omega_h)}{1 + C(j\omega_h)G(j\omega_h)} \right| < H \tag{3.16}$$

Also, the sensitivity function must have a small magnitude at low frequencies in order to reject output disturbances and closely follow references; thus it is required that at some specified frequency ω_l or for a range of frequencies $\omega_i < \omega_l$, its magnitude be less than some specified gain:

$$\left| \frac{1}{1 + C(j\omega_l)G(j\omega_l)} \right| < P \tag{3.17}$$

Please note that the optimization algorithm is independent of whether we work in the time domain or frequency domain, that is to say imposing these constraints in frequency domain where the objective function is defined in the time domain doesn't affect the optimization algorithm. Although the computational burden would be more to compute the tuning parameters using the proposed algorithm, it should be noted that the controller parameters are computed offline.

3.3.3 Algorithm for the proposed method

The algorithm for finding the parameters for FO-PI controller ($\theta = [K_c, K_i, \lambda]^T$) by imposing the constraints is given below.

Step 1 Fix K_r and γ.

Step 2 Initialize the algorithm with some initial value for θ. We can start with tuning parameters for classical PI controller based on integer order approximation of the fractional order process.

Step 3 Iterate on θ based on the minimization of this constraint optimization problem

$$\hat{\theta} = arg \min_{\theta} \begin{bmatrix} w_y \ \sum_{k=1}^{N_s} [y(k,\theta) - y_{ref}(k)]^2 + \\ w_u \ \sum_{k=1}^{N_s} [\Delta u(k,\theta)]^2 \end{bmatrix}$$

subject to

$$0 < \lambda < 2$$
$$|T(j\omega_h)| \ < \ H \text{ and } |S(j\omega_l)| < P$$

where $y(t,\theta)$ and y_{ref}(t) are the closed loop step responses of the system with FO-PI controller (with tuning parameter vector θ) and the reference system, respectively. $\Delta u(k,\theta) = u(k,\theta) - u(k-1,\theta)$ represents the change in control input moves while w_y and w_u are weights or penalties for the two terms in the cost function.

The weighting matrices w_y and w_u can be effectively used to specify relative importance of errors and manipulated input moves. Choosing the appropriate parameters for the reference system decides both the robustness (for example, phase margin) and the performance (set point response) of the closed loop system. There is no simple way to choose these tuning parameters.

The closed loop response depends on the weights and since the tuning parameters are computed offline, setting weights to improve controller performance can be done according to process requirements. The sensitivities of the cost function to these two weights can be computed to determine if the individual weights should be increased or decreased to improve controller performance. The weights can then be adjusted and the cost function can be computed again and this is repeated till one can optimize the controller performance. This is all done offline and therefore is convenient.

3.3.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is relatively new population-based evolutionary computational technique based on group communication behavior between swarms flocks to achieve some optimum property such as nest location or search for food. It was introduced by Kennedy and Eberhart (1995). In PSO, the 'swarm' is initialized in the searching space with a population of random solutions and the objective is to search the solution space by directing the particles towards the best solution encountered in the previous iterations with the intention to observe better solutions over the course of the process and eventually converging on a single optimum. The details on the algorithm can be found in Kennedy and Eberhart (1995) and a modified improved version has been discussed in Liu et al. (2004). More details on the PSO algorithm used in this work has been presented in Appendix A.

In this work, we use the modified PSO algorithm proposed by Liu et al. (2004), with $c_1 = c_2 = 2.5$, $mc = 0.4$ and number of particles = 60 with inertia factor decreasing linearly between 0.9 and 0.4. The defined search space for each parameter is used to find the optimum using the PSO algorithm. PSO requires rigorous search over the defined space and for the present study, this has been proved to be very effective to find the global optimum.

3.3.5 Performance and Robustness comparison

While comparing the performance of different controllers, the trade-off between robustness and performance has to be taken into account. The performance of the controllers can be assessed in many ways: by comparing the integral square error (ISE) or by comparing the standard deviation of the error signal if the set point is a stationary signal. ISE is defined as

$$\text{ISE} = \sum_{k=1}^{N_s} [y(k) - r(k)]^2 \tag{3.18}$$

where N_s is the length of the simulation trial and $r(k)$ represents the set point. A controller that provides minimum ISE for a closed loop response gives the good closed loop performance. Alternatively, the standard deviation of the error signal $(y(k) - r(k))$ or a cost function, J_c (defined in eq. 3.20) can be used as a measure of the performance of the closed loop system.

$$J_e = std[y(k) - r(k)] \tag{3.19}$$

$$J_c = std[y(k) - r(k)] + std[\Delta u] \tag{3.20}$$

The lower the value of these indices, the better the performance of the closed loop system for that controller.

In recent years the maximum sensitivity function has been more and more accepted as an exclusive robustness measure, see Åström and Hägglund (1995).

$$||S(s)||_\infty = \max_\omega \ |S(j\omega)| \ = \max_\omega \ \left| \frac{1}{1 + C(j\omega)G(j\omega)} \right| \tag{3.21}$$

As $||S(s)||_\infty$ decreases, the closed loop system becomes more robust.

It is not entirely fair to compare the closed loop performance of the designed FO-PI controller with some of the classical PI controller settings or tuning design from other methods such as Monje et al. (2004*b*); Chen et al. (2008) and Padula and Visioli (2011), as all the tuning rules are based on different performance and/or robustness criterions. The comparison is presented to give an idea about the benefits for designing a FO-PI controller. For fractional order systems, modeling errors will arise by using the approximate rational model to tune classical PI controllers while the actual process is a fractional order system. This can deteriorate the closed loop performance of a classical PI controllers tuned using standard methods even with using a Smith predictor.

3.4 Simulation Study

The efficacy of the proposed tuning method is demonstrated by carrying out FO-PI controller design on two different FO models. Here, the Oustaloup approximation with $N = 12$ over a wider frequency interval $[10^{-3}, 10^{3}]$ is used for simulating the fractional order system. We have compared both the regulatory and the servo performance of the designed controllers and for regulatory control, we introduced a step change in input type disturbance for the examples presented below.

3.4.1 Example 1

The example of a heating furnace as considered in Podlubny (1999*a*), can be modeled by an integer as well as a fractional order differential equations. The fractional model is given as

$$G_{FO_4}(s)(s) = \frac{1}{14994s^{1.31} + 6009.5s^{0.97} + 1.69} \tag{3.22}$$

The reference model in Eq. (3.4) can be selected by fixing K_r and γ. We can shape the output response, close to the desired response, by varying the reference tuning parameters (K_r, γ). The design specifications for the reference system are chosen as

- $\gamma = 7/6$, or phase margin $\sim 75^{\text{o}}$.
- K_r= 0.005

We did not impose any constraints on the closed loop system. For this process, we kept $w_y = 1$ and $w_u = 0$. For PSO the initial range of parameters are selected as $K_c \in$ [0,500], $K_i \in$ [0,100], $\lambda \in$ [0.01,2]. For these specifications, the FO-PI controller obtained based on the proposed algorithm is

$$C_{FO_4}(s) = 73.67 + \frac{9.123}{s^{0.313}} \tag{3.23}$$

Next we examined the robustness property of this closed loop system by varying the process gain (K) by +40% to −40% i.e. K = {0.6, 0.8, 1, 1.2, 1.4}. The closed loop step responses and the Bode plots for the open loop systems (G(s)C(s)) are illustrated in Figures 3.5 and 3.6 respectively. These plots show that the closed loop system with the FO-PI controller tuned by the proposed method is robust against process gain variations and the step responses exhibit the iso-damping property. In the open-loop Bode plots it is seen that the phase curve is flat around the gain crossover frequency ω_c and other low and moderate frequency region and that the system has a phase margin of approximately 75^o.

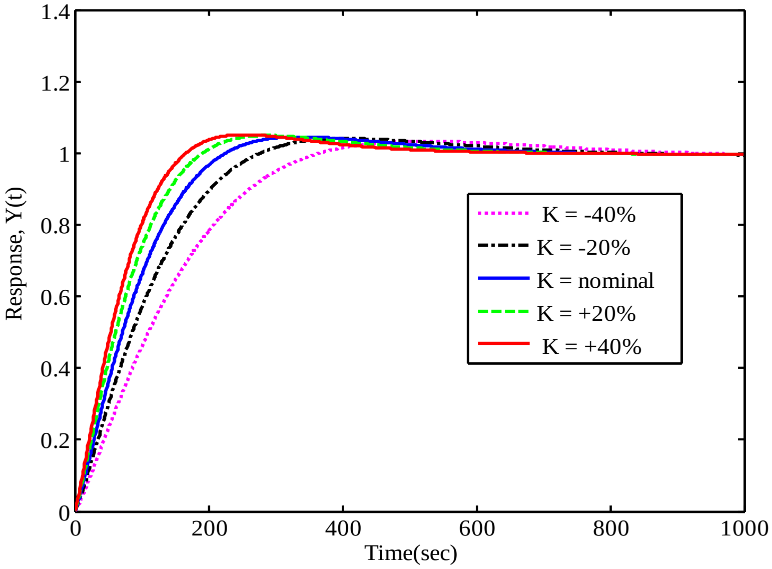


Figure 3.5: Closed loop step responses for G_{FO_4} with $C_{FO_4}(s)$ under different process gain variations

3.4.2 Example 2

A fractional system model with time delays is considered here. The FO system model and its integer model approximation (using the plant's step response

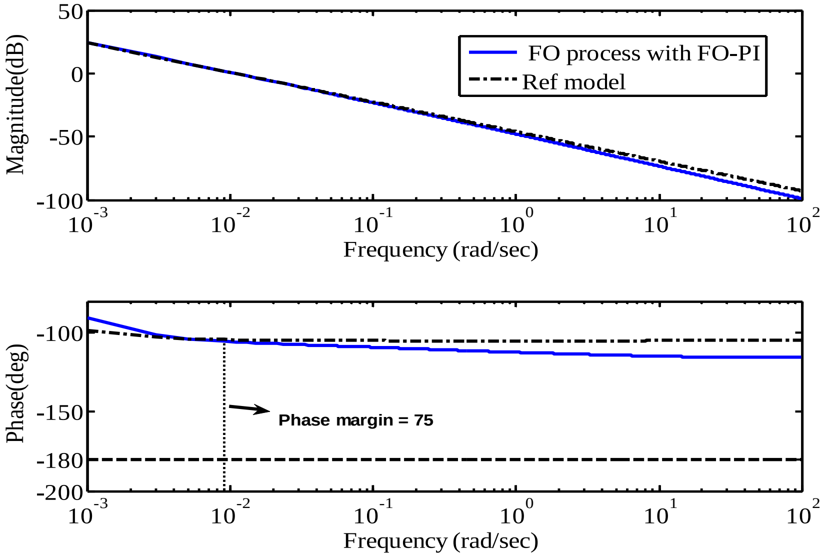


Figure 3.6: Open-loop Bode diagram (with K=1) for G_{FO_4} with $C_{FO_4}(s)$ controller

(Valério and Costa, 2006) are given as

$$G_{FO_5}(s) = \frac{K}{s^{0.5}+1}e^{-0.5s} \tag{3.24}$$

$$G_{IO_5}(s) = \frac{K}{1.5s+1}e^{-0.1s} \tag{3.25}$$

with nominal $K = 1$. The design specifications for the reference system are chosen as

- $\gamma = 9/8$,
- K_r= 2.0 or $1/K_r = 0.5$.
- $|S(j\omega_l)| < -20dB$ at $\omega_l = 0.01\ rad/s$.

Equal weighing matrices ($w_y = w_u = 1$) are used here in the objective function. For PSO the initial range of parameters are selected as $K_c \in$ [0,5], $K_i \in$ [0,5], λ $\in$ [0.01,2]. As the process model include time delays, for these specifications we design two different FO-PI controllers based on the two methods discussed in the previous section:

1. Based on the modified reference model
2. Using the Smith predictor

Based on the modified reference model

The FO-PI controller obtained based on this scheme from the algorithm is

$$C_{FO_{51}}(s) = 0.44 + \frac{1.77}{s^{0.88}} \tag{3.26}$$

The robustness property of this closed loop system is examined by varying the

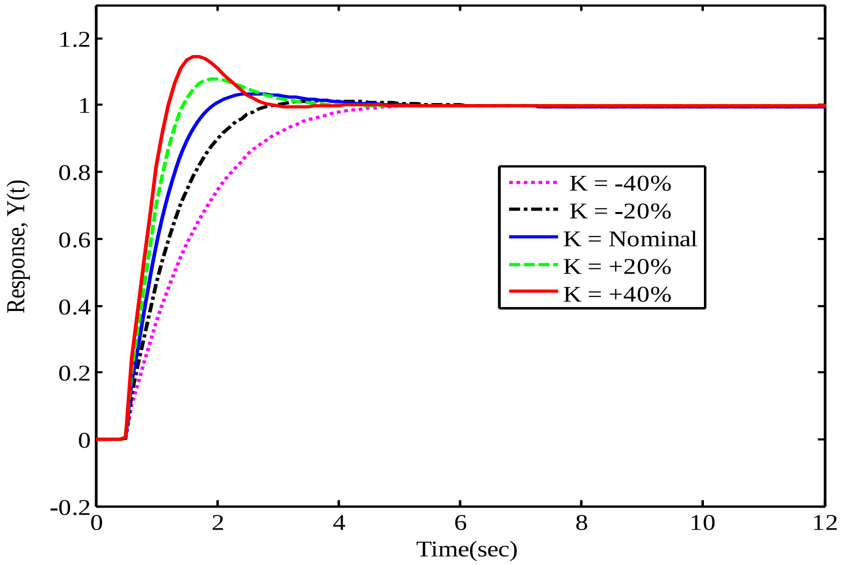


Figure 3.7: Closed loop step responses for G_{FO_5} with $C_{FO_{51}}(s)$ under different process gain variations

process gain (K) by $+40\%$ to -40% i.e. K = $\{0.6, 0.8, 1, 1.2, 1.4\}$. Fig. 3.7 shows that the closed loop system with the FO-PI controller tuned by the proposed method is fairly robust against process gain variations although the iso-damping property is lost because of the presence of delays in the system.

We have an integer order approximation available for this system, so we can compare the performance of this controller with results from standard tuning rules such as IMC-PI (Chien and Fruehauf, 1990) and H&A (Hägglund and Åström, 2002) for classical PI controller settings. Also, the tuning rules for fractional PI controller from Chen et al. (2008) and Monje et al. (2004*b*) are used for comparison purpose. The IMC-PI tuning is obtained using a closed loop time constant of 0.5 sec and the tuning rule for Chen et al. (2008) is obtained using an integer model while the controller from Monje et al. (2004*b*) is obtained using $w_c = 1.4$ with a phase margin of 70^o.

Table 3.3: Controller parameters for process G_{FO_5}

Tuning algorithm	K_c	K_i	λ
FO-PI	0.440	1.78	0.88
IMC-PI	2.500	1.667	1.00
H&A	4.340	1.055	1.00
Chen et al. (2008)	0.318	0.023	1.00
Monje et al. (2004b)	1.195	2.971	1.265

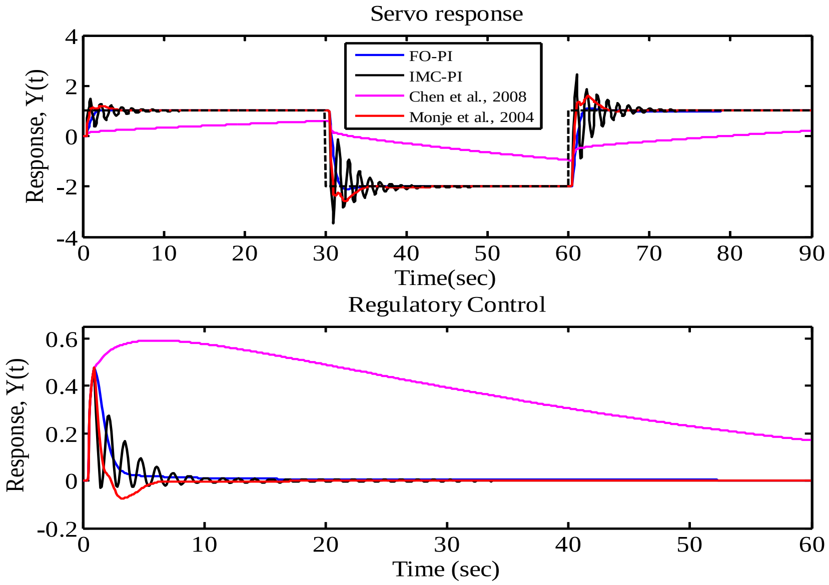


Figure 3.8: Servo and regulatory controller performance for process $G_{FO_{51}}$

Fig. 3.8 shows the closed loop responses from these settings. The H&A controller tuned based on integer order approximation has not been shown as it gave unstable response if implemented on the actual fractional order transfer function model while the Chen et al. (2008) controller gave a sluggish response. Table 3.4 gives the ISE and J_e, J_c values for the closed loop systems using all the controllers. Table 3.5 gives the maximum sensitivity functions using the three controllers. As can be seen FO-PI controller provides both good servo control and regulatory performance. Although, fractional order PID controller from Chen et al. (2008) provides better robustness compared to other controllers, it gives sluggish servo and regulatory performance. FO-PI controller provides sufficient robustness to the closed loop system with good servo and regulatory performance (for step change in input disturbance). The IMC-PI controller provides an oscillatory closed loop

Table 3.4: Comparison of servo and regulatory performance for the closed loop systems for G_{FO_5} using modified reference mode

Tuning algorithm	Servo		Regulatory	
	J$_c$	J$_e$	J$_c$	J$_e$
FO-PI	0.512	0.418	0.062	0.058
IMC-PI	0.978	0.437	0.063	0.049
H&A	Unstable response			
Chen et al. (2008)	1.205	1.159	0.115	0.108
Monje et al. (2004b)	0.633	0.381	0.053	0.046

Table 3.5: Comparison of maximum sensitivity function for controllers for G_{FO_5}

Tuning algorithm	$\|\| S(s) \|\|_\infty$
FO-PI	1.586
IMC-PI	6.252
H&A	8.169
Chen et al. (2008)	1.117
Monje et al. (2004b)	2.079

response for these fractional system models and the H&A controller provides an unstable response.

Using the Smith predictor

For this scheme the controller is designed using the delay free part of the process model, however it is implemented in the Smith predictor framework. We can compare the performance of FO-PI controller with results from standard tuning rules like IMC (Chien and Fruehauf, 1990) for classical PI controller settings and the tuning rules for fractional PI controllers from Monje et al. (2004*b*). The IMC tuning is obtained using a closed loop time constant of 0.5 sec and Monje et al. (2004*b*) controller is obtained using $w_c = 1.4$ and phase margin $= 70^o$. The IMC-PI controller is designed using the delay free part of integer order approximation, however it is implemented on the fractional order model. Fig. 3.8 shows the closed loop response from these settings. Table 3.6 gives the ISE and J_e, J_c values for the closed loop systems using all the controllers. There is not much improvement in the servo closed loop performance compared to the results without the Smith predictor, however, the regulatory performance is slightly better using the Smith predictor. The closed loop response (both servo and regulatory) for the IMC-PI controller is still oscillatory as compared to the FO-PI controller performance.

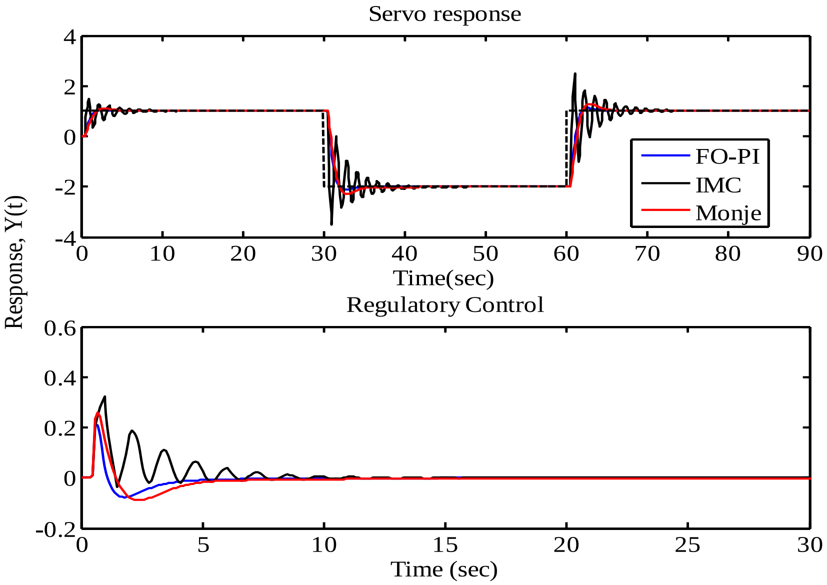


Figure 3.9: Servo and regulatory controller performance with Smith predictor for G_{FO_5}

Table 3.6: Comparison of servo and regulatory performance for the closed loop systems for G_{FO_5} using Smith predictor

Tuning algorithm	Servo		Regulatory	
	J$_c$	J$_e$	J$_c$	J$_e$
FO-PI	0.510	0.419	0.031	0.018
IMC-PI	1.044	0.436	0.081	0.033
Monje et al. (2004b)	0.524	0.446	0.037	0.025

The deterioration in performance is mainly attributed due to the approximations involved in tuning integer order controllers.

3.5 FO-PI design for lag/delay dominant processes

Lag dominant and delay dominant cases are studied using the integer order model. The two cases are presented below.

- Lag dominant case:

$$G_{IO_6}(s) = \frac{1}{s+1}e^{-0.1s} \tag{3.27}$$

Here $\tau = \frac{L}{T+L} = 0.0909$.

The FO-PI controller is designed using $\gamma = 9/8$, K_r= 2.0 or $1/K_r = 0.5$ and $w_y = 1$, $w_u = 0.5$. For PSO the initial range of parameters are selected as $K_c \in$ [0,5], $K_i \in$ [0,5], $\lambda \in$ [0.01,2]. Again, IMC-PI and H&A for classical PI controller settings and fractional PI controllers from Chen et al. (2008) and Monje et al. (2004*b*) are used for comparison purpose. The IMC-PI tuning is obtained using a closed loop time constant of 0.5 sec and the controller from Monje et al. (2004*b*) is obtained using $w_c = 0.5$ with a phase margin of 60^o. Table 3.8 gives the J_e and J_c values for the closed loop system using different controllers for servo as well as regulatory control and Fig 3.10 shows the closed loop responses from these settings. As can be seen FO-PI controller like other controller tuning method, provide good servo and regulatory performance, comparable to well tuned IMC-PI, and better than fractional order controller using Monje et al. (2004*b*) tuning rule; however, Chen et al. (2008) tuning provides the better servo and regulatory performance compared to other tuning rules for this lag dominant process.

Table 3.7: Controller parameters for process G_{IO_6}

Tuning algorithm	K_c	K_i	λ
FO-PI	1.186	2.010	0.95
IMC-PI	1.667	1.667	1.00
H&A	2.940	0.856	1.00
Chen et al. (2008)	3.265	8.008	0.70
Monje et al. (2004b)	0.409	0.509	1.229

Table 3.8: Comparison of servo and regulatory performance for the closed loop systems for G_{IO_6}

Tuning algorithm	Servo		Regulatory	
	J_c	J_e	J_c	J_e
FO-PI	0.320	0.260	0.065	0.061
IMC-PI	0.335	0.247	0.066	0.060
H&A	0.389	0.232	0.077	0.059
Chen et al. (2008)	0.374	0.185	0.028	0.021
Monje et al. (2004b)	0.469	0.447	0.158	0.155

- Delay dominant case:

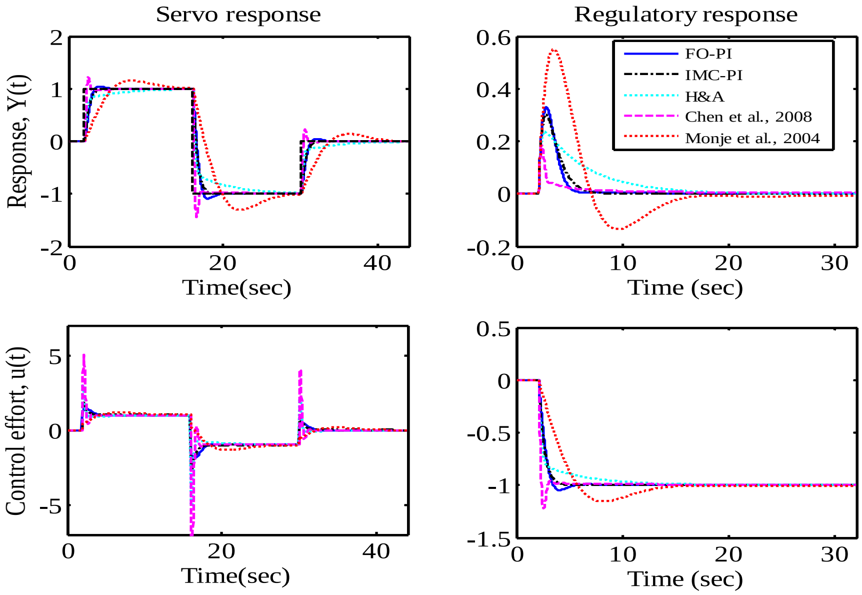


Figure 3.10: Servo and regulatory controller performance for G_{IO_6}

$$G_{IO_7}(s) = \frac{1}{s+1}e^{-2.5s} \tag{3.28}$$

Here $\tau = \frac{L}{T+L} = 0.714$.

The design specifications for FO-PI controller are: $\gamma = 9/8$, K_r= 0.33 or $1/K_r$ = 3 and $w_y = 1$, $w_u = 0.5$ are used in the objective function. For PSO the initial range of parameters are selected as $K_c \in$ [0,5], $K_i \in$ [0,5], $\lambda \in$ [0.01,2]. Again, IMC-PI and H&A for classical PI controller settings and fractional PI controllers from Chen et al. (2008) and Monje et al. (2004*b*), are used for comparison purpose. The IMC-PI tuning is obtained using a closed loop time constant of 3sec and the controller from Monje et al. (2004*b*) is obtained using $w_c = 0.5$ with a phase margin of 60^o. Table 3.10 gives the J_e and J_c values for the closed loop systems using all

Table 3.9: Controller parameters for process G_{IO_7}

Tuning algorithm	K_c	K_i	λ
FO-PI	0.340	0.184	1.072
IMC-PI	0.154	0.154	1.00
H&A	0.252	0.232	1.00
Chen et al. (2008)	0.417	0.236	1.10
Monje et al. (2004b)	0.910	0.223	1.295

the controllers for servo as well as regulatory control and Fig. 3.11 shows the closed loop responses from these settings. Again, FO-PI controller provides both good

Table 3.10: Comparison of servo and regulatory performance for the closed loop systems for G_{IO_7}

Tuning algorithm	Servo		Regulatory	
	J_c	J_e	J_c	J_e
FO-PI	0.542	0.515	0.189	0.185
IMC-PI	0.584	0.570	0.213	0.205
H&A	0.537	0.515	0.198	0.184
Chen et al. (2008)	0.532	0.499	0.184	0.178
Monje et al. (2004b)	0.545	0.475	0.171	0.166

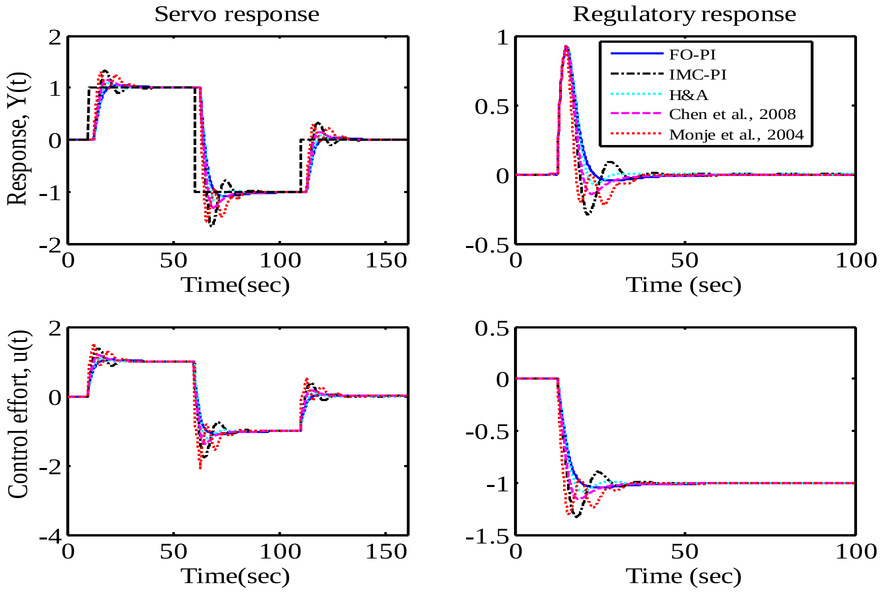


Figure 3.11: Servo and regulatory controller performance for G_{IO_7}

servo control and regulatory performance, and comparable to fractional order PI controller using Monje et al. (2004*b*) and Chen et al. (2008) tuning rule.

3.6 Control strategy applied to Thermal diffusion in a wall

For many real processes, fractional differentiation appears naturally when the system transients are governed by a diffusion equation, and particularly between the variables governing the functioning of the interface. Benchellal et al. (2006) has shown that the transfer function H(s) relating heat flux and the temperature, on the front face of a heated wall (which is governed by classical heat conduction equation), comes out as fractional order transfer function with half integer order. We consider a classical wall (CW) problem as a process for this simulation study as introduced in Chapter 2 to illustrate the importance of our proposed algorithm on a real physical system.

For this study, we define the input and output as $y(t) = T(0,t)$ and heat input (not flux) as, $u(t) = \Omega_{in}(t)S_A$, then the transfer function relating $y(t)$ and $u(t)$ is given as (Gabano and Poinot, 2011)

$$\hat{H}_R(s) = \frac{1}{S_A}\frac{\sum_{i=0}^{R-1} b_i s^{\frac{i}{2}}}{\sum_{i=0}^{R-1} a_i s^{\frac{i}{2}} + s^{\frac{N}{2}}} \tag{3.29}$$

where the constants are defined as

$$a_0 = \frac{R!}{2^{R-1}}\left(\frac{\zeta^{\frac{R}{2}}}{l^R}\right) \tag{3.30}$$

$$a_i = \left(\frac{R!}{i!2^{R-1}l^{R-i}}\right)\zeta^{\frac{R-i}{2}}, \quad 1 \leq i \leq R-1 \tag{3.31}$$

$$b_i = \left(\frac{R!}{(i+1)!\ 2^{R-i-1}l^{R-i-1}}\right)\frac{\zeta^{\frac{R-i}{2}}}{\varrho}, \quad 0 \leq i \leq R-1 \tag{3.32}$$

The detailed derivation of this is presented in Chapter 4. This model for classical wall problem in Eq. (3.29) with R= 8 is used to design the FO-PI controller using the proposed algorithm. The design specifications for the reference system are chosen as

- $\gamma = 9/8$,
- K_r= 0.030 or $\omega_c = 0.0495\ rad/s$.

The sampling time is chosen as $0.1sec$. We do not impose any constraints on the sensitivity and complementary sensitivity functions for this closed loop system. Also, equal weighing matrices (equal to identity matrix) are used in the objective function. For these specifications, the estimated tuning parameters of the FO-PI controller is

$$C_{FO_8}(s) = 1.462 + \frac{2.475}{s^{0.844}} \tag{3.33}$$

In the open-loop Bode plot Fig. 3.12, the phase curve is flat for most of the frequencies and the system has a phase margin of approximately 78^o at crossover

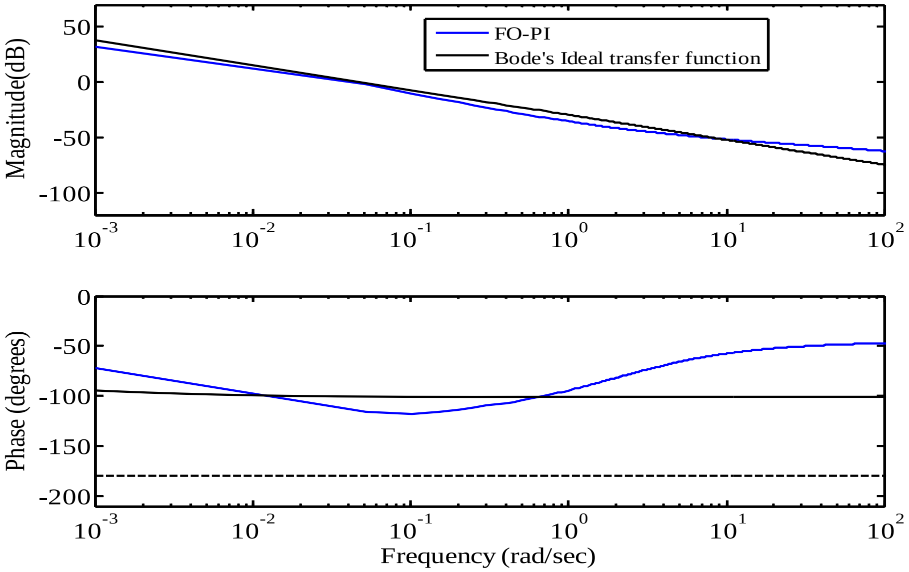


Figure 3.12: Open-loop Bode diagram at nominal condition for CW example

frequency of $\omega_c = 0.044$. An approximate integer order model (first order plus time delay model) is developed for this system to design a classical PI controller and to compare the closed loop performance for these controllers. The integer order approximation is

$$G_{IO_8}(s) = \frac{0.045}{29.5080s + 1} e^{-4.0499s} \tag{3.34}$$

For comparison purpose, we design a classical PI controller using IMC tuning and a fractional PI controller designed using Chen et al. (2008) and Monje et al. (2004b) settings. The IMC-PI controller is designed using a closed loop time constant of 50 sec while the FO-PI controller designed using Monje et al. (2004b) is obtained using $w_c = 0.05$ with a phase margin $= 60^o$. The tuning parameters are derived from the approximate integer order model for this system but is implemented on the real fractional order system. The servo and regulatory performance from the controllers is shown in Fig. 3.13. Table 3.11 gives the J_e and J_c values for the closed loop system using different controllers for servo as well as regulatory control. As can be seen FO-PI controller like other controller tuning method, provide both good servo control and regulatory performance, better than well tuned IMC-PI, and fractional order controller using Monje et al. (2004b) tuning rule and Chen et al. (2008) tuning. Even though with Chen et al. (2008) tuning, the set point tracking is faster, there is excessive control action as reflected in the J_c value for the same.

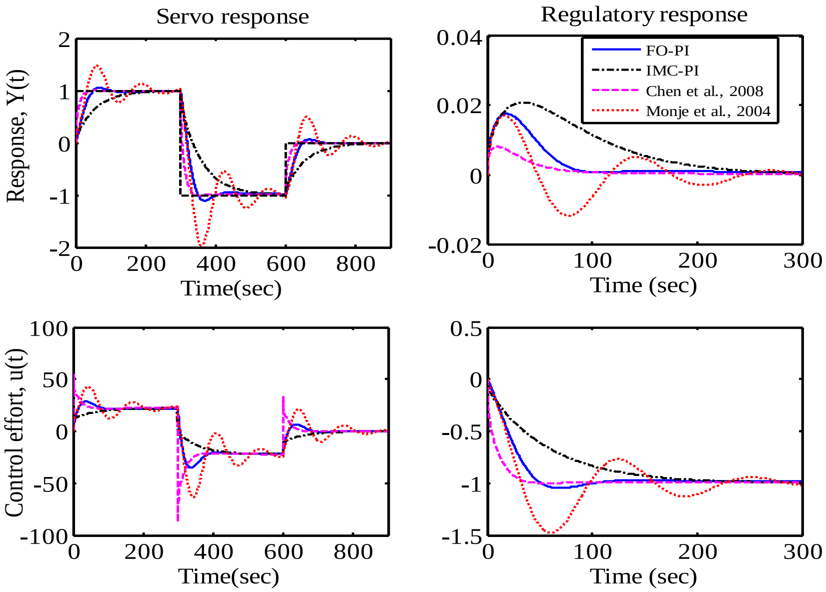


Figure 3.13: Servo and regulatory controller performance for CW

Table 3.11: Comparison of servo and regulatory performance for the closed loop systems for CW

Tuning algorithm	Servo		Regulatory	
	J_c	J_e	J_c	J_e
FO-PI	0.556	0.303	0.019	0.0051
IMC-PI	1.336	0.318	0.011	0.0057
H&A	2.710	0.130	0.027	0.0012
Chen et al. (2008)	3.091	0.154	0.012	0.0021
Monje et al. (2004b)	0.891	0.389	0.020	0.007

3.7 Experimental validation

The FOPI controller designed using the proposed algorithm is applied to a continuous stirred tank heating (CSTH) process. A schematic of the process is shown in Fig. 3.14. The experimental setup is located at the Computer Process Control laboratory in Department of Chemical and Materials Engineering at University of Alberta. The cylindrical glass tank is equipped with steam coil with a controlled input facilitating the manipulation of steam flow to control temperature of water coming out of the tank. The level of water inside the tank is controlled by manipulating the inlet water flow. The water outlet and condensate flow is controlled only manually. The outlet water flows through a big pipeline and a number of thermocouples (two for this case study) are placed at different distances in this pipe to introduce time delay into the system. The laboratory has Emerson Delta-V distributed control system (DCS) to control this process. For this study, the steam flow rate to the tank is used as the manipulated variable and the outlet water temperature is used as the controlled variable. The changes in the room temperature and the steam supply pressure can be considered as disturbances. The hot water flow valve is kept closed for this experiment.

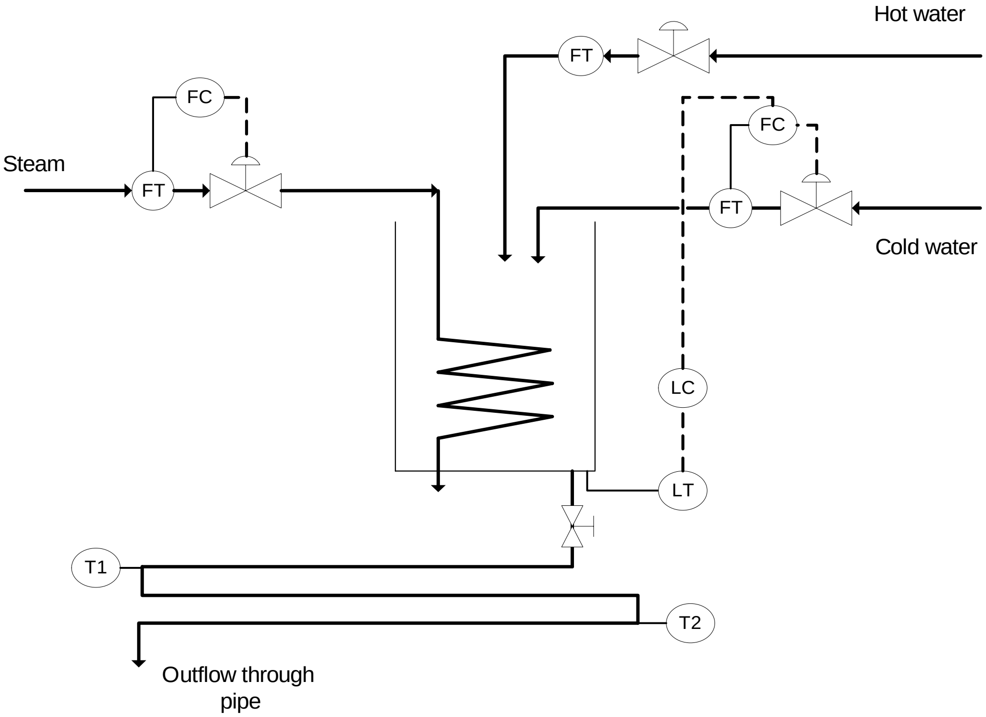


Figure 3.14: Continuous stirred tank heater (CSTH) process

To design FO-PI controller or PI controller we need to model this process first. For generating data for model identification, input perturbations in steam flow (U(t)) are introduced into the process and the temperature measurements (T_1 or T_2) are used as process output to model the process. The steam flow controller is set in auto mode and set point to the steam flow controller is used as the process input. During this experiment the tank level was kept constant at 40% filled using a level controller (PI) cascaded with the cold water flow controller and the output flow manual valve was set at 50% open position. Fig. 3.15 presents the experimental setup of CSTH located at Computer Process Control laboratory at University of Alberta.

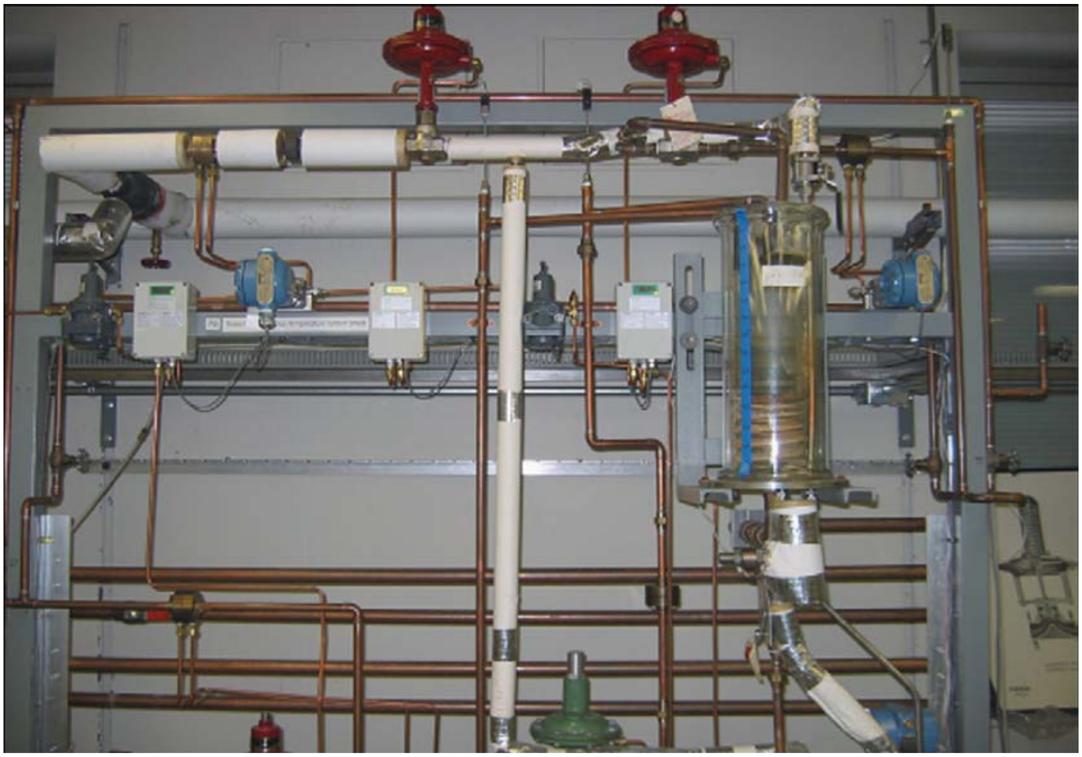


Figure 3.15: Experimental setup for CSTH at University of Albert

The overall data as shown in Fig. 3.16 was partitioned into two parts : (a) an identification set: comprising of the first 60% data points and (b) the validation data set: comprising of the last 40% data points. The temperature from both the thermocouples is measured and is shown in Fig. 3.16. The process is modeled by a first-order transfer function with time delay. The identified model for T_1 in transfer function form is given in equation (3.35) and the model validation is shown in Fig. 3.17.

$$G_{T_1}(s) = \frac{T_1(s)}{U(s)} = \frac{1.441}{41.12s+1}\ e^{-37.3s} \tag{3.35}$$

This transfer function model represents a balanced lag/delay dominant process ($\frac{37.3}{37.3+41.12} = 0.476$). If we write the transfer function for T_2, the process dynamics for T_2 is essentially same as T_1 except that the time delay has increased.

$$G_{T_2}(s) = \frac{T_2(s)}{U(s)} = \frac{1.441}{41.12s+1}\ e^{-67s} \tag{3.36}$$

This nominal process model (Eq. 3.35 and Eq. 3.36) is now used to design FO-PI controller using the proposed algorithm as well as designing classical PI controllers for the comparison purposes. In order to test the robustness property of the two

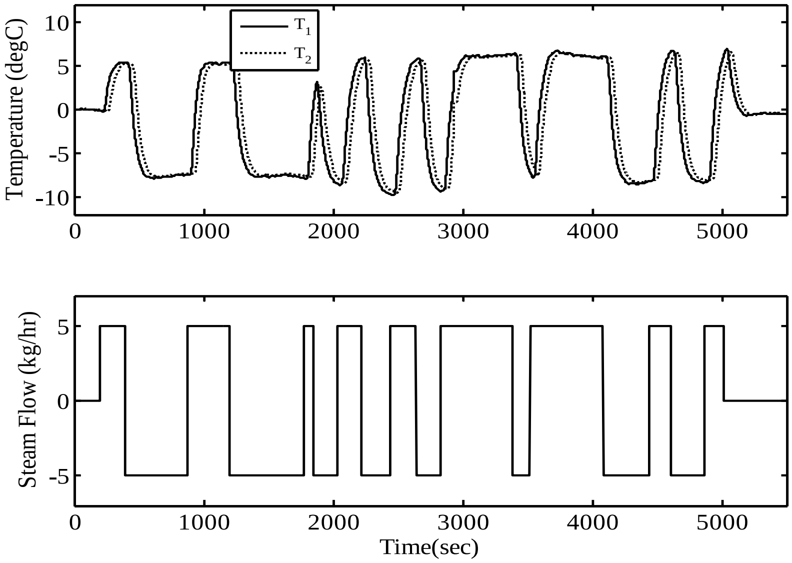


Figure 3.16: Temperature variation to changes in steam flow at nominal condition

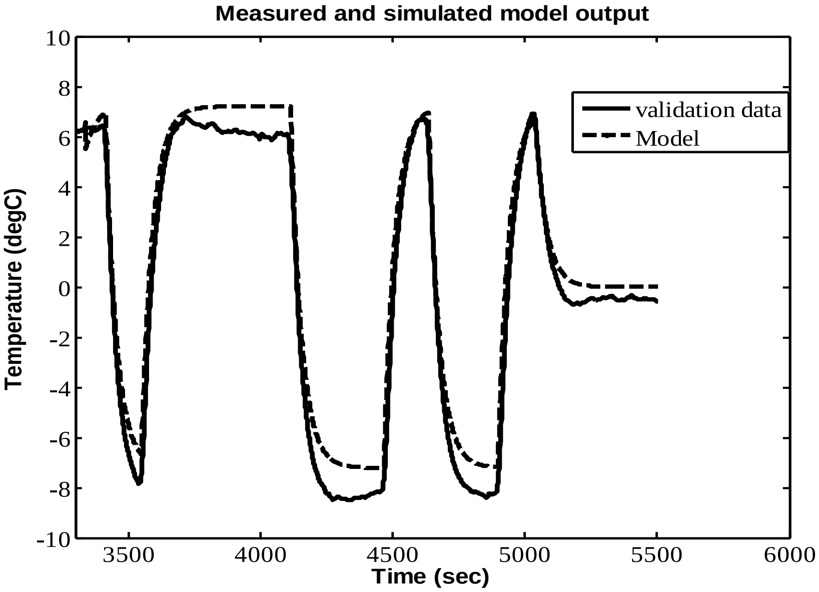


Figure 3.17: Model predictions on the validation set: (- -) black dotted line is model prediction

controllers, we also estimated the process model at a second operating condition. The output flow manual valve was changed from 50% open to 33% open and this causes changes in the dynamics of the water flow inside the tank and in turn changes the dynamics of the temperature loop. Fig. 3.18 shows the open loop data at this operating condition and the identified model at this condition is given as the following transfer function (eq. 3.37)

$$G_{CSTH1}(s) = \frac{1.973}{64.81s + 1}\, e^{-39s} \tag{3.37}$$

Notice that for the second operating condition, all model parameters change by a significant amount. The process gain increases by 37%, the time constant increases by 57% and the time delay changes by 5%. We studied two cases here, where individual controllers are designed and tested using T_1/U(s) as one process (Case-1) and T_2/U(s) as another process (Case-2); however, the robustness of the controller at second operating condition is tested only for Case-1. Note that this process is forced to be modeled as first order plus time delay model; however, the same model is used for the tuning methods for classical PI and fractional PI controllers.

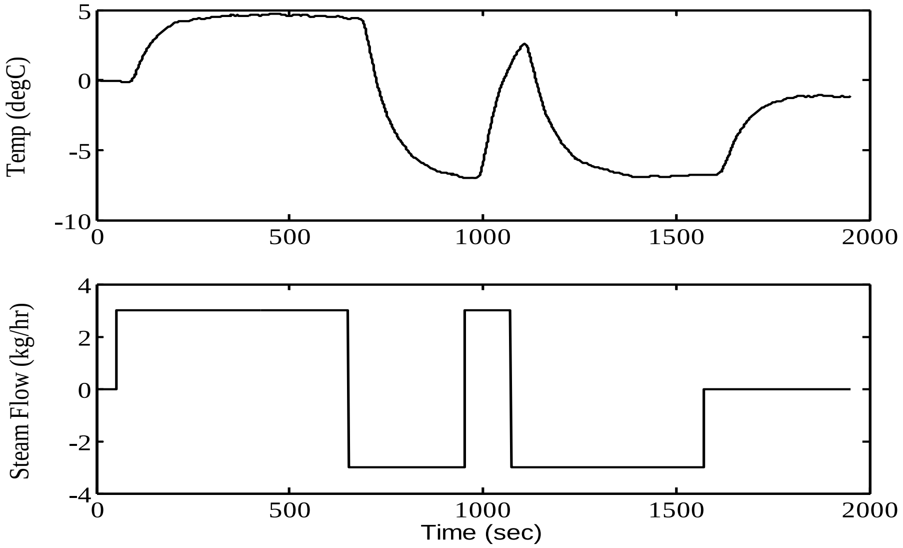


Figure 3.18: Temperature variation to changes in steam flow at operating condition-2

3.7.1 Case-1: Balanced lag/delay process

For this case, the controllers are designed using T_1 as the controlled variable and $G_{T_1}(s)$ as the process model.

The FO-PI controller using the proposed algorithm is designed with the initial range of parameters for PSO selected as $K_c \in$ [0,2], $K_i \in$ [0,2], $\lambda \in$ [0.01,2] and the design specifications for the reference system chosen as

- $\gamma = 7/6$,
- K_r= $1/50 = 0.02$.

Also, we kept $w_y = 1$ and $w_u = 0$. For comparison purpose, we design a classical PI controller using IMC-PI tuning (Chien and Fruehauf, 1990) and a fractional PI controller designed using Chen et al. (2008) settings and compare the performance of these controllers with the designed FO-PI controller at both nominal and second operating condition. The IMC-PI controller is designed using a closed loop time constant of 50 sec while the FO-PI controller designed using Monje et al. (2004*b*) is obtained using $w_c = 0.01$ with a phase margin = 60^o. The controllers were implemented in MATLAB and connected to the Delta-V control system using an OPC interface. The sampling time of 1 sec is used to implement these controllers (using discrete version) on the real process. The execution time for the controllers could be increased because of the slow temperature dynamics. Fig 3.19 shows the closed loop servo responses from these settings. For regulatory control, the position of the manual valve was changed from 50% open to only 33% open at roughly 1800

Table 3.12: Controller tunings for the three controllers for CSTH for balanced lag/delay process

Tuning algorithm	K_c	K_i	λ
FO-PI	0.576	0.0088	1.053
IMC-PI	0.327	0.0079	1.00
Chen et al. (2008)	0.43	0.013	1.00
Monje et al. (2004b)	0.444	0.0024	1.26

Table 3.13: Performance comparison for the controllers at nominal process condition

Tuning algorithm	ISE	J_e	J_c
FO-PI	3195	1.067	1.113
IMC-PI	4184	1.224	1.257
Chen et al. (2008)	3538	1.127	1.163
Monje et al. (2004b)	4208	1.235	1.271

sec and this acts as a disturbance (changes in the level also affects the temperature and more significantly changes the process dynamics) to the process. As can be seen, the FO-PI controller provides both good servo control and regulatory performance compared to other controllers providing minimum J_e and J_c. To test the robustness of the controllers, we looked at the servo performance of the two controllers at the second operating condition. Fig. 3.20 shows the closed loop servo response at the second operating condition where there process-model mismatch and process dynamics are different.

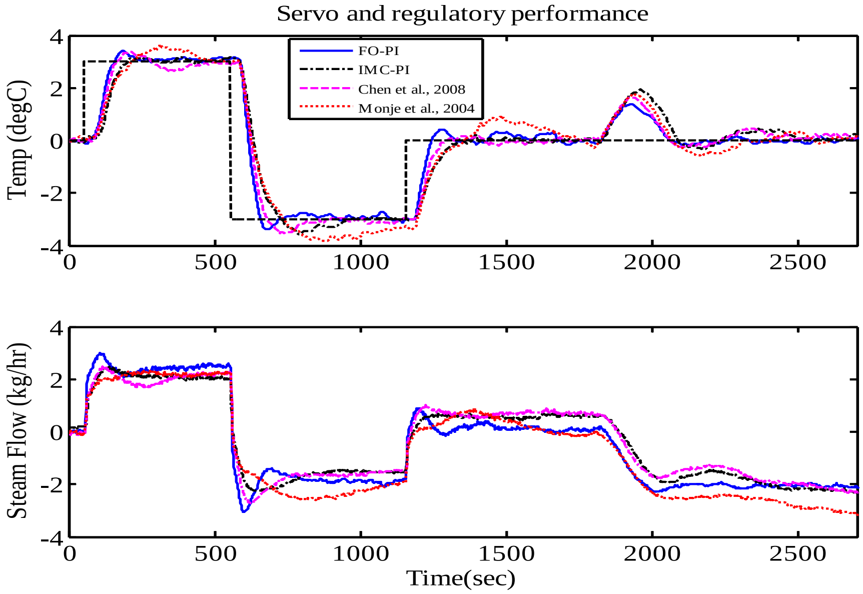


Figure 3.19: Servo and regulatory response for the CSTH process for balanced lag/delay process

As can be seen from Fig. 3.20 that even though the amount of overshoot is more using FO-PI controller, the settling time of the closed loop response is much smaller compared to the other controllers. This shows that the designed FO-PI controller also provides sufficient robustness without affecting the closed loop performance.

3.7.2 Case-2: Delay dominant process

For this case, the controllers are designed using T_2 as the controlled variable and $G_{T_2}(s)$ as the process model.

To design FO-PI controller parameters for PSO are selected as $K_c \in [0,2]$, $K_i \in [0,2]$, $\lambda \in [0.01,2]$ and the design specifications for the reference system chosen

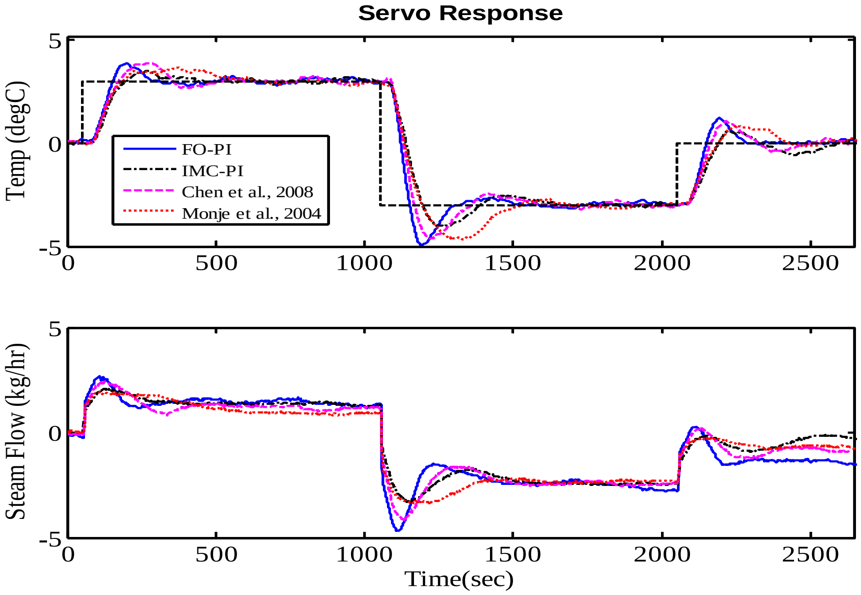


Figure 3.20: Servo response for the CSTH process at operating condition-2

Table 3.14: Performance comparison for the controllers at second operating condition

Tuning algorithm	ISE	J_e	J_c
FO-PI	3528	1.144	1.19
IMC	3965	1.214	1.25
Chen et al. (2008)	3876	1.213	1.25
Monje et al. (2004b)	3988	1.218	1.26

as $\gamma = 7/6$, K_r= 1/75. Also, we kept $w_y = 1$ and $w_u = 0.0$. For comparison purpose, we design a classical PI controller using IMC-PI tuning (Chien and Fruehauf, 1990) and a fractional PI controller designed using Chen et al. (2008) settings and compare the performance of these controllers with the designed FO-PI controller at both nominal and second operating condition. The IMC-PI controller is designed using a closed loop time constant of 75sec while the FO-PI controller designed using Monje et al. (2004b) is obtained using $w_c = 0.01$ with a phase margin $= 60^o$.

The controllers were implemented in MATLAB and connected to the Delta-V control system using an OPC interface. The sampling time of 1sec is used to implement these controllers (using discrete version) on the real process. Fig. 3.21 shows the closed loop servo responses from these settings. For regulatory

Table 3.15: Controller tunings for the three controllers for CSTH for delay dominant case

Tuning algorithm	K_c	K_i	λ
FO-PI	0.484	0.0043	1.089
IMC-PI	0.200	0.0049	1.00
Chen et al. (2008)	0.333	0.064	1.10
Monje et al. (2004b)	0.571	0.0016	1.374

Table 3.16: Performance comparison for the controllers for CSTH for delay dominant case

Tuning algorithm	ISE	J_e	J_c
FO-PI	6519.2	1.229	1.284
IMC	8104.1	1.372	1.396
Chen et al. (2008)	15368	1.891	1.933
Monje et al. (2004b)	8652.1	1.418	1.484

control, the position of the manual valve was changed from 50% open to only 33% open at roughly 3300sec and this acts as a disturbance (changes in the level also affects the temperature and more significantly changes the process dynamics) to the process. As can be seen, the FO-PI controller provides both good servo control and regulatory performance compared to other controllers providing minimum J_e and J_c.

3.8 Conclusions

A fractional order PI controller design method is proposed for fractional order models in this chapter. The proposed strategy is based on a reference model, whose open-loop transfer function is given by Bode's Ideal transfer function. The parameters of the controller are estimated by formulating a constrained non-linear optimization problem. The performance of the fractional order PI controller designed based on the proposed method has been demonstrated through three fractional order dynamic models. The resultant closed-loop system is robust to process gain variations and the step responses exhibit iso-damping property. Simulation results have been presented and analyzed to illustrate the effectiveness of the proposed algorithm. Experimental evaluation of the proposed algorithm has been conducted by designing a FO-PI controller for a computer-interface pilot-scale continuous stirred tank heater (CSTH). The results have shown that the FO-PI

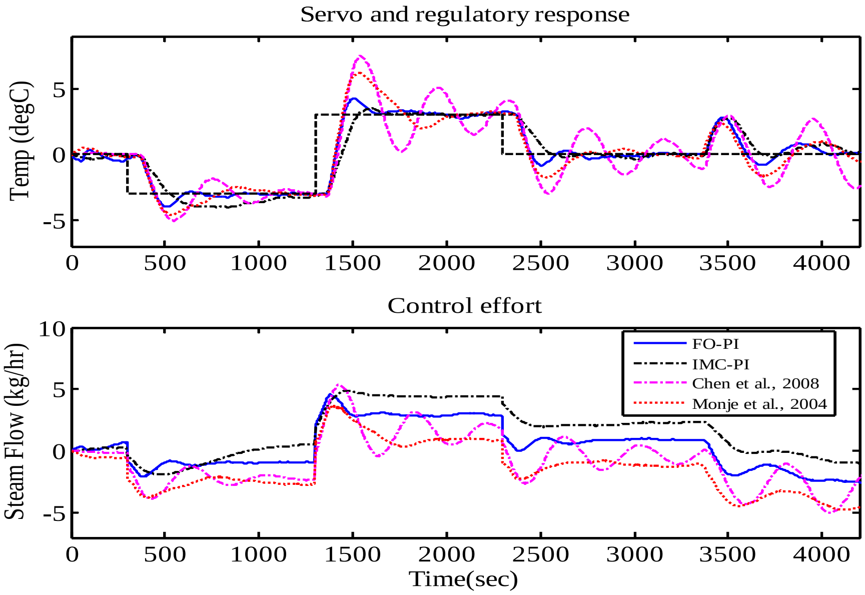


Figure 3.21: Servo and regulatory response for CSTH for delay dominant case

controller tuned with the proposed method provides both good servo control and regulatory performance compared to other controllers in terms of lower J_e and J_c.

Bibliography

K.B. Oldham and J. Spanier. The Fractional Calculus. *Academic Press*, San Diego, 1974.

I. Podlubny. Fractional Differential Equations. *Academic Press*, San Diego, 1999b.

M. Axtell and M.E. Bise. Fractional calculus applications in control systems. *IEEE National Aerospace and Electronics Conference*, New York, USA, pp. 563-566, 1990.

D. Shantanu. Functional Fractional Calculus for System Identification and Controls. *Springer Berlin Heidelberg*, New York, 2008.

C. A. Monje, Y. Q. Chen, B. Vinagre, D. Xue and V. Feliu. Fractional Order Systems and Controls - Fundamentals and Applications. *Springer-Verlag*, London, 2010.

J. B. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *ASME Transactions*, vol. 64, pp. 759-768, 1942.

H.W. Bode. Network Analysis and Feedback Amplifier Design. *Van Nostrand*, New York, 1945.

A. Oustaloup. La Dérivation Non Entière. *Hermés*, Paris, 1995.

I. Podlubny. Fractional order systems and PPI$^\lambda$D$^\mu$- controllers. *IEEE Trans. Automat. Control AC*, vol. 44 (1), pp. 208-214, 1999a.

R. Caponetto, L. Fortuna and D. Porto. Parameter tuning of a non integer order PID controller. *Electronic Proceedings of the 15th International Symposium on Mathematical Theory of Networks and Systems*, 2002.

R. Caponetto, L. Fortuna and D. Porto. A new tuning strategy for a non integer order PID controller. *First IFAC Workshop on Fractional Differentiation and its Applications*, Bordeaux, France, 2004.

B.M. Vinagre, I. Podlubny, A. Hernandez and V. Feliu. Some approximations of fractional order operators used in control theory and applications. *41st IEEE Conference on Decision and Control*, 2002.

C.A. Monje, A.J. Calderón, B.M. Vinagre, Y.Q. Chen and V. Feliu. Proposals for fractional PI$^\lambda$D$^\mu$ tuning. *Fractional Differentiation and its Applications (FDA'04)*, Bordeaux, France, 2004.

C.A. Monje, A.J. Calderǿn, B.M. Vinagre, Y.Q. Chen and V. Feliu. On fractional PI$^\lambda$ controllers: some tuning rules for robustness to plant uncertainties. *Nonlinear Dynamics*, vol. 38, pp. 369-381, 2004.

Y.Q. Chen, T. Bhaskaran and D. Xue. Practical tuning rule development for fractional order proportional and integral controllers. *Journal of computational and nonlinear dynamics*, vol 3(2), pp. 021403-1-021403-8, 2008.

D. Valerio and J.S. Costa. Tuning of fractional PID controllers with Ziegler-Nichols-type rules. *Signal Processing* , vol. 86, pp. 2771-2784, 2006.

J.F. Leu, S.Y. Tsay and C. Hwang. Design of optimal fractional order PID controllers. *Journal of the Chinese Institute of Chemical Engineers*, vol. 33(2), pp. 193-202, 2002.

Y. Luo and Y. Q. Chen. Fractional order [proportional derivative] controller for a class of fractional order systems. *Automatica*, vol. 45(10), pp. 2446-2450, 2009.

J. Y. Cao and B. G. Cao. Design of fractional order controller based on particle swarm optimization. *International Journal of Control, Automation, and Systems*, vol. 4(6), pp. 775-781, 2006.

R.S. Barbosa, J.A.T. Machado and I.M. Ferreira. Tuning of PID controllers based on Bode's ideal transfer function. *Nonlinear Dynamics*, vol. 38, pp. 305-321, 2004.

F. Padula and A. Visioli. Tuning rules for optimal PID and fractional-order PID controllers. *Journal of Process Control*, vol. 21, pp. 69-81, 2011.

Y. Luo, Y. Q. Chen, C. Y. Wang and Y. G. Pi. Tuning fractional order proportional integral controllers for fractional order systems. *Journal of Process Control*, vol. 20, pp. 823-831, 2010.

Y. Luo, Y. Q. Chen and Y. G. Pi. Experimental study of fractional order proportional derivative controller synthesis for fractional order systems. *Mechatronics*, vol. 21, pp. 204-214, 2011.

A.J. Calderón, B.M. Vinagre and V. Feliu. Fractional order control strategies for power electronic buck converters. *Signal Processing* , vol. 86, pp. 2803-2819, 2006.

HongSheng Li, Ying Luo, and Y. Q. Chen. A Fractional Order Proportional and Derivative (FOPD) Motion Controller: Tuning Rule and Experiments. *IEEE Transactions on Control Systems Technology*, vol. 18(2), pp. 516-520, 2010.

H. Delavari, A. N. Ranjbar, R. Ghaderi and S. Momani. Fractional order control of a coupled tank. *Nonlinear Dynamics*, vol. 61, pp. 383-397, 2010.

V. Feliu, R. R. Perez, F.J.C. Garcia, L.S. Rodriguez. Smith predictor based robust fractional order control: Application to water distribution in a main irrigation canal pool. *Journal of Process Control*, vol. 19(3), pp. 506-519 , 2009.

J. Sabatier and A. Oustaloup. Implementation of a thermal platform for the test of non-integer systems. *In Conference on Education of Technologies and Information and Systems Sciences*, Toulouse, France: Univerite Paul Sabatier, 2003.

B. M. Vinagre, I. Petráš, P. Merchan and L. Dorčák. Two digital realizations of fractional controllers: Application to temperature control of a solid. *In European Control Conference (ECC' 2001)*, Porto, Portugal, 2001.

I. Petráš and B.M. Vinagre. Practical application of digital fractional-order controller to temperature control. *Acta Montanistica Slovaca*, vol. 7(2), pp. 131-137, 2002.

C.A. Monje, B.M. Vinagre, V. Feliu and Y.Q. Chen. Tuning and auto-tuning of fractional order controllers for industry applications. *Control Engineering Practice*, vol. 16, pp. 798-812, 2008.

A. Narang, S.L. Shah and Tongwen Chen. Tuning of fractional PI controllers for fractional order system models with and without time delays. *2010 American Control Conference (ACC 2010)*, Baltimore, MD, USA, pp. 6674-6679, 2010.

O. J. M. Smith. Closer control of loops with dead time. *Chem. Eng. Prog.*, vol. 53(5), 1957.

K. J. Åström and T. Hägglund. PID Controllers: Theory, Design, and Tuning. *Instrument Society of America*, Research Triangle Park, NC, 1995.

J. Kennedy and R. C. Eberhart. Particle swarm optimization. *Proc. of the IEEE International Conference on Neural Networks*, pp. 1942-1948, 1995.

Y. Liu, Z. Qin and X. S. He. Supervisor student model in particle swarm optimization. *Proc. of the IEEE Congress on Evolutionary Computation*, pp. 542-547, 2004.

I. L. Chien and P. S. Fruehauf. Consider IMC tuning to improve controller performance. *Chem. Eng. Progress*, vol. 86(10), pp. 33-41, 1990.

T. Hägglund and K. J. Åström. Revisiting the Ziegler-Nichols tuning rules for PI control. *Asian Journal of Control*, vol. 4, pp. 364-380, 2002.

A. Benchellal, T. Poinot and C. Trigeassou. Approximation and identification of diffusive interfaces by fractional systems. *Signal Processing*, vol. 86(10), pp. 2712-2727, 2006.

J.D. Gabano and T. Poinot. Fractional identification algorithms applied to thermal parameter estimation. *15th IFAC Symposium on System Identification*, Saint-Malo, France, pp. 1316-1321, 2009.

R. Malti, J. Sabatier and H.Akçay. Thermal modeling and identification of an aluminum rod using fractional calculus. *15th IFAC Symposium on System Identification*, Saint-Malo, France, pp. 958-963, 2009.

Chapter 4

Fractional order modeling and distributed parameter systems

The application of fractional calculus in the modeling and design of automatic control systems is relatively new. In order to model fractional order dynamical systems, some work has been carried out in the last couple of years; but the proposed models and algorithms are still in a preliminary stage of establishment. For infinite dimensional distributed parameter systems (DPS), it is argued that fractional order calculus will play an important role in modeling and analysis. In this chapter, the identification algorithm proposed in Chapter 2 is used to study the benefits of modeling some of the distributed parameter systems using fractional order models. The objective in this work is to represent a distributed parameter systems using parsimonious fractional order transfer function models.

4.1 Introduction

Mathematical models of a dynamic process takes various forms, such as ordinary differential equations (ODE), partial differential equations (PDE), state-space equations, difference equations, etc. It is well known that the transfer functions of systems modeled by ODEs, often called lumped-parameter systems, are rational functions while the irrational transfer functions are the outcome from PDE models for distributed parameter systems (DPS). Consequently, DPS are also called infinite dimensional systems. It is common practice to describe processes using lumped parameter models; however, many important process units are inherently distributed parameter in nature. These processes are described by partial differential equations, where the output variables are function of both time and the coordinate position. The real world is of distributed parameters in nature;

and with growing power of computing technology, the lumped parameter mode of representation of our world is no longer efficient enough for increasingly demanding performance (Chen, 2006). Fractional order models based on fractional order differentiation appear in variational formulation of dissipative systems, in the analysis of diffusive interface, in elasticity theory, in dielectric polarization and in many other areas (Sabatier et al., 2009). Distributed parameter systems where fractional operators appear in the analysis include long transmission lines, heat transfer and diffusive processes in general. Analytical expressions of these systems show the appearance of half integer order derivatives in the transfer function form. Curtain and Morris (2009) presents a good survey on transfer functions of distributed parameter systems.

Fractional order models are superset of integer order models and they exhibit richer behavior in both time and frequency domains. The fractional order derivative of a variable can be viewed as a limit of an infinite series involving integer order derivatives. Consequently, a fractional order model can be viewed as a parsimonious representation of infinite order ODE models. Muddu et al. (2009, 2010) have used the fractional calculus theory for building parsimonious models for the packed bed distillation column.

For highly complex DPS, a closed-form expression of the transfer function of a system cannot be obtained, and consequently the design of controllers may be difficult. In this chapter, fractional order models are developed for capturing the dynamics of high dimensional and distributed parameter systems. The models are built with the aim to reduce the number of parameters to be estimated. The algorithm proposed in Chapter 2 to estimate all the model parameters including time delay for commensurate fractional order models is used to model these systems. The objective is to present examples of systems with dynamics modeled by partial differential equations in one space dimension; all the examples used are single-input single-output.

This chapter is organized as follows. In Section 4.2, the simplest fractional order model and its step response are presented to differentiate its behavior from integer order models. Section 4.3 presents the derivation of the analytical transfer functions obtained for two DPS, namely the thermal diffusion in a wall and diffusion in a semi-infinite slab. FO models along with rational order models are compared for the thermal diffusion wall problem, an industrial scale froth heater and a laboratory scale stirred tank with transportation delay problem, in Section 4.4. Section 4.5 summarizes the major concluding remarks from these application studies.

4.2 Fractional order models

As discussed in Chapter 2, a commensurate transfer function of order γ for a fractional-order time delay system is given as

$$G(s) = \frac{\sum_{j=0}^{m} b_j s^{j\gamma}}{1+\sum_{i=1}^{n} a_i s^{i\gamma}} e^{-Ls} \tag{4.1}$$

where L is the time delay.

Let us consider a simplest fractional order transfer function (with $m = 0$ and $n = 1$) as

$$G_{simp}(s) = \frac{b_0}{1+a_1 s^{\gamma}} e^{-Ls} \tag{4.2}$$

The analytical expression for the step response for this transfer function is:

$$y(t) \;=\left\{ \begin{array}{ll} 0 & \forall \quad t \leq L \\ \frac{b_0}{a_1} t^{\gamma} \sum\limits_{k=0}^{\infty} \frac{\left(\frac{-1}{a_1} t^{\gamma}\right)^k}{\Gamma(\gamma k+k+1)} & \forall \quad t > L \end{array} \right\} \tag{4.3}$$

where $\Gamma(.)$ denotes the gamma function. Fig. 4.1 presents the step response (for unit change in input) for $a_1 = 1$, $b_0 = 1$ and $L = 1$. The different response for different values of γ are clearly visible. According to the final value theorem, the

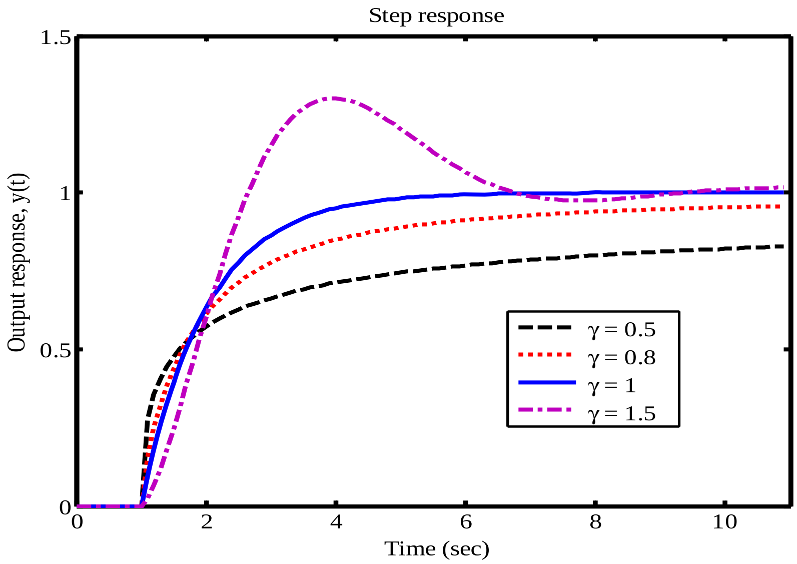


Figure 4.1: Step response for G_{simp} for different values of γ

output response for each γ value will settle at $b_0 = 1$; however, the response is slower for γ value not equal to one. This kind of behavior is often found in some distributed parameter systems where the response reaches steady state after a long period of time because of the irrational nature of the DPS, as will be seen in the examples presented in the next section.

We have used the Oustaloup approximation defined in equation (2.8) for simulating the fractional order operator.

4.3 Distributed parameter systems

4.3.1 Thermal diffusion in a wall

In Chapter 2 we discussed the classical wall problem for simulation study to illustrate the application of the proposed algorithm for a fractional order delay model. In this section we will show with the detailed derivation on how this distributed parameter system presents a fractional order transfer function model. Fig. 2.8 presented in Chapter-2 represents the classical wall problem used to analyze heat transfer.

To demonstrate the fractional behavior of this thermal system, the following assumptions are used: (i) the wall is perfectly isolated; (ii) the wall is initially at ambient temperature, so that there is no thermal exchange with the surroundings; (iii) heat losses on the surface where the thermal flux is applied is neglected. Let us consider again the governing equation for this process as given by the heat diffusion equation:

$$\frac{\partial T(x,t)}{\partial t} = \zeta \frac{\partial^2 T(x,t)}{\partial x^2} \tag{4.4}$$

$$\Omega(x,t) = -\varrho \frac{\partial T(x,t)}{\partial x} \tag{4.5}$$

where $\Omega(x,t)$ is the heat flux ($\frac{W}{m^2}$), $T(x,t)$ is the temperature (^0C), $\zeta = \frac{\varrho}{\rho c}$ is the thermal diffusivity ($\frac{m^2}{s}$), ϱ is the thermal conductivity ($\frac{W}{m\ ^0C}$), ρ is the mass density ($\frac{kg}{m^3}$), and c is the specific heat constant ($\frac{J}{kg\ ^0C}$). The boundary conditions are such that the temperature at face B is kept constant and equal to zero during the overall heating experiment and the external heat is added at face A ($\Omega(0,t) = \Omega_{in}$). The boundary conditions are

$$T(x,0) = 0 \tag{4.6}$$

$$T(l,t) = 0 \tag{4.7}$$

where 'l' is the distance between two walls. If S_A is the cross-sectional area of the wall (m^2) and we define $y(t) = T(x,t)$ and heat input (not flux) as, $u(t) = \Omega(0,t)\ S_A$. Taking Laplace transforms on both sides of equation (4.4) leads to the ordinary differential equation:

$$\pounds\left(\frac{\partial T(x,t)}{\partial t}\right) = \zeta\ \pounds\left(\frac{\partial^2 T(x,t)}{\partial x^2}\right) \tag{4.8}$$

$$s\ T(x,s) - T(x,0) = \zeta\frac{d^2T(x,s)}{dx^2} \tag{4.9}$$

where $\pounds(T(x,t)) = T(x,s)$. Using the first boundary condition in (4.6) we have,

$$\frac{d^2T(x,s)}{dx^2} - \frac{s}{\zeta}\ T(x,s) = 0 \tag{4.10}$$

Solving with respect to x yields:

$$T(x,s) = K_1(s)\ e^{-x\sqrt{\frac{s}{\zeta}}} + K_2(s)\ e^{x\sqrt{\frac{s}{\zeta}}} \tag{4.11}$$

where $K_1(s)$ and $K_2(s)$ are two constants. Now, using the first boundary condition in (4.7),

$$T(l,s) = 0 = K_1(s)\ e^{-l\sqrt{\frac{s}{\zeta}}} + K_2(s)\ e^{l\sqrt{\frac{s}{\zeta}}} \tag{4.12}$$

$$K_1(s)\ = -K_2(s)\ e^{2l\sqrt{\frac{s}{\zeta}}} \tag{4.13}$$

Therefore,

$$\begin{aligned} Y(s) &= T(x,s) = -K_2(s)\ e^{2l\sqrt{\frac{s}{\zeta}}}\ e^{-x\sqrt{\frac{s}{\zeta}}} + K_2(s)\ e^{x\sqrt{\frac{s}{\zeta}}} && (4.14)\\ &= -K_2(s)\left[e^{(2l-x)\sqrt{\frac{s}{\zeta}}} -\ e^{x\sqrt{\frac{s}{\zeta}}}\right] && (4.15)\end{aligned}$$

$$U(s) = S_A\ \Omega(0,s) = -\varrho\ S_A\left[\frac{\partial T(x,s)}{\partial x}\right]_{x=0} \tag{4.16}$$

$$\begin{aligned} U(s) &= -\varrho\ S_A\left[-K_1(s)\ \sqrt{\frac{s}{\zeta}}e^{-x\sqrt{\frac{s}{\zeta}}} + K_2(s)\sqrt{\frac{s}{\zeta}}\ e^{x\sqrt{\frac{s}{\zeta}}}\right]_{x=0} \\ &= \varrho\ S_A\sqrt{\frac{s}{\zeta}}\,[K_1(s) - K_2(s)\ ] \\ &= -\varrho\ S_A\sqrt{\frac{s}{\zeta}}K_2(s)\left[1 + e^{2l\sqrt{\frac{s}{\zeta}}}\ \right] && (4.17)\end{aligned}$$

The transfer function relating $Y(s)$ and $U(s)$ is given as

$$H(s) = \frac{Y(s)}{U(s)} = \frac{1}{\varrho\ S_A\sqrt{\frac{s}{\zeta}}}\frac{e^{(2l-x)\sqrt{\frac{s}{\zeta}}} -\ e^{x\sqrt{\frac{s}{\zeta}}}}{1 + e^{2l\sqrt{\frac{s}{\zeta}}}} \tag{4.18}$$

The exponential function is then replaced by its series expansion:

$$e^z = \sum_{i=0}^{\infty} \frac{z^i}{i!} \tag{4.19}$$

As can be seen from the transfer function in (4.18), this is an irrational transfer function with infinite many poles and zeros. For the case when $y(t) = T(0, t)$, after truncating the infinite series to a finite number of terms, the truncated fractional transfer function $\hat{H}_R(s)$ (Gabano and Poinot, 2011) is given as

$$\hat{H}_R(s) = \frac{1}{S_A} \frac{\sum_{i=0}^{R-1} b_i s^{\frac{i}{2}}}{\sum_{i=0}^{R-1} a_i s^{\frac{i}{2}} + s^{\frac{N}{2}}} \tag{4.20}$$

where the constants are defined as

$$\begin{aligned}
a_0 &= \frac{R!}{2^{R-1}} \left(\frac{\zeta^{\frac{R}{2}}}{l^R} \right) && (4.21) \\
a_i &= \left(\frac{R!}{i! 2^{R-1} l^{R-i}} \right) \zeta^{\frac{R-i}{2}}, \quad 1 \leq i \leq R-1 && (4.22) \\
b_i &= \left(\frac{R!}{(i+1)!\ 2^{R-i-1} l^{R-i-1}} \right) \frac{\zeta^{\frac{R-i}{2}}}{\varrho}, \quad 0 \leq i \leq R-1 && (4.23)
\end{aligned}$$

Therefore, the analytical expression of this system shows the appearance of half integer order derivatives in the transfer functional form. It would require an infinite number of terms in both the numerator and denominator to model this process accurately; and working with a reduced model structure will always result in some modeling errors.

4.3.2 Diffusion in a semi-infinite slab

Diffusion is the net migration (mass transfer-transport) of molecules from regions of high to low concentration. Fick's laws are extensively used as a model for describing tracer diffusion in porous media. For molecular diffusion in 1-D, the Fick's second law is

$$\frac{\partial c(x,t)}{\partial t} = D \frac{\partial^2 c(x,t)}{\partial x^2} \tag{4.24}$$

where, c is the concentration $\left(\frac{mol}{m^3}\right)$, t is time (sec), D is the diffusion coefficient $\left(\frac{m^2}{sec}\right)$, x is the position (m). Using the Fick's first law

$$J(x,t) = -D \frac{\partial c(x,t)}{\partial x} \tag{4.25}$$

J is the diffusion flux $\left(\frac{mol}{m^2 sec}\right)$. The boundary conditions for this semi-infinite slab is considered as

$$\begin{aligned} c(x,0) &= c_\infty \\ As\ x &\rightarrow \infty, c(x,t) = c_\infty \end{aligned} \tag{4.26}$$

If we define a new variable

$$c_1(x,t) = c(x,t) - c_\infty \tag{4.27}$$

then the boundary conditions become

$$c_1(x,0) = 0 \tag{4.28}$$

$$c_1(\infty,t) = 0 \tag{4.29}$$

and

$$\frac{\partial c_1(x,t)}{\partial t} = D\frac{\partial^2 c_1(x,t)}{\partial x^2} \tag{4.30}$$

If we define a system with input defined as flux at x = 0, $J(0,t)$ and the system output as $c(x,t)$ i.e. $u(t) = J(0,t)$ and $y(t) = c_1(x,t)$, then taking Laplace transforms on both sides of equation (4.24) leads to the ordinary differential equation:

$$£\left(\frac{\partial c_1(x,t)}{\partial t}\right) = D£\left(\frac{\partial^2 c_1(x,t)}{\partial x^2}\right) \tag{4.31}$$

$$s\ C_1(x,s) - C_1(x,0) = D\frac{d^2 C_1(x,s)}{dx^2} \tag{4.32}$$

where $£(c_1(x,t)) = C_1(x,s)$ and using the first boundary condition in (4.28),

$$\frac{d^2 C_1(x,s)}{dx^2} - \frac{s}{D}\ C_1(x,s) = 0 \tag{4.33}$$

Solving with respect to x yields:

$$C_1(x,s) = a(s)\ e^{x\sqrt{\frac{s}{D}}} + b(s)\ e^{-x\sqrt{\frac{s}{D}}} \tag{4.34}$$

where $a(s)$ and $b(s)$ are two constants. Based on the second boundary condition in (4.29),

$$a(s) = 0 \tag{4.35}$$

Therefore,

$$C_1(x,s) = b(s)\ e^{-x\sqrt{\frac{s}{D}}} \tag{4.36}$$

and

$$U(s) = J(0,s) = -D\left[\frac{\partial C_1(x,s)}{\partial x}\right]_{x=0} = D\ b(s)\sqrt{\frac{s}{D}}\left[e^{-x\sqrt{\frac{s}{D}}}\right]_{x=0} \tag{4.37}$$

$$u(t) = D\ b(s)\sqrt{\frac{s}{D}} \tag{4.38}$$

The transfer function relating $Y(s)$ and $U(s)$ is given as

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b(s)\ e^{-x\sqrt{\frac{s}{D}}}}{D\ b(s)\sqrt{\frac{s}{D}}} = \frac{1}{\sqrt{s\ D}}e^{-x\sqrt{\frac{s}{D}}} \tag{4.39}$$

where $U(s)$ and $Y(s)$ are the Laplace transform for the input and the output respectively. If we consider heat transfer in a semi-infinite metal rod as a system, we get a transfer function similar to model (4.39) (Malti et al., 2009).

Using the Pth order Padé approximation of e^{-z} (where $z = x\sqrt{\frac{s}{D}}$) yields:

$$e^{-z} \simeq \frac{\sum\limits_{k=0}^{P}\frac{(2P-k)!}{k!(P-k)!}(-z)^k}{\sum\limits_{k=0}^{P}\frac{(2P-k)!}{k!(P-k)!}(z)^k} \tag{4.40}$$

Therefore,

$$H_P(s) = \frac{1}{\sqrt{s\ D}}\frac{\sum\limits_{k=0}^{P}\frac{(2P-k)!}{k!(P-k)!}(-x\sqrt{\frac{s}{D}})^k}{\sum\limits_{k=0}^{P}\frac{(2P-k)!}{k!(P-k)!}(x\sqrt{\frac{s}{D}})^k} \tag{4.41}$$

As can be seen, half integer order derivatives appear in the transfer function form. Again we see that diffusive processes are described with fractional order behavior and it is more appropriate to model these systems using fractional order models. In summary, Distributed parameter systems are ubiquitous and so is fractional order calculus.

4.4 Model identification of DPS

A time delay is also referred to as dead time, transportation lag or distance-velocity lag. Whenever material or energy is physically moved in a process or a plant, there is usually a time delay associated with the movement (Seborg et al., 1989). For the examples considered in this chapter, time delays are present in the systems. In this chapter, for DPS examples with and without delays, we will attempt to find a parsimonious in parameter model using fractional order dynamic models. For comparison purpose, CONTSID (Garnier et al., 2006) toolbox is used to identify continuous-time OE models; and process models (first order or second order plus time delay) are obtained using MATLAB system identification toolbox. For rational models the time delays are estimated using ‘cra’ function is MATLAB prior to estimating other model parameters.

4.4.1 Thermal diffusion in a wall

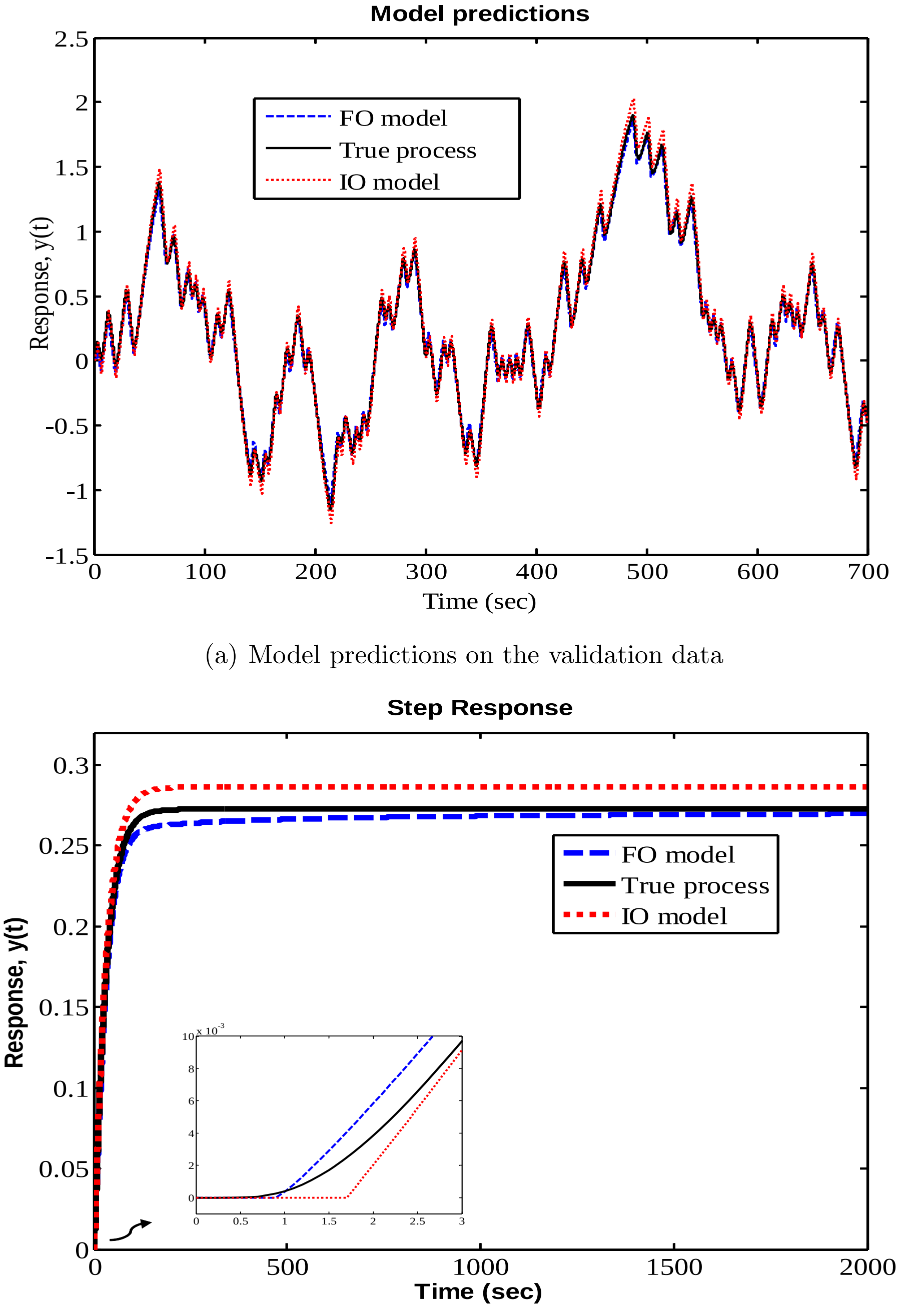
In this section we present the comparison study of the fractional order model with a rational order (integer order) model for the thermal diffusion in a wall (CW). The open-loop perturbation data is used for identifying fractional order and integer order models using an identical data set. For building models, we use the same data that is used in Chapter 2 (see Fig. 2.10) to identify a fractional order plus delay model for the wall problem. The overall data is partitioned into two parts : (a) an identification set: First 8000 data points, and (b) the validation data set: next 7000 data points. The fractional order model (FO model, G_{FO_9}) identified in Chapter 2 using 50 Monte carlo simulation is given as:

$$G_{FO_9}(s) = \frac{0.0273(\pm 0.003)}{46.239(\pm 6.211)s^{2\times(0.585\pm 0.035)} + 2.263(\pm 1.265)s^{0.585(\pm 0.035)} + 1}\, e^{-0.899(\pm 0.058)s} \tag{4.42}$$

A continuous-time rational output error (OE) model (IO model, G_{IO_9}) is used for the identification purpose. For a fair comparison, a second order model with delay is identified with the same number of parameters (= 5) as the fractional model $G_{FO_9}(s)$. The ‘coe’ function of the CONTSID toolbox is used and the model obtained using the average value of the parameters is :

$$G_{IO_9}(s) = \frac{0.00067(\pm 0.000002)s + 0.000278(\pm 0.000003)}{s^2 + 0.3228(\pm 0.00399)s + 0.00971(\pm 0.00013)}\, e^{-1.7s} \tag{4.43}$$

The time delay for $G_{IO_9}(s)$ was estimated explicitly using the correlation analysis, function ‘cra’ in MATLAB while the other parameters were obtained from the CONTSID toolbox. It is not possible to show the predictions from all the 50 models; so only predictions from the average model is presented in Fig. 4.2. It shows the model predictions (infinite step ahead predictions) of $G_{FO_9}(s)$ and $G_{IO_9}(s)$ and the process output for one realization on the validation data set. It is not easy to differentiate which model provides better predictions but if we see the step responses for the two models and compare them with the true step response, we see that the FO model captures both the delay and the gain part better than the rational model. In order to compare two average models, we define a criterion



(a) Model predictions on the validation data

(b) Step responses of the identified models

Figure 4.2: Model validation for $G_{FO_9}(s)$ and $G_{IO_9}(s)$ models for the CW example

known as percent prediction error (PPE) when the model is used on a validation data set as:

$$PPE = \frac{\sum_{k=1}^{N_S}[\mathbf{y}(k) - \hat{y}(k)]^2}{\sum_{k=1}^{N_S}[\mathbf{y}(k) - \bar{y}(k)]^2} \times 100 \tag{4.44}$$

where $\overline{y}$ here represents the mean value of data $\{y(k)\}$ and $\widehat{y}(k)$ represents the predicted value of $y(k)$. A model with a lower value of PPE is a better model.

Table 4.1 presents the percent prediction error (PPE) for the two models. As can be seen from Table 4.1, the fractional order model provides lower value of PPE compared to the integer order model. In Section 4.3.1, we had shown that this process is described by fractional order behavior. As can be seen from the results fractional order models do indeed provide better representation of the process compared to rational (integer) order models. Usually, high order rational models are necessary to represent a fractional order system to obtain comparable results. We can also add one more pole and add no zeros (total five parameters) to see if this form of rational model provides any better predictions.

Table 4.1: PPE values for models for CW example

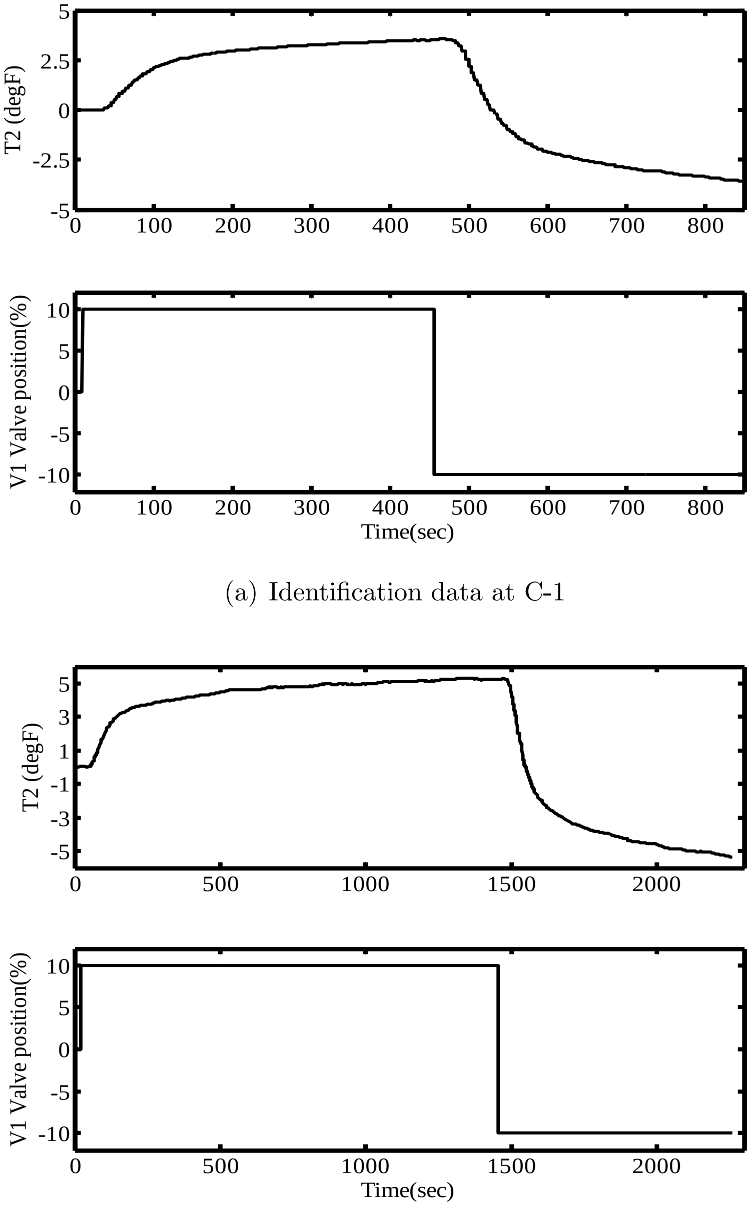
	PPE	# of parameters
G_{FO_9}	0.414	5
G_{IO_9}	0.474	5

4.4.2 Industrial froth heater process

A heat exchanger is a pertinent example of distributed parameter system. The temperature of the fluid flowing through the heat exchanger which is being heated by the steam or any heating fluid is distributed in nature. The temperature of the fluid is a function of time as well as the distance from the fluid inlet.

Here we present the data from an industrial froth heater (FHC) process which is an interstage between primary and secondary extraction units in the oil sands industry. The desired objective of the heater (or heat exchanger) is to increase the temperature of the incoming bitumen froth before it is sent to the next stage of the process. High pressure steam is used to heat the froth. The process has been described in detail in Chapter 6 where the objective is to design a MPC controller.

Here we present real industrial data which was obtained by performing the controlled identification test on the setup for only one of the heat exchangers. The outlet temperature from the first heat exchanger (T_2) is measured (process output) and the steam valve position (V_1 valve position) is used as the manipulating variable (process input) for the study presented here. Again, we present a comparative study of fractional order versus integer order models for this DPS. Fig. 4.3 presents the step response data at two different operating conditions (C-1 and C-2) for the froth heater. The process dynamics are different at two operating



(a) Identification data at C-1

(b) Identification data at C-2

Figure 4.3: Model validation for fractional order and rational order models

conditions. Again, continuous-time rational output error (OE) models are used for the comparison purpose based on the data presented in Figure 4.3. Since the data length is not enough, 100% of the data was used for identification purpose. A process model using prediction error method ($G_{IO_{10}^{(1)}}(s)$) using the MATLAB system identification toolbox is estimated along with OE models. A first order OE model ($G_{IO_{10}^{(2)}}(s)$) as well as a second order OE model ($G_{IO_{10}^{(3)}}(s)$) are computed for comparative study. The 'coe' function of the CONTSID toolbox is used to estimate continuous-time OE models. This function is different from MATLAB system identification toolbox function as 'coe' is a continuous time model identification method and is based on the linear filtering algorithm.

Table 4.2: Models at two operating conditions, C-1 and C-2 for FHC

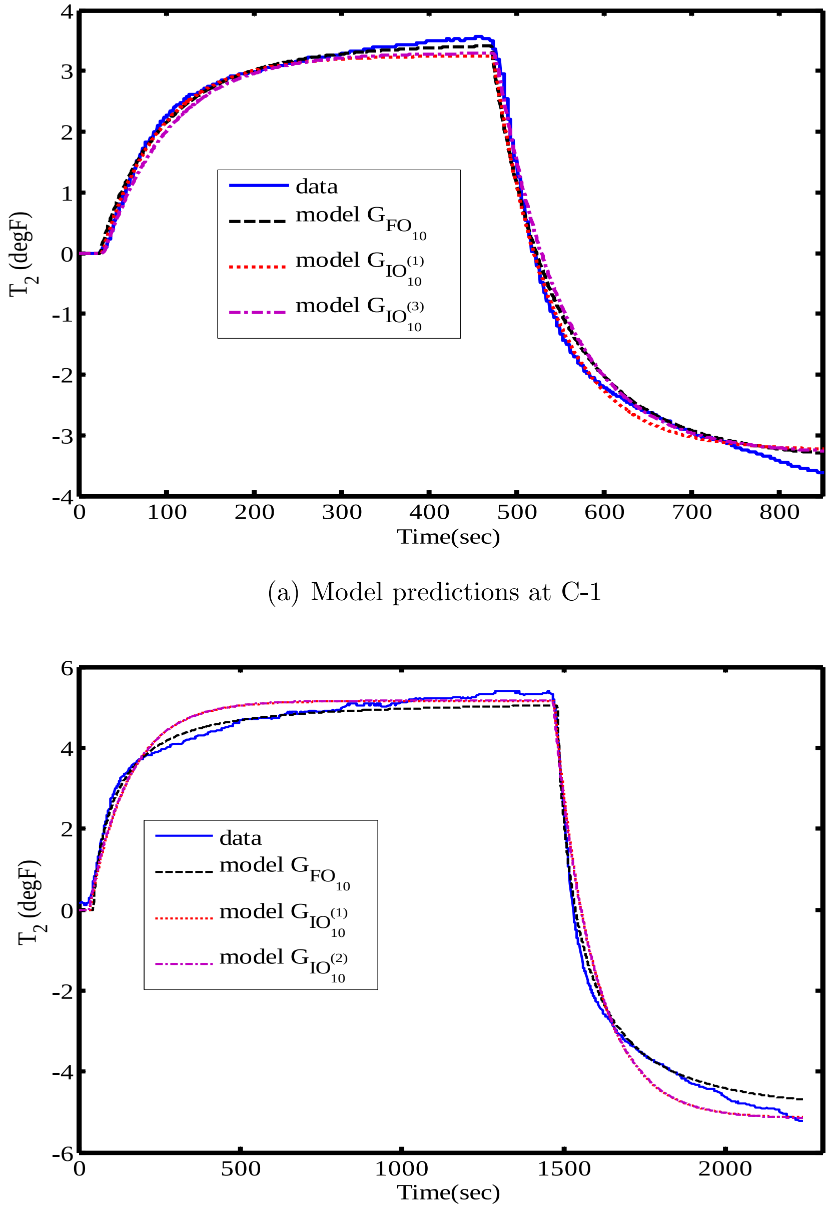
	C-1	C-2
$G_{FO_{10}}(s)$	$\frac{0.353}{58.76s^{0.93}\ +\ 1}e^{-24.94s}$	$\frac{0.533}{32.34s^{0.758}\ +\ 1}e^{-45.82s}$
$G_{IO_{10}^{(1)}}(s)$	$\frac{0.325}{66.9s\ +\ 1}e^{-27s}$	$\frac{0.515}{121.9s\ +\ 1}e^{-32s}$
$G_{IO_{10}^{(2)}}(s)$	$\frac{0.0042}{s\ +\ 0.01266}e^{-27s}$	$\frac{0.0041}{s\ +\ 0.0080}e^{-32s}$
$G_{IO_{10}^{(3)}}(s)$	$\frac{0.00162}{s^2\ +\ 0.3833s\ +\ 0.00491}e^{-27s}$	$\frac{2.45\text{x}10^9}{s^2\ +\ 5.92\text{x}10^{11}s\ +\ 4.75\text{x}10^9}e^{-32s}$

The time delay for integer order models were estimated explicitly using the correlation analysis, function 'cra' in MATLAB. For the fractional order model, the time delay was estimated simultaneously with the other model parameters using the algorithm proposed in Chapter 2. Infinite step ahead predictions for all the models are shown in Figure 4.4. As can be seen the fractional order model captures the process dynamics really well.

Table 4.3 presents the PPE values for all the models. As can be seen from Table 4.3, fractional order model provides a lower value of PPE compared to the integer order models and thus we can say that the fractional order model with delay ($G_{FO_{10}}(s)$) provides a parsimonious parameter model for this process.

4.4.3 Continuous stirred tank heater process

We discussed a continuous stirred tank heating (CSTH) process in Chapter 3. The same process is used again to model the process as a fractional order system. If the temperature inside the stirred tank is assumed to be uniform, the CSTH



(a) Model predictions at C-1

(b) Model predictions at C-2

Figure 4.4: Model validation for fractional order and rational order models

can be modeled as a simple lumped parameter system using integer order models; however, for CSTH, the outlet water from the tank flows through a long pipeline and a number of thermocouples are placed at different distances in this pipe to introduce time delay into the system. If radial heat losses are assumed negligible and distributed nature of flow inside the pipe is ignored, then flow through a pipe is assumed to be modeled as a pure time delay system as used in the experimental

Table 4.3: PPE values for models for FHC example

	PPE		# of parameters
	C-1	C-2	C-1/C-2
$G_{FO_{10}}(s)$	0.351	0.285	4
$G_{IO_{10}^{(1)}}(s)$	0.509	0.904	3
$G_{IO_{10}^{(2)}}(s)$	0.493	0.863	3
$G_{IO_{10}^{(3)}}(s)$	0.491	0.846	4

validation in Chapter 3. However, for this study some modifications are made in the experiments to enhance the distributed flow dynamics inside the pipe, which includes slower flow rates and use of an air fan outside the pipe to promote heat losses. The combined tank and pipe flow process for this study is assumed to be modeled using a fractional order model with a delay term.

Again, the steam flow rate to the tank is used as the manipulated variable and the outlet water temperature at the far end of the pipeline is used as the controlled variable. The data presented in Figure 4.5 is used to build and validate models. Again, we present the comparison study of the fractional order model with a rational order (integer order) model for this DPS. For comparison purpose, process

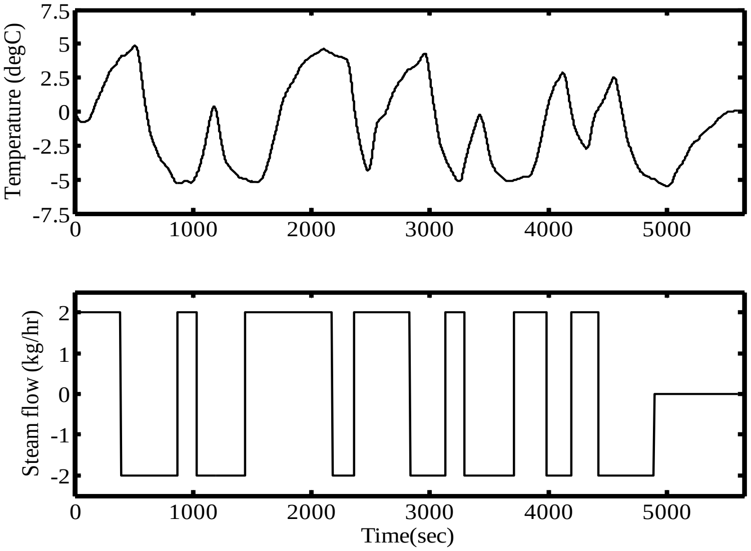


Figure 4.5: Time response for CSTH process

models ($G_{IO_{11}^{(1)}}(s)$, $G_{IO_{11}^{(2)}}(s)$) using the prediction error method in MATLAB and $G_{IO_{11}^{(3)}}(s)$ is estimated using CONTSID. The overall data as shown in Fig. 4.5 was partitioned into two parts : (a) an identification set: comprising of the first 3000 data points, and (b) the validation data set: comprising of all data points. Table 4.4 presents the identified models for this process and their corresponding PPE values.

Table 4.4: Identified models and PPE values for CSTH

	Models	# of parameters	PPE
$G_{FO_{11}}(s)$	$\frac{2.75}{229.7s^{1.065}+1}e^{-137.5s}$	4	14.80
$G_{IO_{11}^{(1)}}(s)$	$\frac{2.458}{148.97s+1}e^{-135s}$	3	15.37
$G_{IO_{11}^{(2)}}(s)$	$\frac{2.338}{2183.3s^2+140.87s+1}e^{-135s}$	4	15.30
$G_{IO_{11}^{(3)}}(s)$	$\frac{0.00228}{s^2+0.133s+0.000875}e^{-135s}$	4	14.73

The time delay for integer order models were again estimated using the correlation analysis, via the function 'cra' in MATLAB; however, for the fractional order model, the time delay was estimated simultaneously with the other model parameters using the algorithm proposed in Chapter 2. Infinite step ahead predictions for all models are shown in Figure 4.6. As can be seen, all models including the fractional order model capture the process dynamics equally well. As can be seen from Table 4.3, the fractional order model provides slightly lower value of PPE compared to the integer order models, $G_{IO_{11}^{(1)}}(s)$ and $G_{IO_{11}^{(2)}}(s)$; however the advantage of using fractional derivative model is not very prominent in this example as the process can be modeled easily via linear differential equations. Even an integer order model ($G_{IO_{11}^{(3)}}(s)$) provides comparable PPE using the same number of parameters. We need to make conditions more favorable to enhance the distributed nature of the pipe flow for further model testing. This is something which could be explored in the future.

4.5 Conclusions

In this chapter, we used the identification method proposed for commensurate fractional order models with time delay in Chapter 2 and applied it to a number

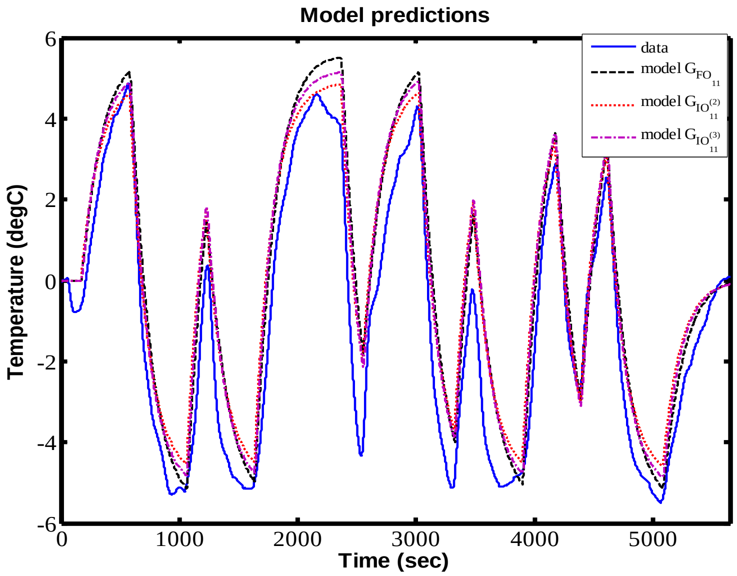


Figure 4.6: Model predictions for CSTH process

of distributed parameter systems. Both simulation and experimental studies were carried out to build fractional order models. The DPS are treated as black boxes and both fractional order as well as rational order (integer) models are identified from input-output perturbation data. For some cases, fractional order models with time delay were found to be very effective in dealing with distributed parameter systems, especially the diffusive interface problem. In the future, the interesting perspective would be to carry experimental evaluation of the proposed method to diffusive interface examples such as heat conduction in a wall and metal rod. Also, if an identification algorithm for non-commensurate fractional order models is proposed, applying the same on DPS could provide additional flexibility.

Bibliography

Yangquan Chen. Ubiquitous fractional order controllers. *In the second IFAC Symposium on Fractional Derivatives and Applications*, Portugal, 2006.

J. Sabatier, O. P. Agrawal and J. A. T. Machado. Advances in Fractional Calculus: Theoretical Developments and Applications in Physics and Engineering. *Springer Publishing Company*, 1st edition, 2007.

R. Curtain and K. Morris. Transfer functions of distributed parameter systems: A tutorial. *Automatica*, vol. 45(5), pp. 1101-1116, 2009.

M. Muddu, A. Narang and Sachin C. Patwardhan. Development of ARX models for predictive control using fractional order and orthonormal basis filter parametrization. *Ind. Eng. Chem. Res.*, vol. 48(19), pp. 8966-8979, 2009.

M. Muddu, A. Narang and Sachin C. Patwardhan. Reparametrized ARX models for predictive control of staged and packed bed distillation columns. *Control Engineering Practice*, vol. 18(2), pp. 114-130, 2010.

J.D. Gabano and T. Poinot. Estimation of thermal parameters using fractional modelling. *Signal Processing*, vol. 91(4), pp. 938 - 948, 2011.

R. Malti, J. Sabatier and H.Akçay. Thermal modeling and identification of an aluminum rod using fractional calculus. *15th IFAC Symposium on System Identification*, Saint-Malo, France, pp. 958-963, 2009

H. Garnier, M. Gilson, and O. Cervellin. Latest developments for the matlab contsid toolbox. *IFAC Symposium on System Identification, SYSID 2006*, Newcastle, Australia, 2006.

D. Seborg, T. Edgar and D. Mellichamp. Process Dynamics and Control. *John Wiley & Sons*, New York, 1989.

Chapter 5

Model based predictive controllers design for interface level regulation in oil sands separation cells [1]

Large-scale separation cells, used in the primary extraction in the oil sands industry, are integral parts of the overall process of bitumen extraction. Good regulation of the interface level between the bitumen froth and the middlings in these cells can result in a significant improvement in bitumen recovery and throughput and heavily influence process economics. This chapter details a case study application of identification and design of a model based predictive controllers for the separation cell process. IMC and MPC schemes using linear models are designed, implemented and tested in real time on the industrial separation cell. The testing result shows that both IMC and MPC schemes provide significant benefits over the current operations which uses a PID controller. The benefits include significant reduction in the variance of the interface level and underflow pump movement, resulting in higher bitumen recovery, smoother operations downstream and pump energy savings.

[1] Industrial case study-1: The results presented in this chapter has been submitted to Control Engineering Practice. A condensed version of this chapter has been accepted for presentation at American Control conference, ACC2012. A condensed version of this chapter has also been submitted to International Mineral Processing Conference, IMPC' 2012. First part of this chapter was presented at 60th Canadian chemical engineering conference (CSCHE 2010), held at Saskatoon, Saskatchewan, Canada on October 24 27, 2010

5.1 Introduction

Oil sands mainly contain mineral solids (silica sands and clays), water, and bitumen, a highly viscous petroleum-like hydrocarbon. The Athabasca oil sands reserves in Northern Alberta, Canada, has huge deposits of oil and makes Canada the second largest oil reserves only after the Saudi Arabia (Alberta-Energy, 2011).

A majority of the oil sands industries that use surface mining to prepare the feed uses hot water based extraction to extract bitumen from the oil sands. One of the earliest papers in this regard was Clark (1944). The economic benefits from the extraction of oil from oil sand deposits in Alberta are huge (Dougan and McDowell, 1997) and with such huge quantities of material being processed in an oil sands extraction operation, even a small improvement in process efficiency and operation can greatly influence bitumen recovery, reduce environmental impact and influence process economics. The extraction process is the next step after mining and most of the variability in the oil sands feed is channeled to the extraction process. There is tremendous scope for improvement in the way advanced process control is used in oil sands processing; and the extraction process is an ideal candidate for the application of dynamic model based control (Dougan and McDowell, 1997).

The extraction process consists of large gravity vessels known as primary separation vessels (PSV) or just separation cells (SepCell). Inside the cell three layers are formed due to density of the slurry mixture; the interface level between the top layer (bitumen froth) and the middle layer (middlings) is particularly important as it influences froth quality and process economics. In the oil sands industry, control of this interface at an optimum level to improve bitumen recovery is of particular interest to the operating engineers. However, interface level control in separation cells presents a lot of challenges for controller design. Modeling of these separation cells to control the interface level based on the physical laws is difficult not only because of the constantly changing nature of the oil sands entering the plant (disturbances in the feed quality and quantity), but also because of the lack of plant information due to lack of available measurement systems which can reliably track the various process and disturbance variables. In the last few years, reliable estimation of the interface level has been obtained from an image based soft sensor. Recent development in the area of soft sensors and their industrial implementation includes Jampana et al. (2010); Fortuna et al. (2006); Domlan et al. (2011). Using the image based soft sensor developed in Jampana et al. (2010, 2011), interface level estimation is currently available every second and may be used to design a suitable model based predictive control scheme.

The complex flow dynamics, because of the multi-phase flow inside the separation cell, presents a big challenge. Inside the SepCell, the interface dynamics are distributed in nature because of the presence of three layers. Modeling of the interface dynamics is non-trivial. The phenomenon is highly nonlinear and fundamental understanding of the interface level is a multidisciplinary area that overlaps with colloid science and surface chemistry, and requires understanding of the interfacial properties of oil/water and water/sand grain layers at the molecular level. In contrast to this, building a data based input-output model offers a relatively simple way to design a model based control scheme for such processes. Even though some of the measurements of flow and densities are available for mass balance calculations and validation, since there is a lot of wear and tear due to nature of the slurry that is being transported, the quality of these measurements are not reliable and depends heavily on maintenance programs. Therefore, the operating strategy for the separation cell efficiency is totally dependent upon the quality of the information used to build models and the control design schemes which use these models. For this study, linear multiple models are built at different operating conditions to model the SepCell.

It is important to understand the process and instrumentation constraints to have a successful application of advanced model based control schemes. The separation cell process has large and variable time delays, and in addition to this, because of the disturbances upstream and downstream of the process, the process gain and delay vary significantly. A varying feed flow (disturbance) because of improper design at the upstream, is one shortcoming commonly found in the mining and oil sands industries. The ore quality, which translates into the amount of bitumen and sand grain particles present in the oil sands slurry being carried to the SepCell, has a direct effect on the process gain between the interface level and the pump speed. Also, if there is too much back pressure in the underflow line which occurs at one operating condition, it becomes difficult to pump heavy sands and it adds up as a significant dead time for the interface level dynamics. Because of these limitations it is important to have multiple models at different operating conditions because the interface level dynamics change very quickly with changes in feed conditions (upstream) as well as downstream changes. If there is too much turbidity or the fuzziness due to the presence of too many fine particles, the interface level is generally not well defined and measurements from the image based soft sensors are not available. Sometimes the operator workload and behavior also has an effect on process operations and this clearly does not offer favorable conditions to reap automation benefits. A recent paper by Bergh and Yianatos (2011) highlights some of the challenges for multivariable predictive control in the mineral processing industry. The challenges discussed in Bergh and Yianatos (2011) are also applicable in the oil sands industry. Model-based control, or more specifically MPC, is appropriate for these highly complex processes. However, the conditions in the oil sands processing plant are not commensurate for a regular MPC application due to the challenges discussed above. Such issues are not commonly encountered in the refining and petro-chemical industries where MPCs is now commonly employed.

The widespread adoption of MPC methods in the process industry is a clear indication of its success and these ideas are now starting to attract interest in other process industries as well. For single-input single-output (SISO) systems, internal model control (IMC) proposed in Rivera et al. (1986) archives good robustness and predictive controller capabilities. However, it cannot explicitly handle constraints. In contrast to any other controllers, MPC computes optimal control moves by solving an optimization problem over a finite horizon, taking into account dynamic behavior of the process as well as the operational constraints (Qin and Badgwell, 2003; Rawlings, 2000). MPC techniques have been explored and implemented in various forms, some of which have found their way to the market place. The review by Qin and Badgwell (Qin and Badgwell, 2003) describes some of these implementation algorithms. Some of the recent industrial MPC applications include applications as reported in Stanler et al. (2011); O'Brien et al. (2011). Even though most industrial processes are nonlinear, locally linear models are often sufficient to approximate a process around single set point. Also, since linear models simplify the control design, they are easy to work with when the control involves rejection of disturbances. Industrial application of MPC rely mainly on linear empirical models obtained by employing time series analysis (Qin and Badgwell, 2003). Linear time series model development from operating data is a well researched area and a wide variety of model structures are available for capturing the dynamics of a system with respect to known inputs and unmeasured disturbances (Ljung, 1999; Garnier et al., 2008).

The main objective of this work is to assess the benefits of using advanced process control algorithms to regulate the interface level in an oil sands separation cell. Model based predictive controllers, IMC for SISO design and MPC controllers for SISO as well as Multi-input single-output (MISO) processes, are designed for the interface level control and experimentally evaluated on a separation cell. The online testing was carried out at the primary extraction unit at Suncor Energy Inc.'s, operation in Fort McMurray, Alberta, Canada. The objective of the control is to regulate the interface level when subjected to process feed rate changes, and provide reduction in the variance of the main controlled variable, the forth bitumen and middlings interface, and thereby improve bitumen recovery and also save on pump energy with smoother operations downstream. Both feedback and feedback plus feed forward control are applied to operate the process dynamically using MPC.

The organization of this chapter is as follows: Section 5.2 provides a brief description of the investigated process. The model identification approach to build reduced order linear models for control purpose is described in Section 5.3. The formulation of the control algorithms used for testing on the process is described in Section 5.4. The closed loop results for the on-line testing on the SepCell are presented in Section 5.5. Section 5.6 presents the economic benefits followed by concluding remarks in Section 5.7.

5.2 Process description

An overview of an industrial hot water based extraction process can be found in Masliyah et al. (2004); Dougan and McDowell (1997). The extraction process flowsheet at Suncor Energy Inc., Fort McMurray, is shown in Fig. 5.1.

The surface mining of the oil sands is done using big trucks and shovels at the mining site. The mined oil sand lumps are crushed, moved on to the conveyor and then mixed with process water in slurry boxes, stirred tanks, and rotary breakers. This forms the oil sands slurry which is send to hydro-transportation pipelines or to tumblers, where the oil sand lumps are sheared, and reduced in size. Some chemical additives are also added during the slurry preparation stage. Within these hydro-transport pipelines, bitumen is liberated from the sand grains and entrained air attaches to the bitumen particles. The flow from these hydro-transport pipelines is transferred to large gravity separation vessels, normally referred to as separation cells (SepCell), as shown in Fig. 5.1. Inside these cell the slurry separates into three different layers: 1) primary bitumen froth, 2) middlings and 3) underflow (also known as tailings). Typically, a 60^0C or 65^0C slurry temperature is used in the current operations. Hot process water is also added to the top of the SepCell to enhance the extraction process and cold water is added into the cone section of the SepCell to help with the underflow flow. The aerated bitumen floats on the top

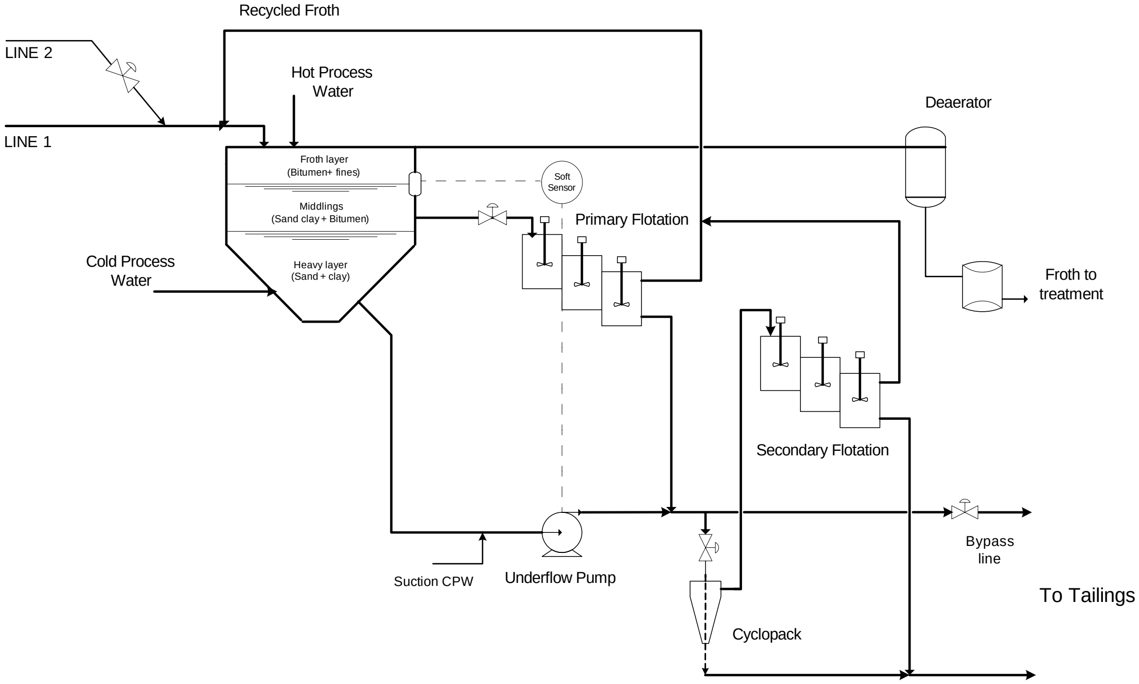


Figure 5.1: Flowsheet of the separation cell process

and is subsequently skimmed off from the slurry. In this layer very small quantities of fine sand particles called fines can also be present. The bitumen froth recovered is then de-aerated, and transported to the later stages of the process for upgrading to synthetic crude oil. The second layer known as middlings mainly contains a high percentage of sand and fines, along with considerable quantities of bitumen. The last layer is the heavier part and is made of sand, clay and water. Small amount of bitumen droplets (usually un-aerated bitumen) remaining in the slurry (in the middlings and underflow stream) are further recovered using induced air flotation in mechanical flotation cells (primary and secondary flotation units) and cyclones separators. The recovered bitumen is recycled and added to the oil sands feed line going into the separation cell. The heavier product from the mechanical flotation cells and cyclones separators are directly rejected into the tailings ponds. More details of the extraction process can be found in Masliyah et al. (2004). Fig. 5.2 presents the view of the industrial separation cell where the bitumen froth is overflowing from the top.

In the extractions process, the interface level between the bitumen-froth and the middlings is known to affect bitumen froth quality and thus heavily influences process economics. For example, when this level is too high, fines (fine sand parti-

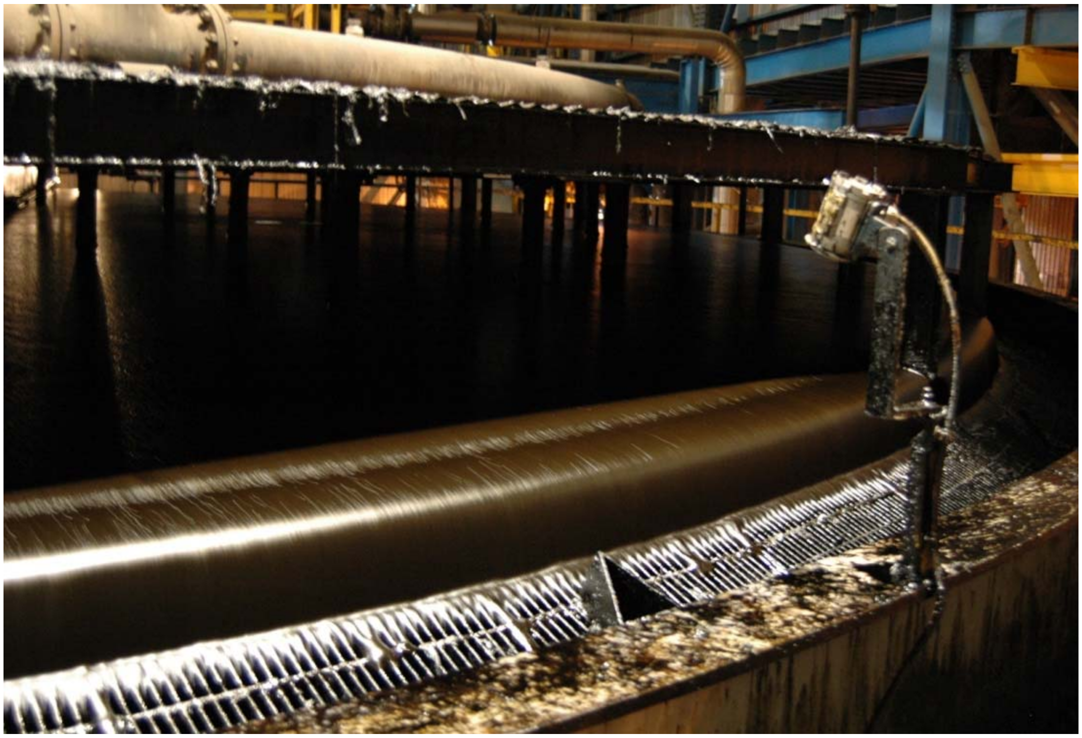


Figure 5.2: Top view of the industrial separation cell at Suncor Energy Inc. The lighter bitumen froth overflows from the top of the separation cell and transported for further processing.

cles) escape with the bitumen-froth, degrading its quality and adversely affecting the downstream upgrading process; and when the level is too low, bitumen is lost to the tailings ponds, causing financial losses and environmental problems. For these reasons, there has been much interest in the oil sands industry to control this interface at an optimum level for good bitumen recovery. Presently, an image based soft sensor developed in Jampana et al. (2011), is used to detect and measure the froth bitumen and middlings interface by using the image obtained from 3 sight-glass windows as shown in Fig. 5.3. The image from the sight glass window is processed in the image based soft sensor algorithm (Jampana et al., 2011) and an estimate of the interface level is used for inferential control.

The process is complex and highly dynamic in nature and due to the large volumes being processed in the mining operation, most of the oil sand variability is propagated to the froth bitumen and middlings in the separation cell. Good model based control of the interface level between the bitumen froth and the middlings in these cells can result in a significant improvement in bitumen recovery and have a favorable impact on process economics. Stabilization of the interface level results

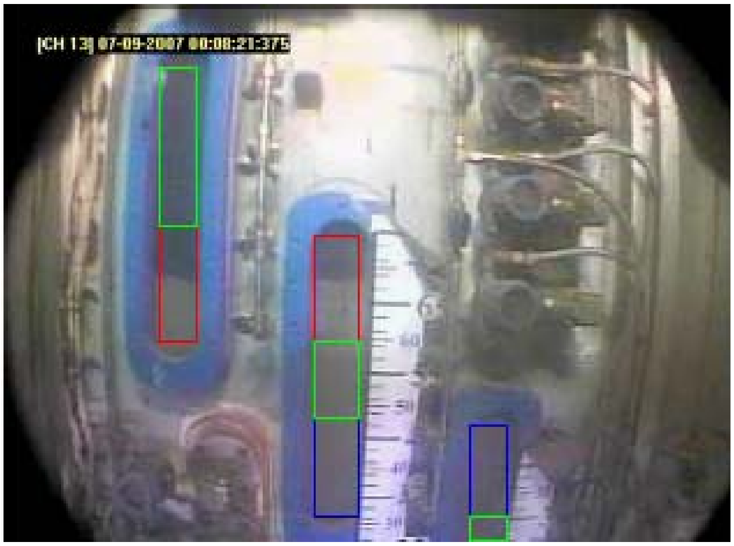


Figure 5.3: Separation cell: Sight glass window showing the interface between the Bitumen froth (dark surface) and the Middlings (light surface). For each sight glass window, the green portion is the non-overlapping part with the remaining glasses, the red boundary shows the overlap between the first and second glasses and the blue boundary shows the overlap between the second and third glasses.

in much smoother operation of the separation vessel and provides conditions that are more favorable for froth treatment and bitumen recovery from middlings and the underflow.

5.3 Model development for Separation Cells

Model based advanced control methods applied to the extraction process could reduce the variability in plant operating conditions which could consequently lead to much smoother plant operations downstream while maintaining safety margins. Also, these techniques could improve plant efficiency by compensating for external disturbances that separation cells are routinely subjected to. However, modeling these processes is a big challenge.

In extraction, some of the physical and chemical mechanisms are not completely understood. Building models based on physical laws such as mass and energy balances is difficult due to multi-phase or slurry flows in the large gravity separation vessel. In the SepCell, the aerated bitumen floats to the top as froth while heavy sands settle at the bottom of the cell due to gravity. Thus, the distributed nature of the flow inside the SepCell makes it difficult to model the process based on first principles model using physical laws. The dynamics of the interface level between the bitumen froth and the middlings is governed by a lot of factors. These include:

1. **Feed composition** (or feed quality): If the oil sands feed contains higher content of fines, due to hindered settling, it takes longer time for heavier coarse material to settle. Similarly, if the fines are low, the settling is much quicker. This influences the dynamics between the interface level and the underflow flow.

2. **Rate of settling** of sands from the Bitumen froth to the Middlings layer: The aerated bitumen separates from lighter middlings and heavy components by gravity. So, the rate at which heavy components settle at the bottom also influences the interface level between the bitumen froth and the middlings.

3. **Flow rate of oil sands feed**: Fluctuations in the oil sands feed flow entering the SepCell causes fluctuations in the interface level. The feed flow also includes the **recycle flow**, which is added to the oil sands feed before it enters the SepCell.

4. **Middlings flow rate**: The middlings stream is one of the outflows from the SepCell. If there is more middlings output from the SepCell, the interface level would go down; however, the middling flow draw is a gravity flow, hence the influence on the interface level would be less when the interface level is low.

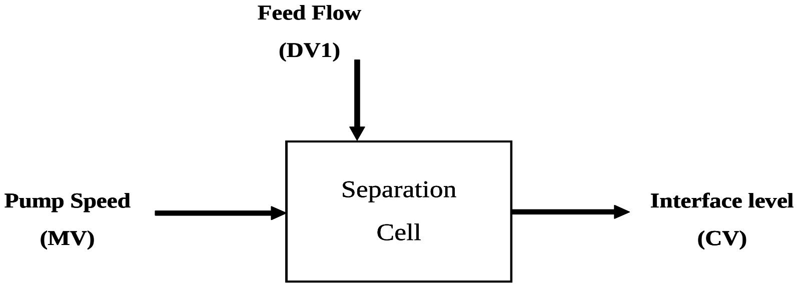
5. **Underflow flow rate**: The underflow flow rate is varied by varying the pump speed and for normal operation this pump speed is used as a manipulating variable to control the interface level between the bitumen froth and the middlings in the SepCell.

Since the objective is to design a model based predictive controllers, building data based input-output models is the best way to model SepCell. To build an accurate model using input-output data, it is necessary to excite the plant and collect a rich data set. In this respect model identification is a costly affair as it requires the plant to be perturbed for sufficient period of time and during this period the bitumen yield may be poor leading to poor recovery. To balance the requirements for good signal to noise ratio and yet have the plant run at normal operating conditions, the plant excitation is kept to a minimum level. To start off, a simple step change is first conducted followed by rigorous excitation, with more information gained at each step of this exercise. The controlled plant tests were done to include a sufficiently wide operating range, while still remaining in the safe operation. To design a model based controller, system identification tools (Ljung, 1999) were used to identify a linear model relating the controlled variable, the interface level, with the manipulating variable, the pump speed. The image-based soft sensor proposed by Jampana et al. (2011) is used to obtain an estimate of the interface level which is the controlled variable.

5.3.1 Identification data

The operating conditions inside the SepCell differ mainly because of the feed quality and hence the process has to be operated at different regimes or zones. In addition, the process dynamics also change in terms of the amount of the oil sands fed being hydro-transported to the SepCell. Relative to the quantity of feed being feed to the SepCell, the operating condition do not change for long periods of time. In this respect a majority of the plant operation is carried out in either one of two operating conditions: 1) operating condition-1: when either LINE-1 or LINE-2 are operational and 2) operating condition-2: when LINE-1 and LINE-2 both are operational. Favorable conditions downstream helps process operation inside the SepCell and at operating condition-1, the bypass line in the downstream side is closed (see Fig. 1) while it is open at operating condition-2 to ensure there is not too much back pressure. Therefore the main objective of this study was to build linear models at these two different operating conditions.

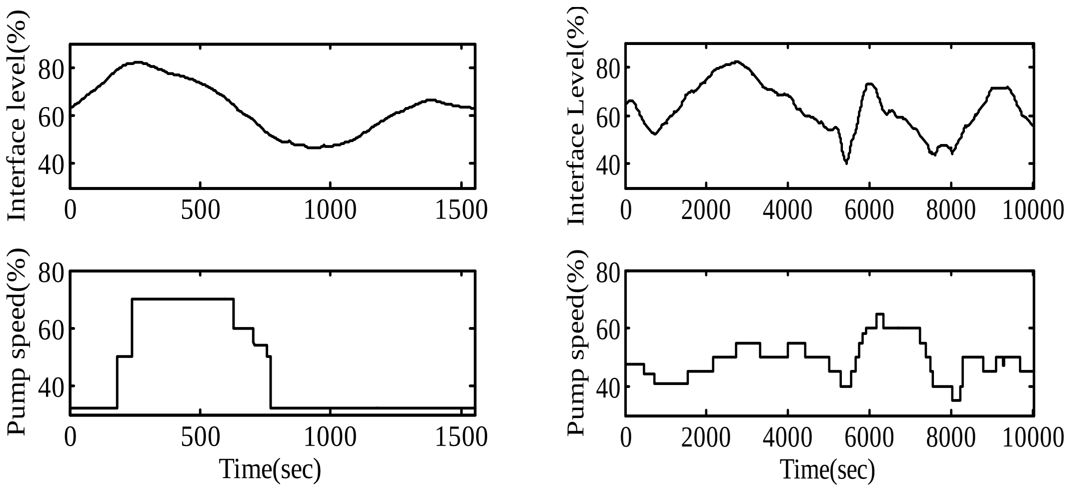
Fig. 5.4 shows the process structure for the SISO control design where the control variable (CV) is the interface level and the manipulating variable (MV) is the pump speed. Feed flow (DV1) represents a measured disturbance which can be used as feed forward disturbance for the feedforward component of the feedback plus feedforward SISO MPC scheme. Open loop experiments were performed with step signals of various step sizes, to collect a rich data set. Fig. 5.5 presents the time-series data from the step test at the operating condition-1. As can be seen, the data quality is not very good, due to the fuzzy image characteristics inside the cell. There is therefore a loss of interface measurement as evident from the flat line on the interface level sensor plot (see right plot in Fig. 5.5) over the sampling instant period from 9000-9500. If there is a loss of interface, the measurement is held at the last value. Fig. 5.6 presents the plot for step test at operating condition-2.



Target requirement: Good regulatory performance

Constraints: Minimum pump movement and interface level between operational range.

Figure 5.4: Process structure for SISO control design



(a) Step changes

(b) Square pulse changes

Figure 5.5: Interface level response to changes in the pump speed at operating condition-1

5.3.2 Model identification

Due to the size of these SepCells and the distributed nature of the flow to and from the cell, time delays are present in the process and the process behaves as a higher order system. Apparent time delays also appear in the identification exercise when such a higher order process is modeled by a low order model to design a reduced-complexity model based controller. The input-output response data presented above is used with the MATLAB system identification toolbox and CONTSID toolbox (Garnier et al., 2006) to identify a number of best fit linear parametric models.

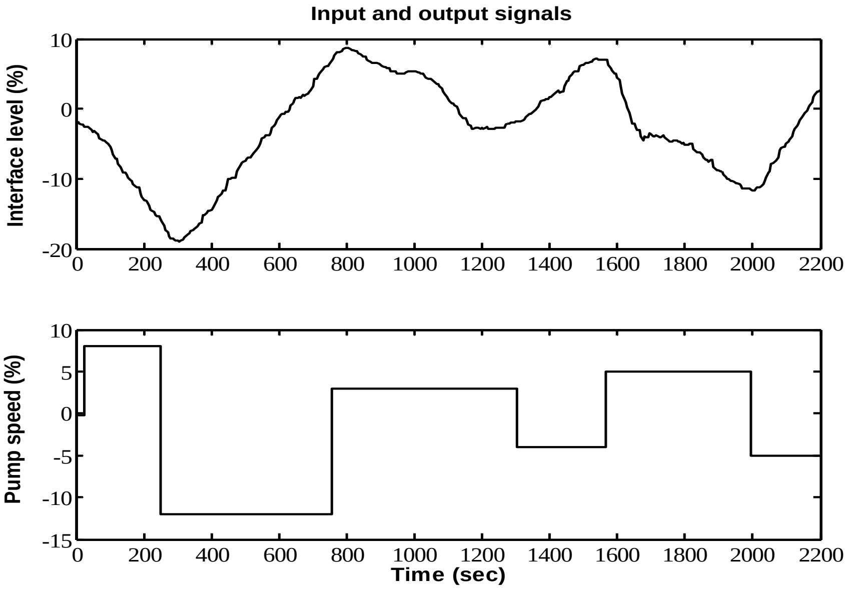
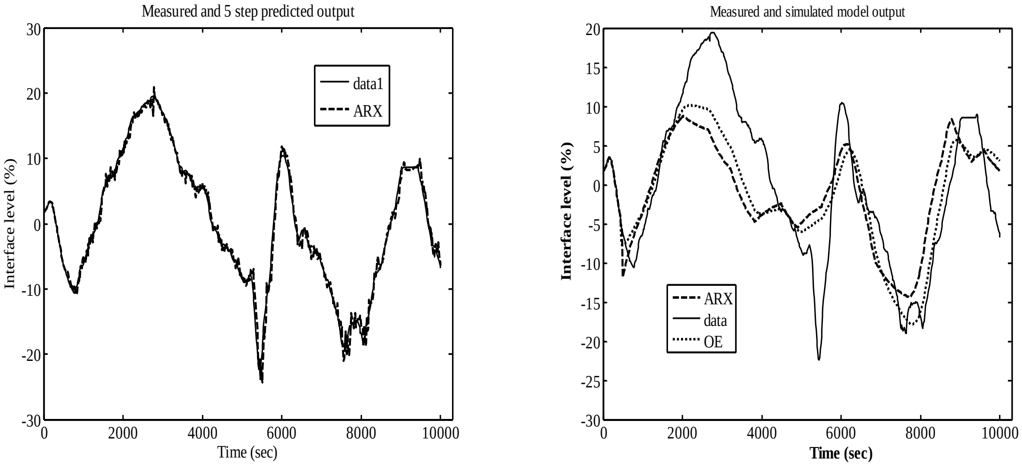


Figure 5.6: Interface level response to step change in pump speed at operating condition-2

The process has a large dead time and the dead time is modeled by careful examination of the step response plots. If the bypass line is open in the downstream side (operating condition-2), it is easy to pump the underflow and consequently the interface level responds faster to changes in the pump speed. The time delays in the process are of the order of 6-8 minutes at operating condition-1 and there is a delay of approximately 2 minutes at operating condition-2. The overall data set was partitioned into two parts : (a) an identification set: consisting of the first 70% data points and (b) the validation data set: consisting of 100% of the data. Model validation was carried out by not just looking at the model predictions but also by checking with the plant engineers if the model actually makes sense. For example, a model with a positive gain would clearly be an incorrect model, since interface level should go down if we increase pump speed. Also, can get rough estimates of the magnitude of the process gain by looking at the step response for different models.

At operating condition-1, for the model identification exercise, the data is down sampled to 15 second sample period. An autoregressive exogenous (ARX) model of second order and a first order continuous-time OE model using CONTSID are fitted to the data set and they provide good models for the control design at operating condition-1. Fig. 5.7 shows the model predictions for both the models.

The two models are:



(a) Five step ahead predictions

(b) Infinite step ahead predictions

Figure 5.7: Model predictions at operating condition-1

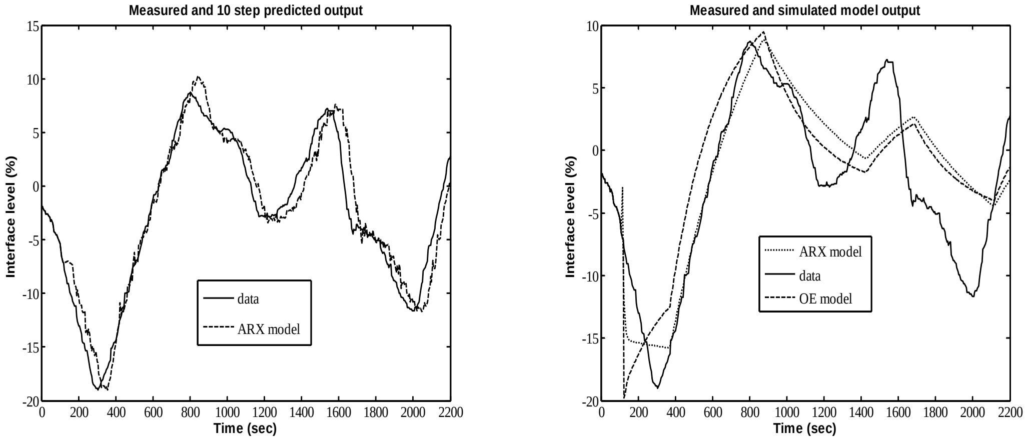
$$\text{ARX} \quad : \quad (1 - 1.85q^{-1} + 0.855q^{-2})\; y(t) = -0.00468\; q^{-32}\; u(t)\; + e(t) \tag{5.1}$$
$$\text{OE} \quad : \quad y(t) = \frac{-0.0024}{s + 0.0017} e^{-480s}\; u(t) + e(t) \tag{5.2}$$

For operating condition-2, a first order continuous-time OE model using CONTSID and second order ARX model were obtained. They both provide parsimonious models for control design purposes. For this condition, the data is down sampled to 5 seconds to build models. Fig. 5.8 shows the model predictions at operating condition-2. The two models are:

$$\text{ARX} \quad : \quad (1 - 1.53q^{-1} + 0.534q^{-2})\; y(t) = -0.00884\; q^{-24}\; u(t)\; + e(t) \tag{5.3}$$
$$\text{OE} \quad : \quad y(t) = \frac{-0.004263}{s + 0.003992} e^{-120s}\; u(t) + e(t) \tag{5.4}$$

As evident from Figures 5.7 and 5.8, the models are not perfect under the infinite-step ahead prediction. This is the acid-test for model validation. These models do perform much better at the 10-step ahead prediction clearly indicating that the process has many unmeasured disturbances. In summary, the models do capture the major transients and direction in the data, which is sufficient to provide fairly reliable model predictions and therefore these models can be used for the design of predictive control schemes with a prediction horizon of 10 to 20 sample periods.

There are a lot of disturbances which affect this process. In mineral processing, Hodouin et al. (2001) emphasizes that the optimization and control cannot



(a) Ten step ahead predictions

(b) Infinite step ahead predictions

Figure 5.8: Model predictions at operating condition-2

be performed without some information on the input disturbances (the material properties), the process states, and the final product quality. This is very also true in the oil sands industry. The operating efficiency is dependent on the quality of information used, which is used to build models and subsequently design model based control algorithms. Inside the SepCell, interaction effects of the mined feed, recycle flow, and other variables, with the interface level are needed, in order to study the effect of the changing disturbances that affects the SepCell. These are fast scanned measured disturbance variables unlike the feed quality which is obtained using lab analysis. For current process operations, there are frequent drop in the feed flow for LINE-2 compared to LINE-1 and this in turn also affects the interface level dynamics.

Fig. 5.9 shows the process structure for the MISO control design. We studied the effect of the recycle flow and the feed flow on the interface level as the potential measured disturbances and middlings flow as an additional manipulating variable; however, the middlings flow is be manipulated to a limited extent because of the process operational constraints. Fig. 5.10 presents the plots for a step test for these variables. During the tests for these variables, the pump speed was kept constant. First order plus dead time models are build for these variables and the models for recycle flow and feed flow were used as feedforward signals as will be described later.

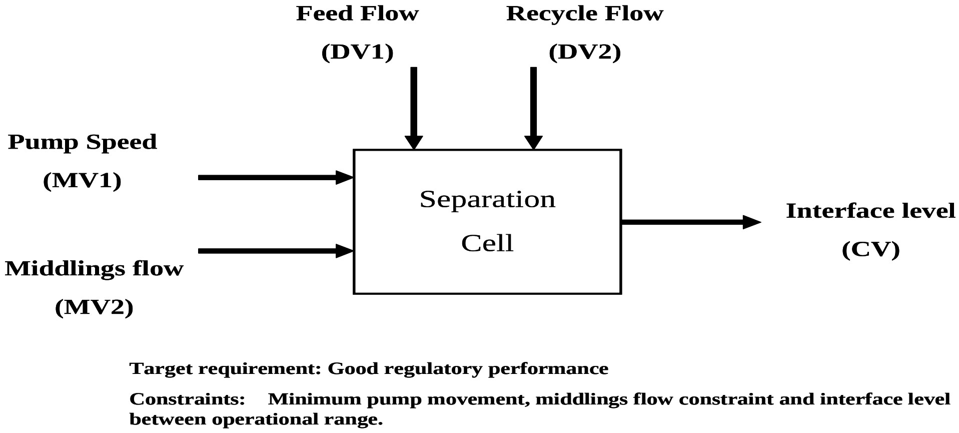
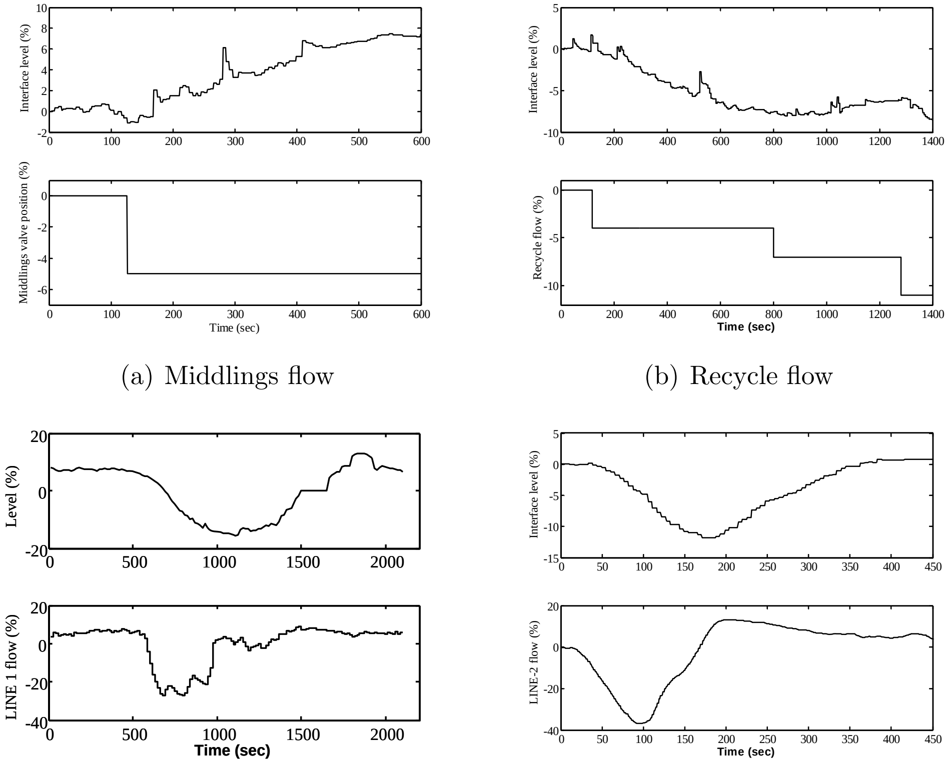

Feed Flow
(DV1)
Recycle Flow
(DV2)
Pump Speed
(MV1)
Middlings flow
(MV2)
Separation
Cell
Interface level
(CV)
Target requirement: Good regulatory performance
Constraints: Minimum pump movement, middlings flow constraint and interface level between operational range.

Figure 5.9: Process structure for MISO control design



(a) Middlings flow

(b) Recycle flow

(c) LINE-1 flow

(d) LINE-2 flow

Figure 5.10: Effect of measured disturbances and middlings flow on interface level

5.4 Model based predictive control

In order to obtain good overall control of separation cells, one needs good control of the local variables such as flow rates, feed density. Only after these are satisfactory, can one integrate them with the interface level control and the primary and secondary floatation cells to obtain specific grades for the outflows from the separation cell such as middlings and the underflow tailings. However, with current process operations, it is not easy to change the set points of a majority of the local controllers and also one should not upset the process much by 'experimenting' with so many factors at the same time. Thus, the objective here is to apply advanced control schemes to improve the regulatory interface level control compared to current operations, and move in a sequential manner to integrate the local controllers, for example, feed flows, feed characteristics and plant constraints to achieve performance close to the target requirements.

One can find examples of many important industrial control loops in process industries which exhibit long dead time characteristics and very large dominant time constants. The reason for very slow dynamic responses of these control loops is the presence of dead time. The controller design of such processes is a challenging problem. PID controllers are still widely used in the industry because they are easy to implement and they perform well for a wide class of processes. However, PID controllers are not very well suited for control of processes with large time delays since it can cause stability issues for the closed loop systems. The most popular and effective dead-time compensator in use today is the Smith predictor (Smith, 1957). The controller can be designed as if the process is delay free and the output response to a reference input can be adjusted as desired. Åström et al. (1994); Watanabe and Ito (1981) present some of the early works on how to tune controllers using Smith predictors for processes with time delays. For processes with time delays, the Smith predictor structure and the internal model control (IMC) structure are equivalent. These simple modifications in the control structure work very well for SISO processes; however, for multi-variable systems more advanced control schemes such as MPC, which can handle process constraints, are more practical and present an opportunity for long term operation without expert intervention. In the overall operation of the process, MPC presents significant advantages over other multi-variable controllers which primarily include handling interactions and constraints. The real economic benefits accrue when the dynamic control allows the set points to be moved closer to the constraints without violating them (Prett and Gillett, 1980); and this is what we expect to achieve in the SepCell by using MPC. The final objective for this case study is to implement MIMO MPC and this is the first step in achieving this objective and therefore the use of MPC controller.

5.4.1 IMC control design

Internal model control (IMC) was developed by Rivera et al. (1986). The distinguishing characteristic of the IMC structure is the incorporation of the process model which works in parallel with the actual process, and leads to analytical expressions for the controller setting. The IMC method is based on the simplified feedback structure as shown in Fig. 5.11. A process model $\hat{G}(s)$ (of order n) appears in the control structure for the process $G(s)$ and if $G(s)$=$\hat{G}(s)$, that is, perfect modeling, and there are no disturbances present, the system behaves like an open loop system. The controller $C(s)$ is designed as the inverse of the process model $\hat{G}(s)$. This means perfect set point tracking and asymptotic rejection of disturbances can be achieved if there is no plant-model mismatch. However, in practice a filter is introduced in order to make the IMC controller physically realizable. Thus, the first step in the IMC controller design is to factor the process model as

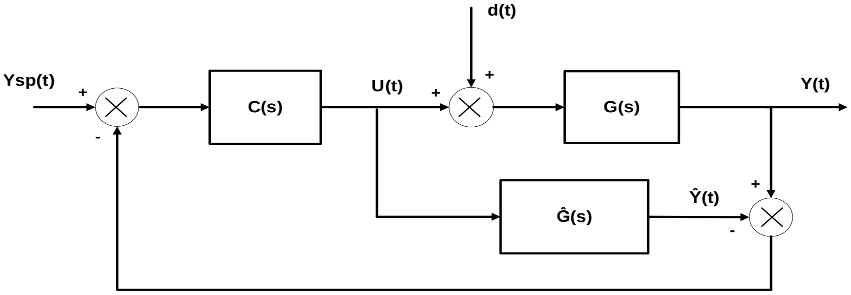


Figure 5.11: Internal model control control strategy

$$\hat{G}(s) = \hat{G}_{+}(s)\hat{G}_{-}(s) \tag{5.5}$$

where $\hat{G_{+}}$ contains all the time delays and right-half plane zeros. Then the IMC controller is defined as

$$C(s) = \hat{G}_{-}^{-1}(s)F(s) \tag{5.6}$$

where $F(s)$ is a low pass filter and the simplest design is given as Rivera et al. (1986)

$$F(s) = \frac{1}{(\tau s + 1)^{n}} \tag{5.7}$$

For processes with time delays, the Smith predictor structure and the internal model control structure are equivalent and by suitably selecting the tuning parameter τ, we can design the closed loop performance of the system.

5.4.2 Model predictive controller design

A class of control algorithms that use dynamic process models to incorporate process knowledge into the controller structure are known as predictive control algorithms. Model predictive controllers are good for highly complex and interconnected dynamic systems. In contrast to a standard PID controller, MPC attempts to minimize the sum of squares of future control errors over a user specified time horizon with some regard for the control energy. It calculates an optimal control action by taking into account process dynamic behavior and operational constraints (Qin and Badgwell, 2003; Rawlings, 2000).

Model predictive algorithms utilize the moving horizon approach, computing a set of control moves over a horizon. The initial control moves are implemented, while the remaining calculated moves are discarded. The process is then repeated at subsequent sampling times. Implementations of MPC in industry commonly use linear process models. MPC techniques have been implemented in various forms, some of which have found their way to the market place. Examples of such implementation algorithms are available in the review by Qin and Badgwell (2003).

The objective in MPC consists of minimizing the following dynamic objective function:

$$J(\mathbf{u}^{CH}) = \sum_{j=1}^{PH} \left\{ \left\| \mathbf{e}^{y}_{k+j} \right\|^2_{\mathbf{Q}} + \|\mathbf{s}_j\|^2_{\mathbf{T}} \right\} + \sum_{j=0}^{CH-1} \left\{ \left\| \mathbf{e}^{u}_{k+j} \right\|^2_{\mathbf{R}} + \|\mathbf{\Delta u}_{k+j}\|^2_{\mathbf{S}} \right\} \tag{5.8}$$

subject to model constraint (model could be linear):

$$\mathbf{x}_{k+j} = \mathbf{f}(\mathbf{x}_{k+j-1}, \mathbf{u}_{k+j-1}) \ \ \forall\ j = 1, 2, ...PH. \tag{5.9}$$

$$\mathbf{y}_{k+j} = \mathbf{g}(\mathbf{x}_{k+j}, \mathbf{u}_{k+j}) \ \ \forall\ j = 1, 2, ...PH. \tag{5.10}$$

and subject to inequality constraints:

$$\begin{aligned} \mathbf{y}_L - \mathbf{s}_j &\leq \mathbf{y}_{k+j} \leq \mathbf{y}_U + \mathbf{s}_j \qquad \forall\ j = 1, 2, ...PH \\ \mathbf{s}_j &\geq 0 \quad \forall\ j = 1, 2, ...PH \\ \mathbf{u}_L &\leq \mathbf{u}_{k+j} \leq \mathbf{u}_U \qquad \forall\ j = 0, 1, ...CH - 1 \\ \Delta\mathbf{u}_L &\leq \Delta\mathbf{u}_{k+j} \leq \Delta\mathbf{u}_U \qquad \forall\ j = 0, 1, ...CH - 1 \end{aligned} \tag{5.11}$$

Thus, the objective function penalizes the deviation from the desired output trajectory $\mathbf{y}^r_{k+j}$, defined as $\mathbf{e}^y_{k+j} = \mathbf{y}_{k+j}$ - $\mathbf{y}^r_{k+j}$, over the prediction horizon, PH. Output constraint violations are penalized by minimizing the size of the output constraint slack variables $\mathbf{s}_j$. Also, for the manipulating inputs, future input deviations from the desired steady-state input $\mathbf{u}_s$ are controlled using input penalties defined as $\mathbf{e}^u_{k+j} = \mathbf{u}_{k+j}$ - $\mathbf{u}_s$, over a control horizon of length CH. Rapid input changes are penalized with a separate term involving the moves $\Delta\mathbf{u}_{k+j}$. The relative importance of each terms in the objective function is defined by the weighting matrices $\mathbf{Q}$, $\mathbf{T}$, $\mathbf{R}$ and $\mathbf{S}$. The MPC dynamic optimization is formulated as a quadratic programming (QP) problem and the solution to the quadratic objective function (5.8) is a set of CH input moves

$$\mathbf{u}^{CH} = (\mathbf{u}_k, \mathbf{u}_{k-1}, ..., \mathbf{u}_{k+CH-1}) \tag{5.12}$$

Only the first input $\mathbf{u}_k$ is implemented on the process and the calculation is repeated at every execution interval. Also, note that the MPC is designed to account for any plant-model mismatch by introducing the integral action in the controller which is achieved by using the difference between the measured and predicted controlled variable from the model.

5.5 Control strategy applied to industrial setup

Before implementing the control scheme (both IMC and MPC) on the real process, a number of off-line simulation studies were undertaken to identify suitable parameters for the controller and achieve the required closed loop performance and provide some robustness to the controller. Finally, the controllers were tested and implemented at the Suncor's Extraction unit. On-line implementation of the controllers was performed using OPC (Object Linking and Embedding for Process Control) objects as the communication channel between the commercial software Controller Performance Optimizer (CPO) and the DCS (distributed control system). Thus, the calculations were done in CPO and the results of the calculations are sent back to the DCS through an OPC client object that connects to an OPC server. The results presented here are actual field test results to show the benefits of the advanced process control application to the extraction unit. The performance of the controllers (regulatory performance) is assessed by comparing the standard deviation ($\bar{s}$) of the process variables around the mean value; the interface level, the pump speed and the underflow. Here $\bar{s}$ is defined as

$$\bar{s} = std(PV) \tag{5.13}$$

5.5.1 Internal model control

Fig. 5.12 shows the closed loop performance for a period of 7.5 hrs where the IMC (blue line) was tested, along side conventional closed loop performance of the same process (PID Control) a few days earlier as well as after the IMC test (black dotted line). The results are for testing at operating condition-1. The plot shows the interface level along with the pump speed (manipulated variable) and the underflow (flowrate through underflow line). The data as shown in Fig. 5.12 is not contiguous. As is the case in industry, one has to make do with whatever opportunity is available to test the proposed control schemes, therefore, we have just compared the closed loop performance of the IMC controller with the current operation which uses PID including some manual control as evident by constant pump speeds in the right portion of fig. 5.12. IMC is configuration for this case by keeping closed loop time constant (λ) for the process same as open loop time constant.

As can be seen, the IMC controller delivers satisfactory performance and keeps the process variable near the set point while keeping the pump movement to the minimum compared to the results before and after the testing period. Validation and comparison of controllers in an industrial setting are difficult as the variabilities of the influences (disturbances) from outside are significant and differ during different times. Fig. 5.13 shows the histograms for the interface level and the underflow. It is clearly evident that using IMC, we get much tighter regulatory control compared to the current operations using PID control. The time delay compensation of the IMC controller helps to reduce the variance of the process variables. Table 5.1 shows the comparison of all the key performance indices (KPI) with and without the IMC control.

Table 5.1: Comparison of regulatory performance ($\overline{s}$) with and without IMC at operating condition-1

KPI (unit)	During IMC testing	Current operations (PID) - After IMC testing
Interface level (%)	3.23	8.89
Pump speed (%)	3.55	13.68
Underflow (%)	3.70	11.86

Therefore, there is a significant improvement in the regulatory closed loop response with an IMC and this provides encouraging results to further test and improve process operations. As can be seen, the advanced process control benefits

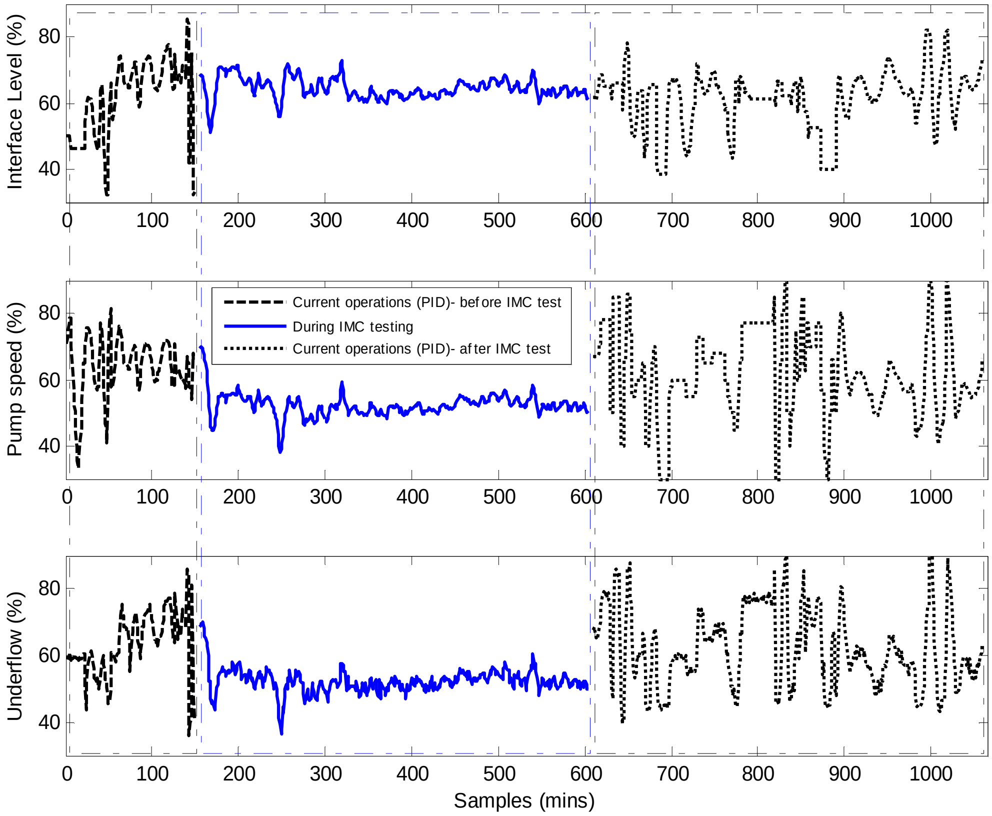


Figure 5.12: Closed loop performance of IMC at operating condition-1

are visible and it is important to get a feel for the potential benefit a control system like the one presented here may have in comparison to current operation. However, to handle process constraints and feedforward disturbance along with further moving the set point closer to the limits, we tested industrial an MPC, the results for the same are presented next.

5.5.2 Model predictive control

Using MPC, we added process constraints for both the interface level and the pump speed. We used the estimated models as obtained at two operating conditions to design MPC with and without feedforward control.

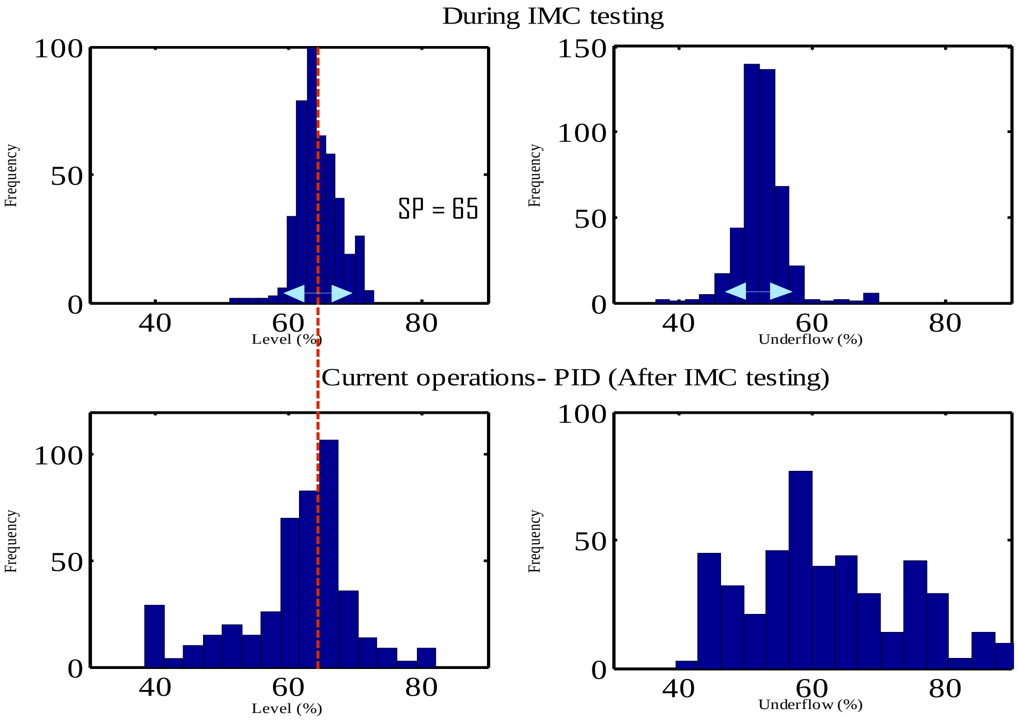


Figure 5.13: Histograms of process variables for IMC at operating condition-1

MPC at operating condition-1

Fig. 5.14 shows the closed loop performance for roughly 15 hours of testing data (blue line). The plot shows the interface level along with the pump speed (manipulated variable) and the underflow, before (black dashed line) as well as after (black dotted line) the MPC testing. The data as shown in Fig. 5.14 is not continuous as configuration steps were required before and after the switch from current operations to 'MPC on/off' was done. The execution time for the MPC was chosen to be 10 sec. For MPC, the prediction horizon was set to be 150 and the control horizon as 35. The weighting matrices **Q**, **R**, **S** and **T** were set to the following:

$$\mathbf{Q} = 1.75, \mathbf{T} = 2.0, \mathbf{R} = 0.0, \mathbf{S} = 2.0.$$

These settings were chosen such that we achieve the required closed loop performance from the controller. The desired output trajectory is built using a time constant of 600 sec. The constraints on MV and CV are shown in the Fig. 5.14. As can be seen, the benefits of the MPC are clearly visible, the control effort (pump movement) is small with the interface level within the constraints for 95% of the time. The interface level violates the constraints when there is a significant drop in the feed flow rate as will be explained later. A fair and perfect comparison is difficult as the disturbances might be different during the three periods when these tests were carried out. Fig. 5.15 shows the histograms for the interface level and underflow in the order of testing. It is clearly evident that the MPC controller gives much tighter regulatory control compared to the current operations using PID control. Table 5.2 shows the comparison of all the key performance indices (KPI) with and without the SISO MPC control.

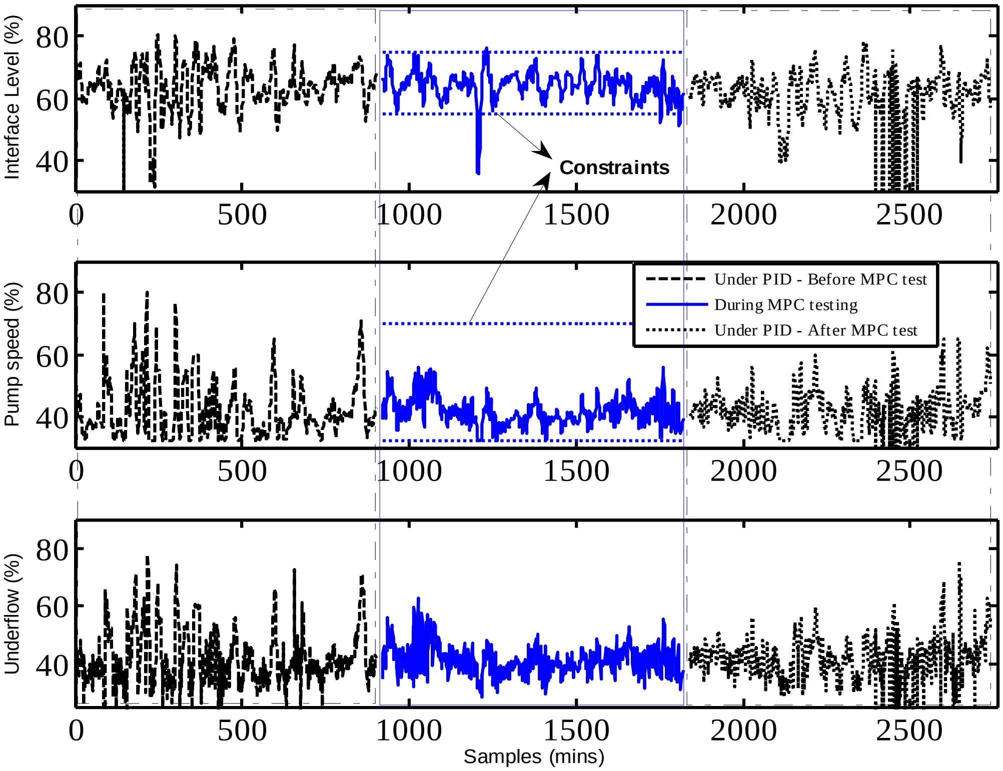


Figure 5.14: Closed loop performance of MPC at operating condition-1

As can be seen, there is significant reduction in the standard deviations of all three key performance indices using MPC. The pump speed is also relatively constant using MPC, resulting in a smoother operation of the controlled system downstream. Fig. 5.16 shows the SISO MPC closed loop response along with the feed flow conditions. As can be seen from Fig. 5.16, whenever there is significant change in the feed flow, it affects the interface level. The time when the interface level violates the constraints is the time when there is a loss of feed or significant

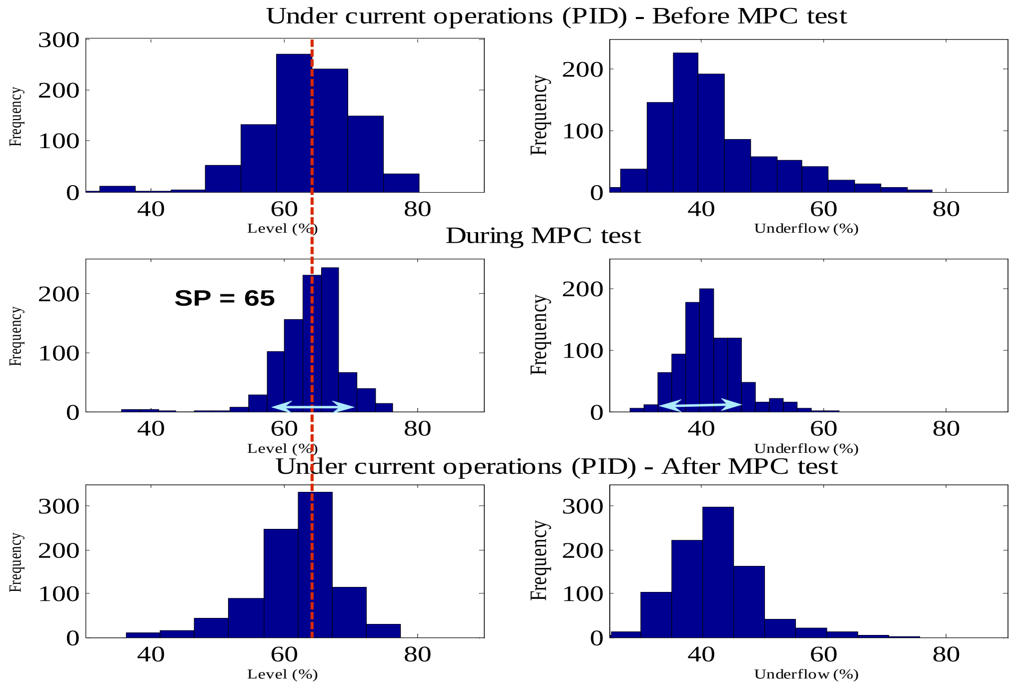


Figure 5.15: Histograms of process variables at operating condition-1

Table 5.2: Comparison of regulatory performance ($\overline{s}$) with and without SISO MPC at operating condition-1

KPI (unit)	Current operations (PID)- Before MPC test	During MPC test	Current operations (PID)- After MPC test
Interface level (%)	7.71	4.80	10.95
Pump speed (%)	9.14	4.53	8.78
Underflow (%)	9.48	4.99	9.12

drop in feed flow. Even though the pump speed reaches the minimum, it cannot prevent the upset in the interface level when there is this extent of feed loss. Therefore, we need to add some of the measured disturbance variables as feedforward signals and add additional manipulating variables such as middlings flow to further make the closed loop control proactive with respect to the disturbances. This is the main reason for designing the MISO MPC.

MPC at operating condition-2

The execution time for the MPC at this operating condition was also set to 10sec. At this operating condition, the prediction horizon was set to be 100 and the

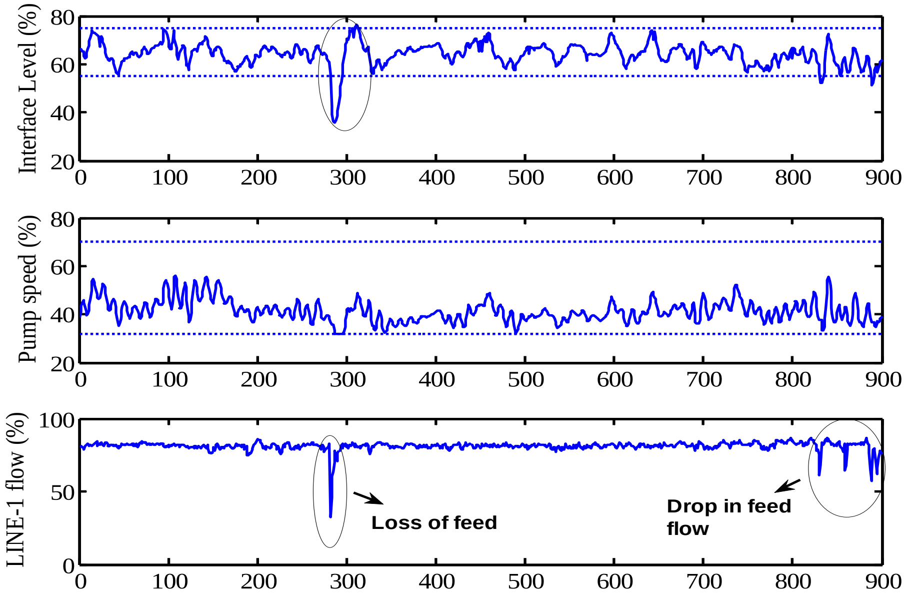


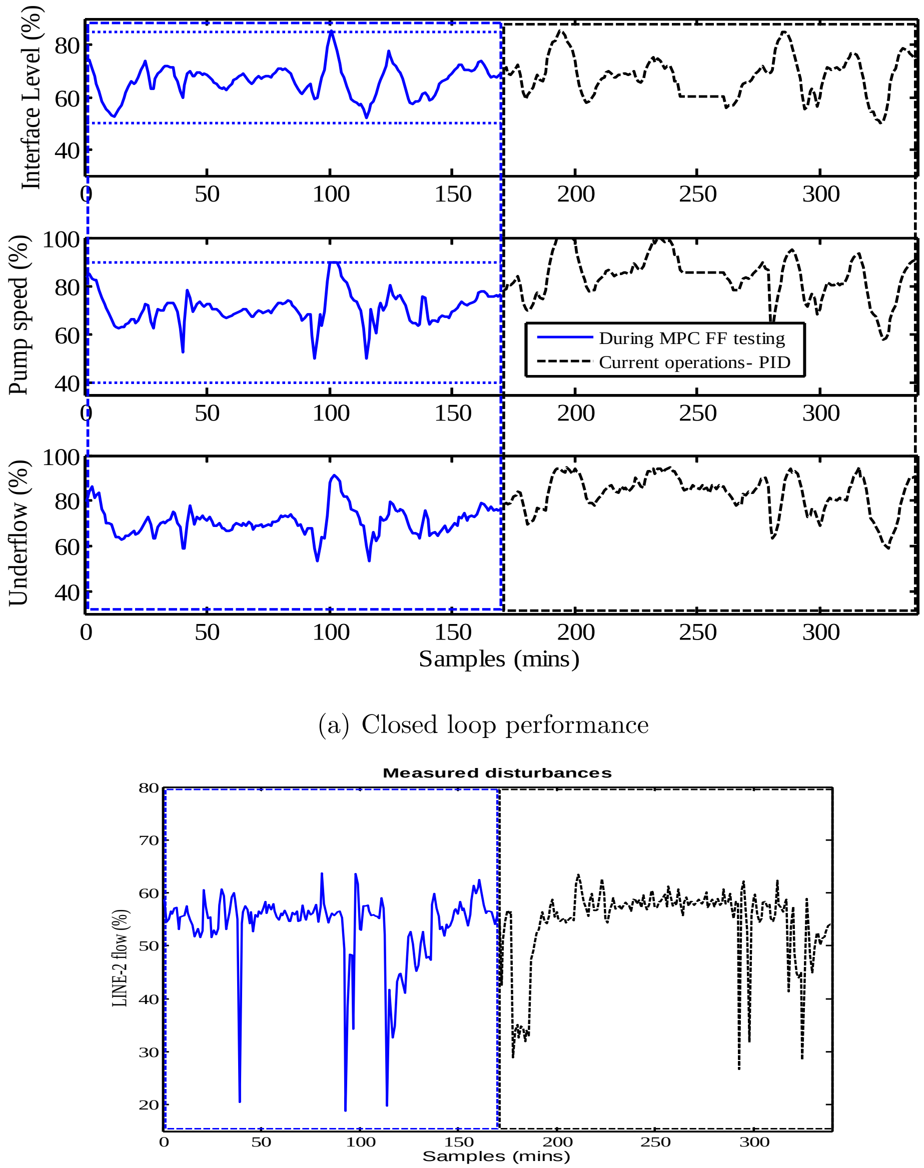
Figure 5.16: Closed loop response along with feed flow condition for the SISO MPC test at operating condition-1

control horizon as 25. The weighting matrices, **Q**, **R**, **S** and **T** were set to the following:

$$\mathbf{Q} = 1.5, \mathbf{R} = 0.0, \mathbf{T} = 2.0, \mathbf{S} = 2.0.$$

Again, these settings were chosen such that we achieve good performance from the controller. The desired output trajectory is built using a time constant of 300 sec. The constraints on MV and CV are shown in the Fig. 5.17. At this operating condition we tested the MPC with feedforward (MPC FF) for LINE-2 and compared the performance with the current operations. The switch from MPC FF to PID is instantaneous in Fig. 5.17. Fig. 5.17 shows the closed loop performance comparison of MPC FF (blue line) with PID control (black dashed line) for roughly 3 hours of process operation. A fair and perfect comparison is difficult as the disturbances might be different for the two cases; however we have presented the changes in feed flow (major disturbance) in Fig. 5.17 along with the closed loop performance. The numbers of feed drops for the two cases are almost same.

Table 5.3 shows the comparison of all the key performance indices (KPI) and as can be seen the MPC with feedforward control performs better than current operations which uses PID control. Fig. 5.18 shows the histogram plots for the



(a) Closed loop performance

(b) Feedforward disturbance, LINE-2 flow

Figure 5.17: Closed loop performance of MPC FF at operating condition-2

interface level and underflow for these two controllers.

Fig. 5.19 shows closed loop performance comparison for current operation (PID), MPC without feedforward (MPC), and MPC with feedforward (MPC FF). Even though there are more feed flow drops for the MPC FF case, the MPC with feed forward helps keep the interface level between the constraints. Again as can be seen from Table 5.4, the MPC with feedforward performs better in providing better interface regulation and is able to reduce the variability of the process variables.

Table 5.3: Comparison of regulatory performance ($\overline{s}$) for MPC FF at operating condition-2

KPI (unit)	During MPC FF testing	Current operations - PID
Interface level (%)	5.57	7.86
Pump speed (%)	6.57	9.23
Underflow (%)	6.40	8.23

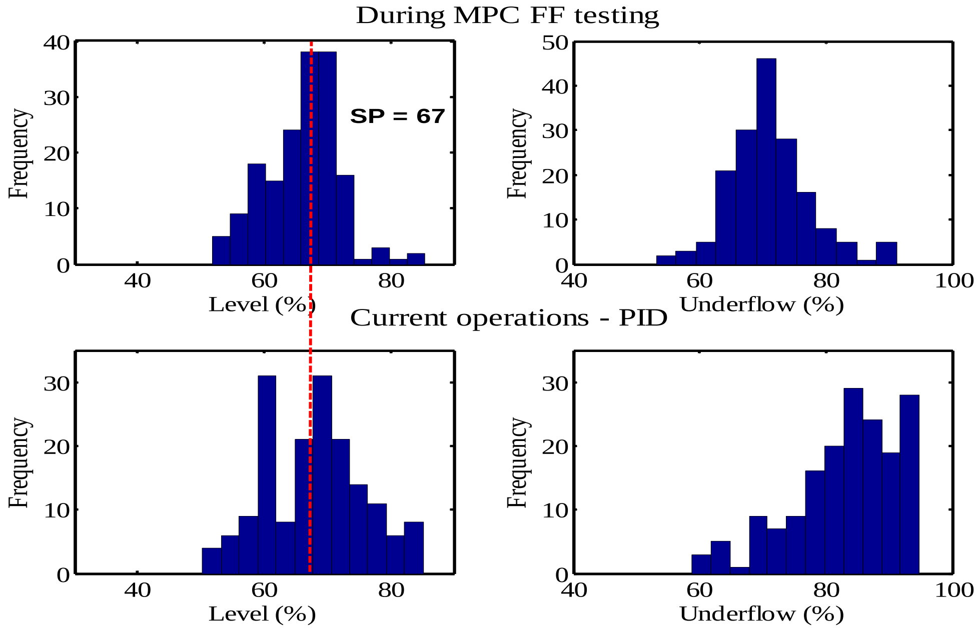


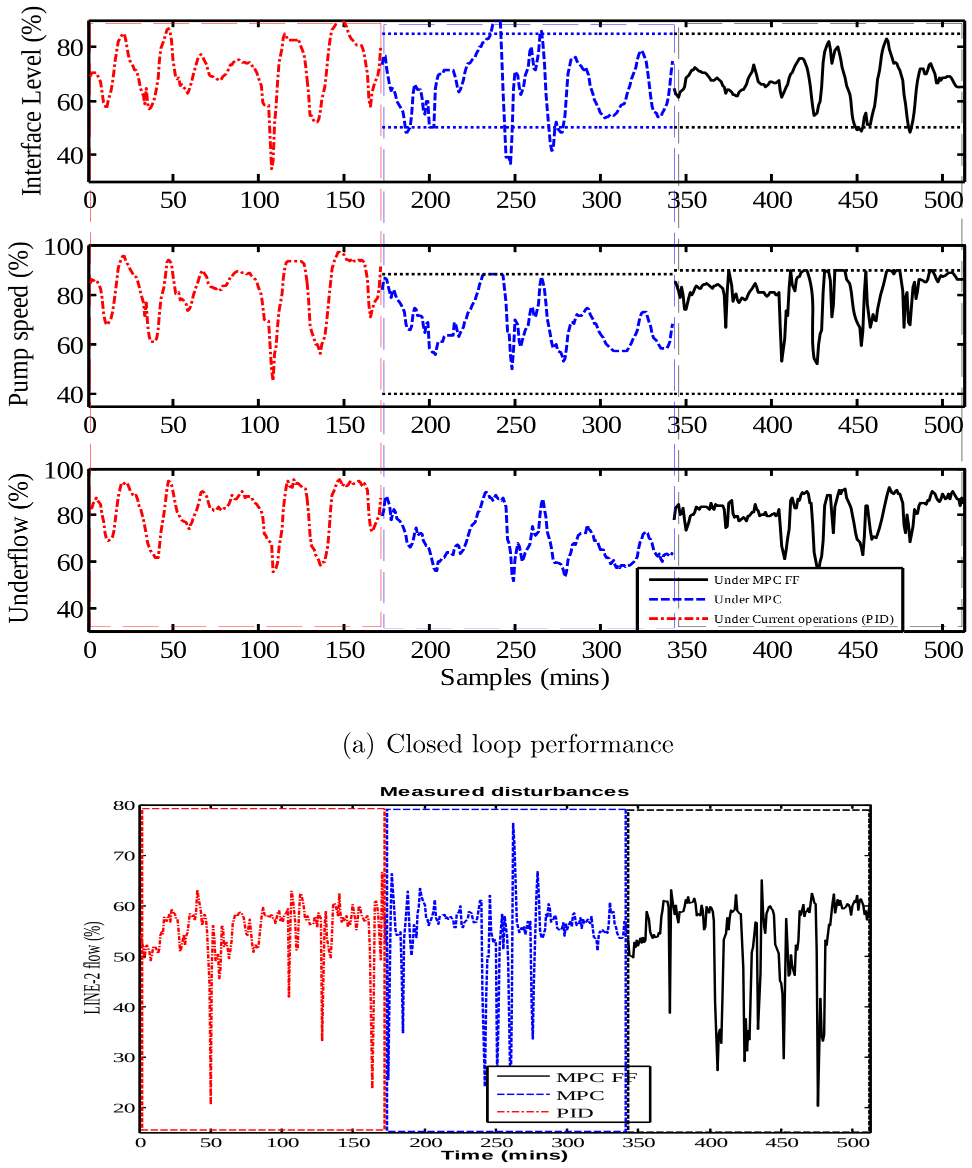
Figure 5.18: Histograms of process variables at operating condition-2

Fig. 5.20 shows the histograms for the interface level and underflow rates for the three controllers.

Table 5.4: Comparison of regulatory performance ($\overline{s}$) for MPC with and without feed forward at operating condition-2

KPI (unit)	Under current operations (PID)	Under MPC	Under MPC FF
Interface level (%)	10.24	11.43	7.37
Pump speed (%)	11.02	9.64	7.94
Underflow (%)	10.49	9.25	7.33

Therefore, we see that feed forward control really helps in regulating in the



(a) Closed loop performance

(b) Feedforward disturbance, LINE-2 flow

Figure 5.19: Closed loop performance comparison for MPC with and without feed forward at operating condition-2

presence of one of the major disturbance to the process and helps to improve the closed loop performance of the MPC. The reduced variance of these process variables results in smoother operations downstream and therefore, the result suggests that there are financial benefits in using the MPC strategy.

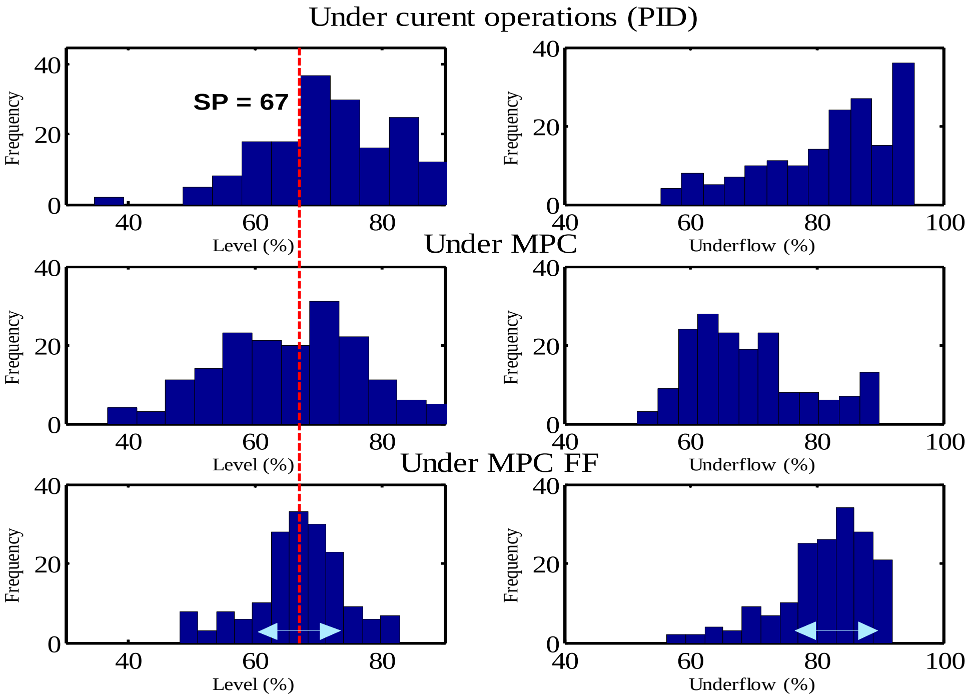


Figure 5.20: Histograms of process variables for MPC with and without feed forward at operating condition-2

MISO model predictive control

The testing using MISO MPC is ongoing research for this process and more detailed testing at the two operating conditions will be done to have a multivariate MPC application in the extraction unit. The results presented in this section are some of the initial testing results. We do not have closed loop response data before and after MISO MPC testing and the results are presented to show our approach for more testing.

Fig. 5.21 shows the closed loop performance for roughly 3.3 hours of testing data when we used the middlings as a second manipulating variable and the recycle flow and feed flow as feed-forward disturbances. We expect to achieve much better closed loop performance by adding an additional manipulating variable, the middlings flow. However, due to process restrictions, the middlings flow is only changed by ± 5% and we cannot fully exploit the middlings flow. Therefore, the closed-loop response hits the constraints more often. MPC controllers can provide much better and safer responses to disturbances and we expect to see that in further testing on the process.

5.6 Economic benefits

In the introduction we discussed that a small improvement in the process efficiencies and operation can greatly influence bitumen recovery, reduce environmental impact and influence process economics. The test results using IMC and MPC have been very positive and show significant reduction in the variance of the interface level compared to current operations. As a result of this the overall variability of the process and manipulated variables were shown to be considerably reduced when using IMC and MPC controllers. Since there is also significant reduction in the pump movement, this translates into pump energy savings and hence significant cost reduction. Since the extraction process is a continuous process, a smoother operation of the separation cell ensures smoother operation downstream, and all these benefits are directly related to increased bitumen recovery and throughput.

5.7 Conclusions

In this chapter, a case study application of identification and model based predictive control design for the interface level control in an industrial separation cell process in the oil sands industry has been discussed. IMC and MPC were designed, implemented and tested in real time on the industrial separation cell to observe the benefits of using advanced process control schemes in the extraction unit. The test results show that both IMC and MPC schemes provide significant benefits over the the current operations which uses PID (plus manual) control. The benefits include significant reduction in the variance of the interface level and underflow pump movement, leading to higher bitumen recovery and reduced losses at the tailings and also pump energy savings and smoother operation downstream.

In future studies, a multi-input single output (MIMO) control formulation, incorporating additional CVs and MVs would be tried and tested. This is expected to make the control problem more challenging.

Bibliography

Alberta Energy: Oil Sands <http://www.energy.gov.ab.ca/OurBusiness/oilsands.asp>.

J. Masliyah, Z. Zhou, Zhenghe Xu, J. Czarnechki and Hassan Hamza. Understanding water-based bitumen extraction from athabasca oil sands. *The Canadian Journal of Chemical Engineering*, vol. 82, pp. 628-654, 2004.

K. A. Clark. Hot-Water separation of Alberta bituminous sand. *Transactions of Canadian Institute of Mining and Metallurgy*, vol. 47, pp. 257-274, 1944.

P. Dougan and K. McDowell. Sensor development in oil sand processing. *Proceedings of the Dynamic Modeling Control Applications for Industry Workshop*, Vancouver, Canada, 1997.

P. Jampana, S.L. Shah and R. Kadali. Computer vision based interface level control in separation cells. *Control Engineering Practice*, vol. 18, pp. 349-357, 2010.

P. Jampana and S.L. Shah. An image differencing method for interface level detection in separation cells. *Machine Vision and Applications*, July issue, 2011.

L. Fortuna, S. Graziani, A. Rizzo and M.G. Xibilia. Soft sensors for monitoring and control of industrial processes. *In Advances in Industrial Control*, Secaucus, NJ, USA, 2006.

E. Domlan, B. Huang, F. Xu, and A. Espejo. A decoupled multiple model approach for soft sensors design. *Control Engineering Practice*, vol. 19, pp. 126-134, 2011.

L.G. Bergh and J.B. Yianatos. The long way toward multivariate predictive control of flotation processes. *Journal of Process Control*, vol. 21, pp. 226-234, 2011.

S.J. Qin and T.A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, vol. 11, pp. 733-764, 2003.

J. Rawlings. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, vol. 20, pp. 38-52, 2000.

K. S. Stadler, J. Poland and E. Gallestey. Model predictive control of a rotary cement kiln. *Control Engineering Practice*, vol. 19, pp. 1-9, 2011.

M. O'Brien, J. Mack , B. Lennox, D. Lovett and A. Wall. Model predictive control of an activated sludge process : A case study. *Control Engineering Practice*, vol. 19, pp. 54-61, 2011.

L. Ljung. System Identification: Theory for the Users. *Prentice-Hall*, 2nd ed., New Jersey, 1999.

H. Garnier and L. Wang. Identification of Continuous-time Models from Sampled Data. *Advances in Industrial Control*, Springer, London, 2008.

H. Garnier, M. Gilson, and O. Cervellin. Latest developments for the matlab contsid toolbox. *IFAC symposium on system identification, SYSID 2006*, Newcastle, Australia, 2006.

O. J. M. Smith. Closer control of loops with dead time. *Chem. Eng. Prog.*, vol. 53(5), pp. 217-219, 1957.

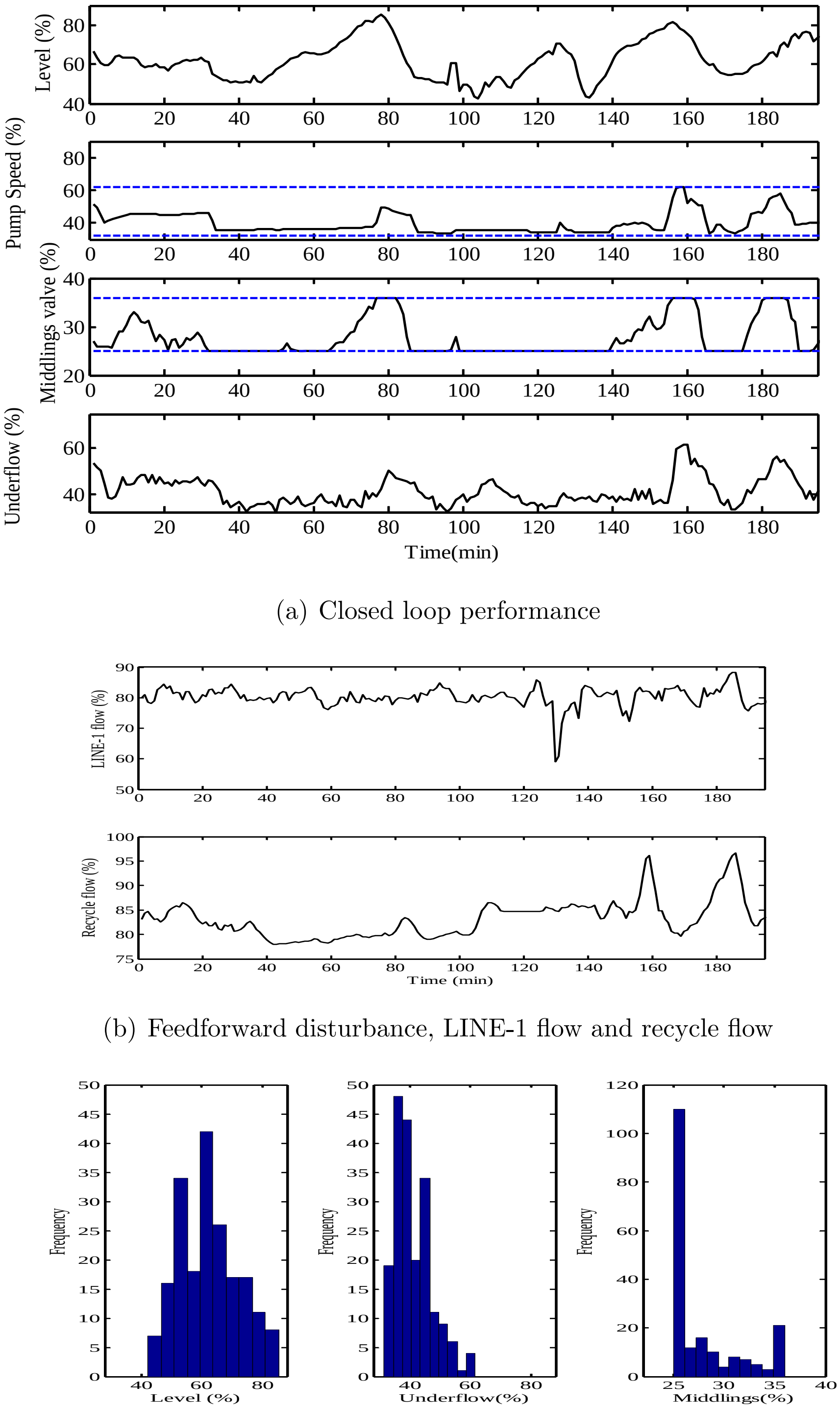
K. J. Åström, C. C. Hang and B. C. Lim. A new Smith predictor for controlling a process with an integrator and long dead-time. *IEEE Transaction on Automatic Control*, vol. 39(2), pp. 343-345, 1994.

K. Watanabe and M. Ito. A process-model control for linear systems with delay. *IEEE Transaction on Automatic Control*, vol. 26(6), pp. 1261-1266, 1981.

D. Hodouin, S.-L. Jamsa-Jounela, M.T. Carvalho and L.G. Bergh. State of the art and challenges in mineral processing control. *Control Engineering Practice*, vol. 9, pp. 995-1005, 2001.

D. E. Rivera, M. Morari and S. Sigurd. Internal model control- PID controller design. *Industrial Chemistry in Process Design and Development*, vol. 25(1), pp. 252-265, 1986.

D. M. Prett and R. D. Gillette. Optimization and constrained multivariable control of a catalytic cracking unit. *In Proceedings of the Joint Automatic Control Conference*, 1980.



(a) Closed loop performance

(b) Feedforward disturbance, LINE-1 flow and recycle flow

(c) Histogram plots of process variables for MISO MPC

Figure 5.21: Closed loop performance comparison at operating condition-2 for MISO MPC

Chapter 6

Model based predictive control of bituminous froth heaters in oil sands extraction

In a majority of oil sands operations, the hot-water floatation based extraction process is used for extracting bitumen hydrocarbon from the ore. In order to de-aerate the bitumen froth obtained from the hot water based flotation method and make it less viscous and easy to handle, it needs to be heated before being passed on to the next stage. This chapter details a case study application of identification and model based predictive control design for the bitumen froth steam heater (FHC) process. The process represents a non-square multi-variable problem with more controlled variables than manipulating variables; and a gain scheduling model predictive controller (MPC) using linear models obtained from industrial data is designed, and tested in simulations. The industrial testing is in progress. Simulation results shows that the multi variable MPC scheme can provide significant benefits over the current operation which uses a number of local PID controllers. These benefits include better handling of process interactions and better regulatory performance resulting in the reduction of the variance of exit froth temperature and steam usage (steam energy savings).

6.1 Introduction

In the oil sands industry, the process of bitumen production from oil sands involves various inter-related unit operations such as mining, utilities, extraction, froth treatment, water management and upgrading (Masliyah et al., 2004). The mining operation directly affects extraction, which in turn affects the froth treatment and the upgrading process. Only through proper integration of all these units, can the bitumen be efficiently and economically recovered. There is tremendous scope for improvement in the way advanced process control is used in the oil sands operation and there are a number of potential candidates for application of dynamic model based predictive control schemes.

The Clark hot water floatation method is one of the methods to extract Bitumen from oil sands ore. One of the earliest papers in this regard can be found in Clark (1944). The aerated froth produced from the large floatation cells is difficult to work with and has properties similar to roofing tar. It is highly viscous, does not readily accepts heat and needs to be de-aerated before it can move to the next stage of the process. This is done by passing the froth through a series of froth steam heaters where superheated steam is injected and mixed with the bitumen froth (direct contact) resulting in solution at a homogenous temperature. The development of first principle models is usually time-consuming and needs a lot of effort, especially for a complex process (Henson, 1998). The froth heater operation is highly nonlinear and dynamic in nature. The aerated froth coming from the primary extraction does not readily accepts heat; the temperature as well as flow of the incoming bitumen froth to the heaters affects the dynamics of the two heaters and thus the process behaves differently when operated at different operating conditions. In addition, the heaters work in series, and hence the presence of interactions between the temperature loops for the two heaters. MPC based on data-based models seems to be an appropriate control methodology for this complex, interconnected dynamic system.

MPC computes optimal control moves by solving an optimization problem over a finite horizon, taking into account dynamic behavior of the process and the operational constraints (Qin and Badgwell, 2003; Rawlings, 2000). MPC techniques have been explored and implemented in various forms, some of which have found their way to the market place as discussed in Chapter 5 (introduction). Industrial applications of MPC rely mainly on linear empirical models obtained by employing time series analysis (Qin and Badgwell, 2003) which however can quickly become obsolete. Both the development and the maintenance of the dynamic model is of paramount importance for achieving good closed loop performance. Classical control design based on linearized dynamics may not exhibit good or even stabilizing performance when operating away from the equilibrium. In the case of changing operating conditions for a process, the problem of maintaining the stability and/or performance of a linear MPC scheme is handled in the following ways

(Morari and Lee, 1999): (a) incorporating robustness at controller design, (b) employing multiple models based controller design, and (c) updating the parameters of the linear prediction model either intermittently or on-line.

Incorporating robustness in the controller design to uncertainties in the process is an attractive option. However, for a chemical process, suitable modeling of the uncertainties to the process and then designing a controller that can deal with large variations in the operating conditions is a non-trivial task. Let us assume that if such a robust controller exists and can be designed, it may turn out to be quite conservative and may not provide uniformly satisfactory performance over the entire operating range. The second approach to handle changes in the operating conditions is to develop multiple linear models. For this, we need to perturb the plant at different desired operating conditions. The last option is updating linear model parameters intermittently, which is another attractive strategy. In this work, we use the second approach.

Leith and Leithead (2000) describe the important theoretical results and design procedures for gain-scheduling (GS) control by decomposing nonlinear design problems into linear sub-problems. Conventional GS methods employ divide and conquer approach where the parameter space is divided into small operating regions, where the process is regarded as a linear time invariant (LTI) system and LTI controllers (fairly robust) are designed for each of these operating conditions. The operating point is parameterized by a scheduling variable, and scheduling of controller gains is established based on this scheduling variable. At each interval of time, one of the linear controllers is activated based on the value that the scheduling variable achieve during closed-loop operation. This is a good compromise between performance and robustness and the stability question in the switching zone has been solved in Shamma and Xiong (1999). In general, the scheduling variable is time-varying and may either be an internal plant variable, or a function of internal plant variables, called endogenous variables, or an externally prescribed exogenous variable. It is assumed that the model based on first principles is not available and therefore empirical models are used, which are directly identified from the experimental data. In this work, the focus is on using the linear models, as linear system and control theory is well developed.

The main objective of the work in this chapter is to assess the benefits of using advanced process control algorithms to regulate the exit temperature of the bitumen froth from froth heaters. MPC is designed for this multi-input multi-output (MIMO) process. The main focus is on the application; therefore, the stability or robustness of gain scheduling linear controllers are not discussed in this chapter. The process represents a non-square multi-variable problem with more controlled variables, than manipulating variables and a gain scheduling MPC using linear models is designed, and tested in simulations. The industrial testing is in progress. The objective of the control is to regulate the exit froth temperature during process feed changes, and provide reduction in the variance of the process variables that result in steam energy savings and smoother operations downstream.

The organization of this chapter is as follows: Section 6.2 provides a brief description of the investigated process. The model identification for the control design is described in Section 6.3. Section 6.4 describes briefly the formulation of the control algorithms used for testing on the process. Section 6.5 presents the closed loop results in simulations for the gain scheduling MPC. Section 6.6 presents the economic benefits and the conclusions.

6.2 Process description

The Froth heater process represents an interstage between primary and secondary extraction units in the oil sands industry. The desired objective of these heaters is to increase the temperature of the incoming bitumen froth before it is sent to the next stage of the process. Traditionally, processing of aerated bitumen froth requires the froth to be heated to 190^0F to 200^0F (Gaston et al., 2011). This is done in order to make the froth less viscous and easy to handle in the later stages.

The froth steam heater process flowsheet is shown in Fig. 6.1. The aerated froth produced from the large floatation cells in the primary extraction is difficult to work with; it is highly viscous, does not readily accepts heat, and needs to be de-aerated before it can move to the next stage of the process. This bituminous froth is pumped and passed through a series of inline froth steam heaters to increase the temperature of the froth. The froth heater is a multistage unit that injects and thoroughly mixes the superheated steam with bitumen resulting in a solution at some homogeneous temperature. High pressure steam is injected directly into the bitumen froth flowing in a pipeline where initial contact takes place. The heaters allow direct contact between the two inputs, and the steam loses its heat to the froth and condenses at the bottom. The reduction in bitumen viscosity allows the release of entrapped air. The froth passes through heater-1 (H-1) and then through heater-2 (H-2) and the heated bitumen froth from H-2 is discharged to a downstream holding tank (Tank-1) and then sent for further froth treatment.

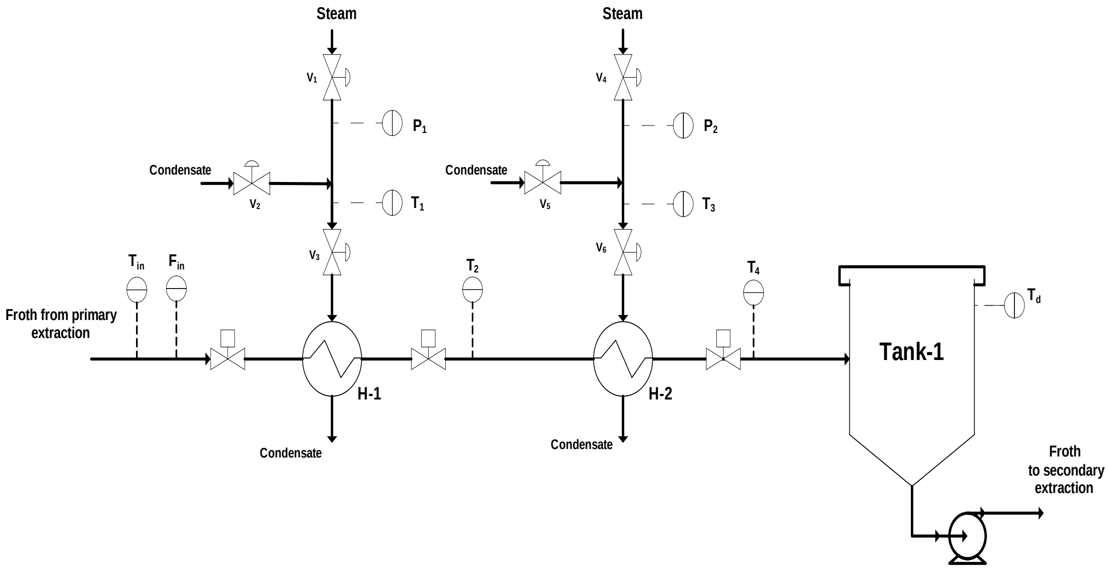


Figure 6.1: Flowsheet of the froth steam heater process

Pressure control valves V_1 and V_4 are designed to control the pressure of the input steam measured using pressure transmitters P_1 and P_2 respectively. The high pressure steam is mixed with the condensate to reduce the steam temperature to its saturation point and the temperature control valves V_2 and V_5 provide the control for the temperatures of the steam entering the heaters which are monitored by temperature sensors T_1 and T_3. Temperature control valves V_3 and V_6 are used to maintain the temperatures of the froth exiting from H-1 and H-2 respectively. In current operations, the steam pressure is controlled using two PID loops with V_1 and V_4 as the manipulating variables. Also, exit temperatures of the froth from H-1 and H-2 are controlled by manipulating the V_3 and V_6 respectively. The changing nature of the incoming froth feed flow and the froth temperature are disturbances to the process and affects the dynamics of the froth heaters.

6.3 Model development for the froth heater process

The process clearly represents a multi-input multi-output process with significant process interactions between various control loops. Modeling this MIMO process is a big challenge. The froth heater process is highly complex and involves direct contact of the bitumen froth with the steam; it is a tedious task to model this process using first principles. The process dynamics change with the quality of feed and the feed conditions (flow and temperature). The objective is to design linear MPC; and thus data based input-output models are identified for the MIMO froth heater process. To build an accurate model using input-output data, it is necessary to excite the plant and collect a rich data set. The controlled identification tests were done at different operating conditions to collect data and build models.

6.3.1 Linear time invariant (LTI) modeling

The process is nonlinear, however local linear models at different operating conditions can capture the dynamics of the process. After talking to the plant operators and plant engineers, the operating conditions inside the froth heaters differ mainly because of the input feed conditions (both flow and temperature). The operating condition, in regard to the temperature of the input bitumen froth being feed to the froth heaters, does not change for a long period of time. Hence, we are looking at developing linear models at different operating conditions with respect to the input froth temperature.

Let Σ_{nl} describes the nonlinear plant dynamics:

$$\Sigma_{nl} : \left\{ \begin{array}{c} \dot{x}(t) = f(x(t), u(t)) \\ y(t) = g(x(t), u(t)) \end{array} \right\} \tag{6.1}$$

where $x(t) \in R^n$ represents the states of the system, $u(t) \in R^m$ is the input and $y(t) \in R^p$ denotes the measured output. When a nonlinear model is not available, a set of linear black-box models may be used to describe the process dynamics over the operating range of the process. A set of linear plant models $\Sigma(\rho)$ is given by

$$\Sigma(\rho) : \left\{ \begin{array}{c} \dot{x}(t) = A(\rho)x(t) + B(\rho)u(t) \\ y(t) = C(\rho)x(t) + D(\rho)u(t) \end{array} \right\} \qquad \rho \in \Xi \tag{6.2}$$

or

$$\Sigma(\rho) : \{\mathbf{G}(s, \rho)\} \qquad \rho \in \Xi \tag{6.3}$$

where ρ is the scheduling variable and is regarded as constant for modeling purpose, Ξ gives the operating envelope and $\mathbf{G}$ gives the process transfer function matrix. Therefore, we are looking at building these sets of linear models, $\Sigma(\rho)$. For this study, froth feed temperature (T_{in}) is used as the scheduling variable and the process model is represented in the transfer function form.

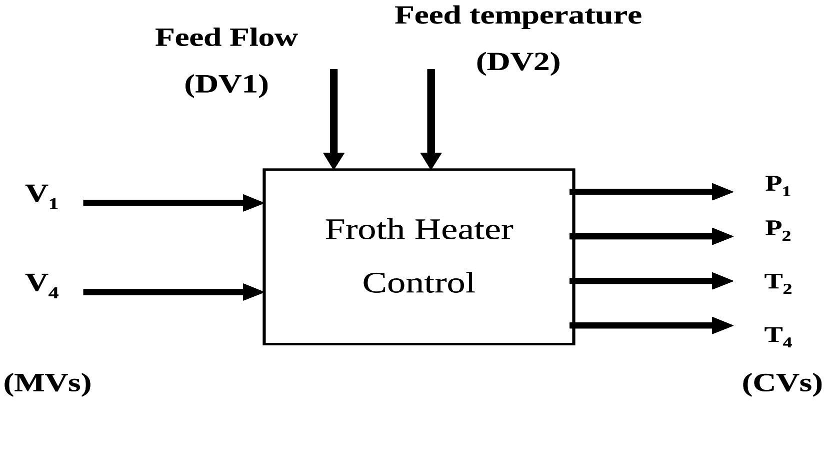
6.3.2 Model identification

Model identification experiments were conducted at three operating conditions with respect to T_{in}. The main objective is to monitor bitumen froth temperatures

T_2 and T_4 as well as the pressures, P_1 and P_2 of the steam lines. For heater-1, we have two control valves, V_1 and V_3 to monitor the steam flow to the heaters. After running some preliminary tests on the industrial setup and talking to operators to study the effect of these on T_2 and T_4, we found that small movements in control valve V_3 has no effect on bitumen froth temperature T_2. Also, in current operations, valve V_3 is kept almost fully open and exit temperatures from the two heaters are adjusted using valve V_1 and V_4 positions. Similar conclusion holds for T_4 for changes in valve position V_6.

During the model identification testing period, pressure transmitter P_1 was down for maintenance. After talking to process engineers, the steam line with valves V_1 and V_4 are very similar and since the source of the steam is the same for both the heaters, modeling pressure P_1 for changes in V_1 is essentially the same as modeling pressure P_2 for changes in V_4. Therefore, we can only build models for P_2 and models for P_1 are assumed to be the same as those for P_2. The valve positions V_2 and V_5 for condensate flow were kept constant during the experiments.

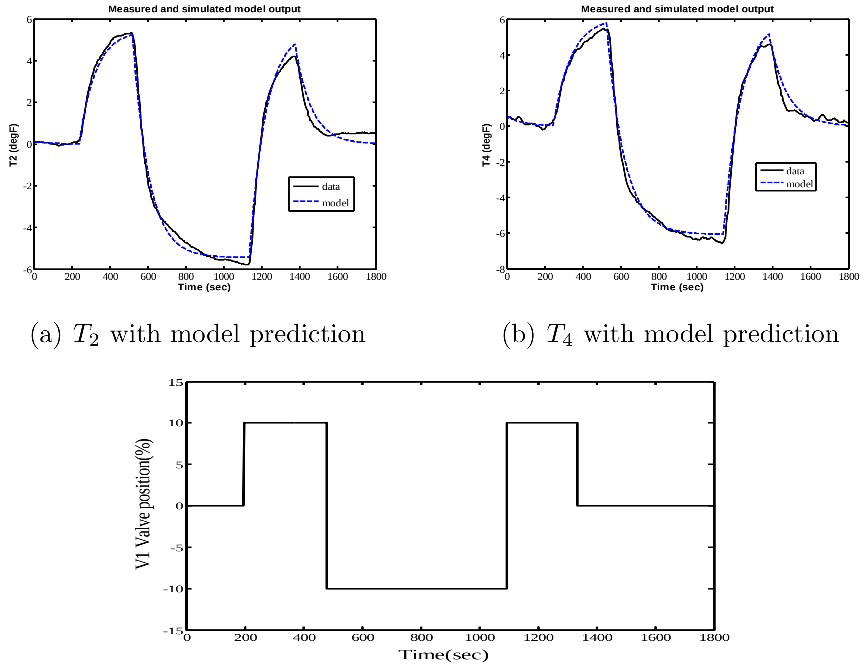
Fig. 6.2 shows the process structure for the MIMO control design where the controlled variables (CV) are P_1, P_2, T_2 and T_4 and the manipulating variables (MV) are valve V_1 and V_4 positions. Since the experiments are performed during normal operations, each manipulating variable is perturbed separately and in this way useful process information was gained.



Target requirement: Good regulatory performance

Figure 6.2: Process structure for MIMO control design

The input-output response data for each operating condition is used with the MATLAB system identification toolbox and CONTSID toolbox to identify best fit linear process models. Fig. 6.3, 6.4 and 6.5 present the plots for step tests for the operating condition 1 (OC-1) along with the model predictions.



(a) T_2 with model prediction

(b) T_4 with model prediction

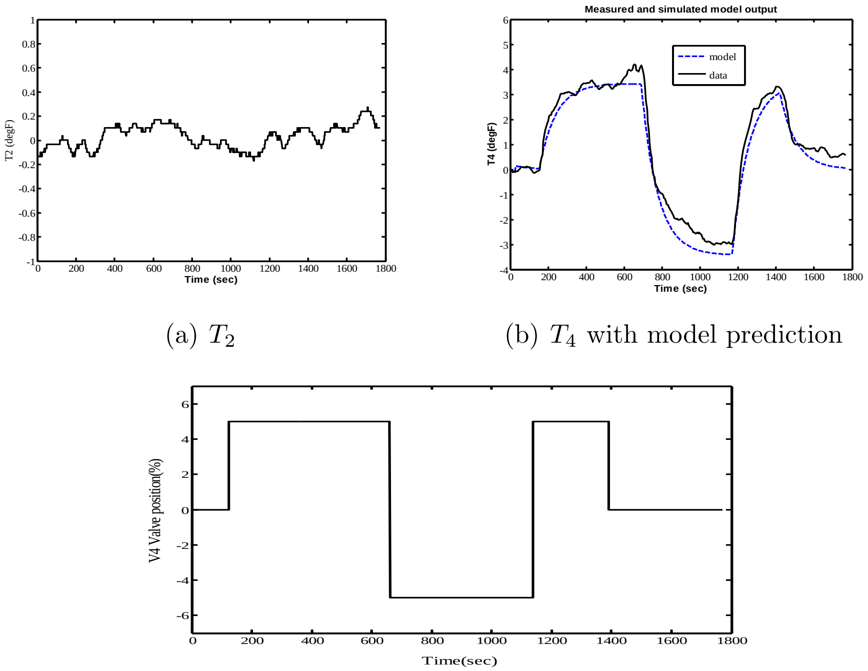
(c) Valve V_1 position (%)

Figure 6.3: Model validation at OC-1: T_2 and T_4 for changes in V_1

Table 6.1 shows the process models relating CVs with MVs for this MIMO process at operating condition-1. As V_1 is opened, the pressure P_1 goes up and P_2 goes down. Please note that these models are identified considering the fact that in industrial practice parsimonious models are preferred and these models captures the major transients and the time delays present in the process, which is typically sufficient to provide accurate model predictions and design of predictive control schemes.

Table 6.1: Process models at OC-1

	V_1 valve position (%)	V_4 valve position (%)
$\text{T}_2(^0\text{F})$	$\frac{0.54}{85.26s+1}e^{-35s}$	0
$\text{T}_4(^0\text{F})$	$\frac{0.60}{92.90s+1}e^{-40s}$	$\frac{0.69}{91.3s+1}e^{-20s}$
$\text{P}_1(\text{psi})$	$\frac{2.48}{9.1s+1}e^{-6s}$	$\frac{-0.12}{50s+1}e^{-5s}$
$\text{P}_2(\text{psi})$	$\frac{-0.12}{50s+1}e^{-5s}$	$\frac{2.48}{9.10s+1}e^{-6s}$

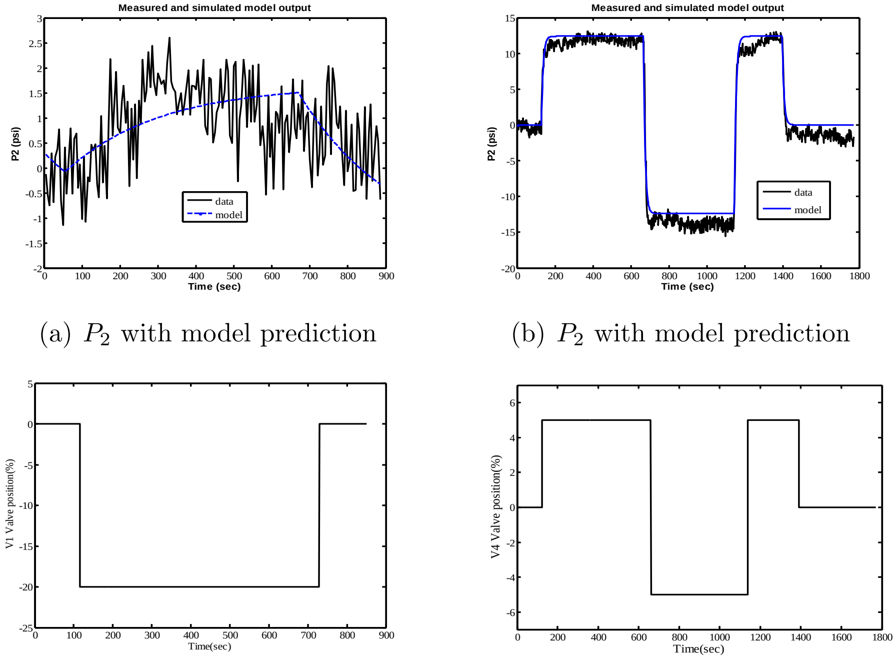


(a) T_2

(b) T_4 with model prediction

(c) Valve V_4 position (%)

Figure 6.4: Model validation at OC-1: T_2 and T_4 for changes in V_4



(a) P_2 with model prediction

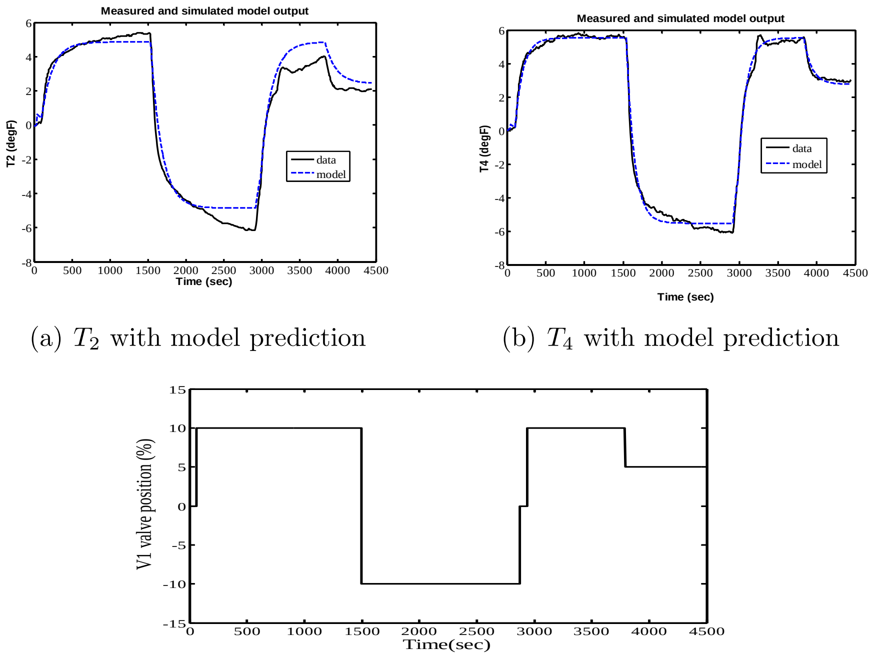
(b) P_2 with model prediction

(c) Valve V_1 position (%)

(d) Valve V_4 position (%)

Figure 6.5: Model validation at OC-1: P_2 for changes in V_1 and V_4

Figures 6.6, 6.7 and 6.8 present the plots for step tests for the operating condition-2 along with the model predictions. Table 6.2 presents the process models at operating condition-2 (OC-2).



(a) T_2 with model prediction

(b) T_4 with model prediction

(c) Valve V_1 position (%)

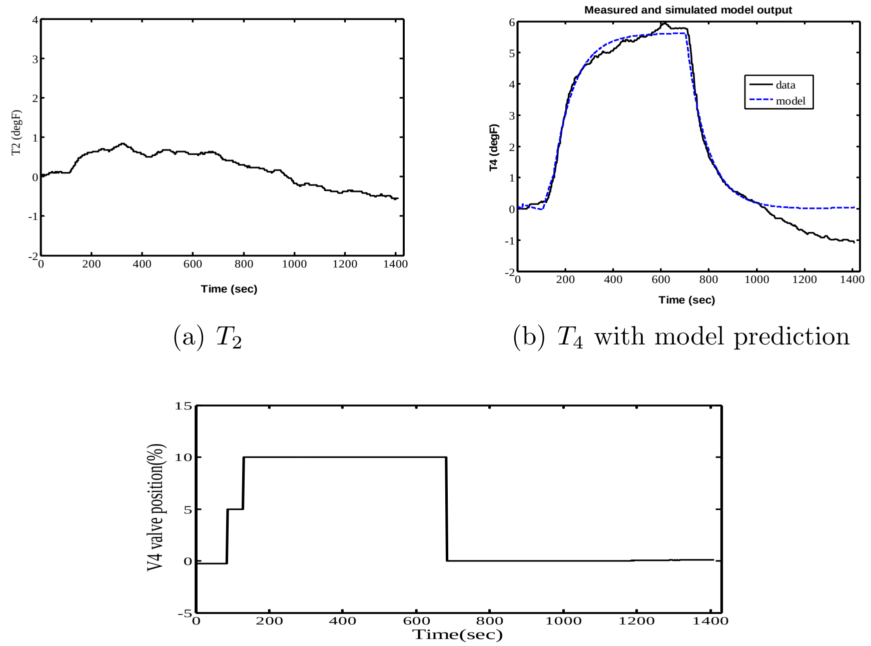
Figure 6.6: Model validation at OC-2: T_2 and T_4 for changes in V_1

Table 6.2: Process models at OC-2

	V_1 valve position (%)	V_4 valve position (%)
$\mathrm{T}_2(^0\mathrm{F})$	$\frac{0.49}{141.50s+1}e^{-35s}$	0
$\mathrm{T}_4(^0\mathrm{F})$	$\frac{0.55}{101.77s+1}e^{-40s}$	$\frac{0.56}{85.70s+1}e^{-20s}$
$\mathrm{P}_1(\mathrm{psi})$	$\frac{2.83}{7.35s+1}e^{-6s}$	$\frac{-0.09}{33s+1}e^{-5s}$
$\mathrm{P}_2(\mathrm{psi})$	$\frac{-0.09}{33s+1}e^{-5s}$	$\frac{2.83}{7.35s+1}e^{-6s}$

Figures 6.9, 6.10 and 6.11 present the plots for step tests for the operating condition-3 along with the model predictions. Table 6.3 presents the process models at OC-3.

Figures 6.12 presents the inlet froth temperature at the three operating conditions. The inlet temperature is fairly constant during the experiments at particular

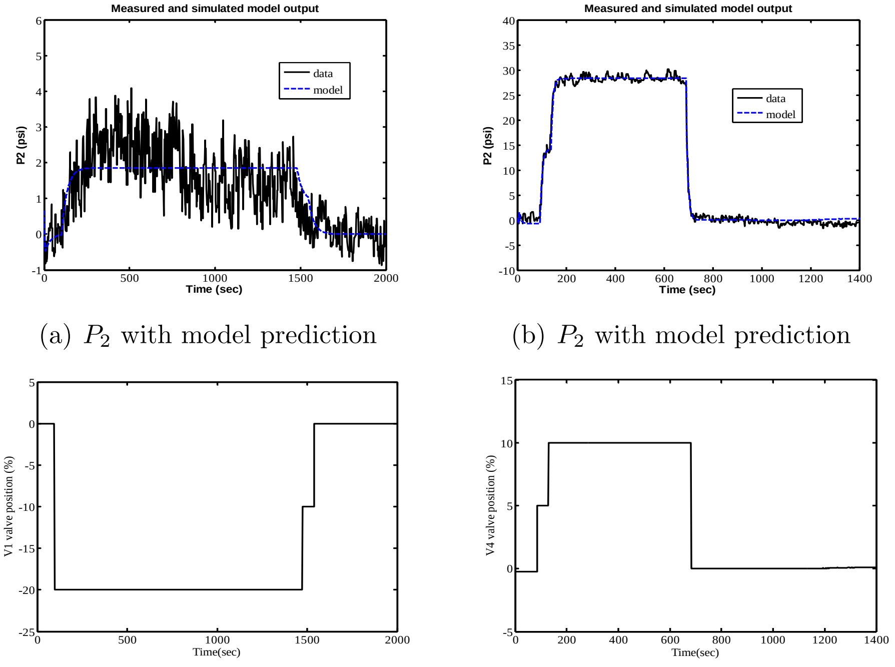


(a) T_2

(b) T_4 with model prediction

(c) Valve V_1 position (%)

Figure 6.7: Model validation at OC-2: T_2 and T_4 for changes in V_4



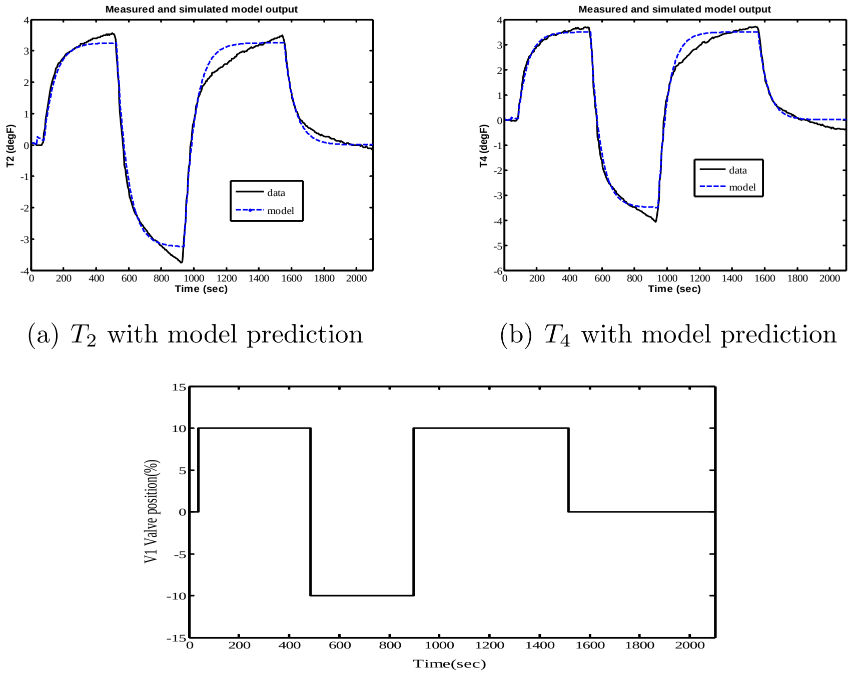
(a) P_2 with model prediction

(b) P_2 with model prediction

(c) Valve V_1 position (%)

(d) Valve V_4 position (%)

Figure 6.8: Model validation at OC-2: P_2 for changes in V_1 and V_4

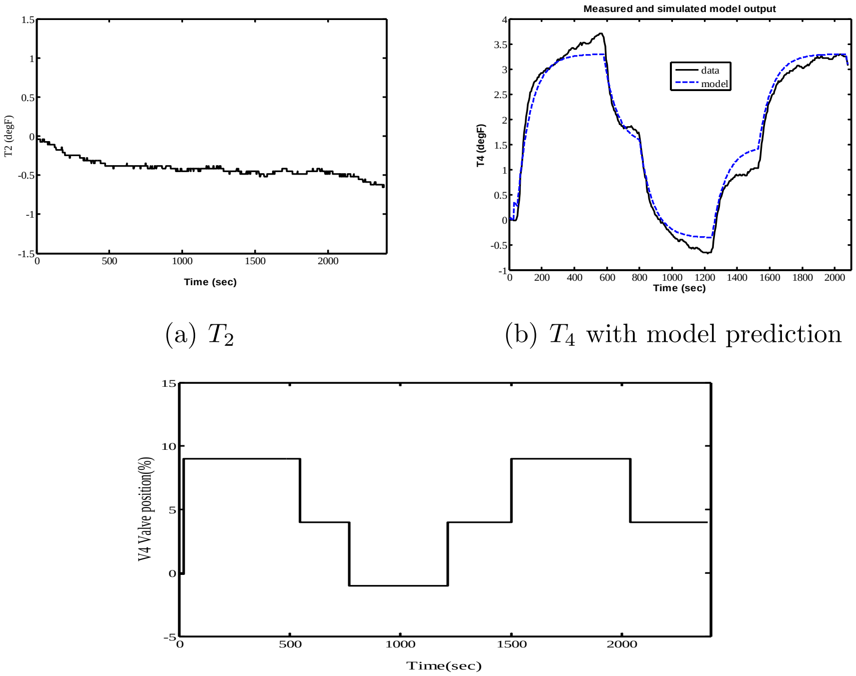


(a) T_2 with model prediction

(b) T_4 with model prediction

(c) Valve V_1 position (%)

Figure 6.9: Model validation at OC-3: T_2 and T_4 for changes in V_1

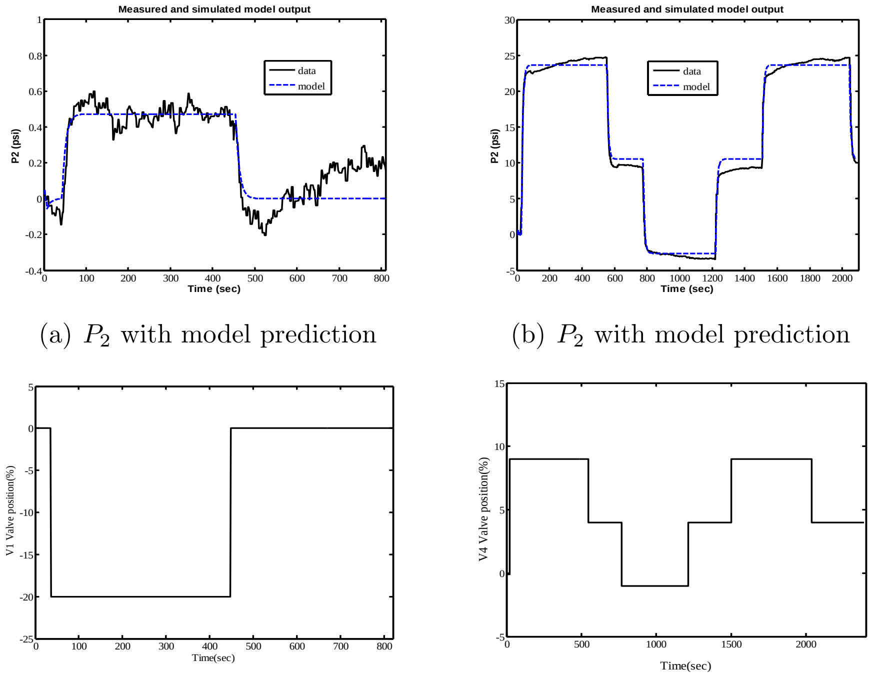


(a) T_2

(b) T_4 with model prediction

(c) Valve V_1 position (%)

Figure 6.10: Model validation at OC-3: T_2 and T_4 for changes in V_4



(a) P_2 with model prediction

(b) P_2 with model prediction

(c) Valve V_1 position (%)

(d) Valve V_4 position (%)

Figure 6.11: Model validation at OC-3: P_2 for changes in V_1 and V_4

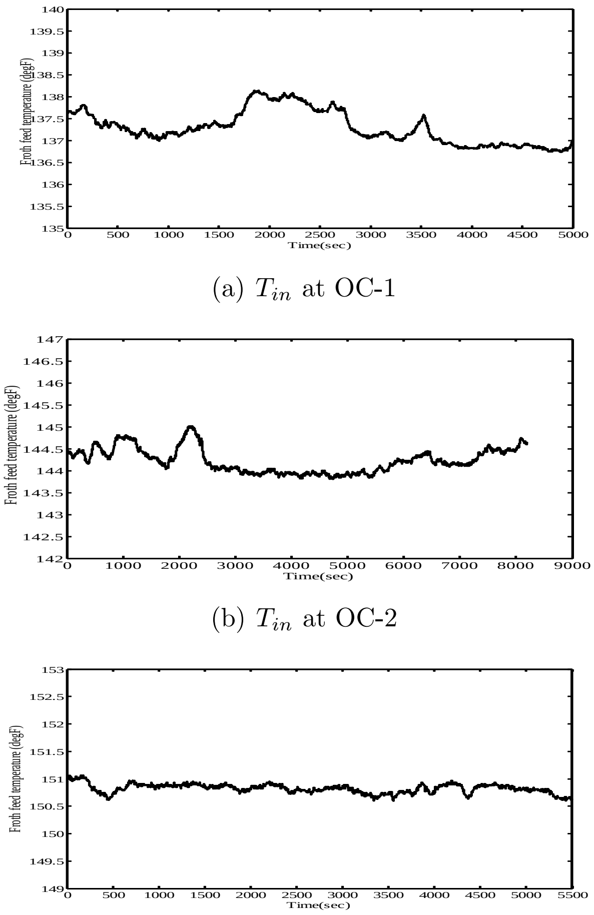
Table 6.3: Process models at OC-3

	V_1 valve position (%)	V_4 valve position (%)
$\text{T}_2(^0\text{F})$	$\frac{0.33}{66.90s+1}e^{-35s}$	0
$\text{T}_4(^0\text{F})$	$\frac{0.35}{59.94s+1}e^{-40s}$	$\frac{0.37}{82.77s+1}e^{-20s}$
$\text{P}_1(\text{psi})$	$\frac{2.62}{8.1s+1}e^{-6s}$	$\frac{-0.03}{10s+1}e^{-5s}$
$\text{P}_2(\text{psi})$	$\frac{-0.03}{10s+1}e^{-5s}$	$\frac{2.62}{8.1s+1}e^{-6s}$

operating condition. Thus, ρ value at different operating conditions is given in Table 6.4.

Table 6.4: ρ at different operating conditions

Process condition	$\rho(^0F)$
OC-1	137.5
OC-2	144.5
OC-3	151



(a) T_{in} at OC-1

(b) T_{in} at OC-2

(c) T_{in} at OC-3

Figure 6.12: Froth feed temperature, T_{in} at the three operating conditions

As can be seen from Tables 6.1, 6.2, 6.3, the process gain increases as the temperature of the incoming froth feed goes down. Figure 6.13 presents the process gain for changing incoming froth feed temperature. It is difficult to heat the bitumen froth if it is coming at a lower temperature and therefore, the magnitude and the direction of the process gain relating the temperature to steam flow is as we expect. The process delays are the same at all the operating conditions because the time delays are not affected by changes in froth feed conditions; however, the changes in the time constant of the process can be explained in terms of change in dynamics at these three operating conditions. Now the objective is to design local linear controllers at these three operating conditions and make them robust enough around the ρ value for that operating condition.

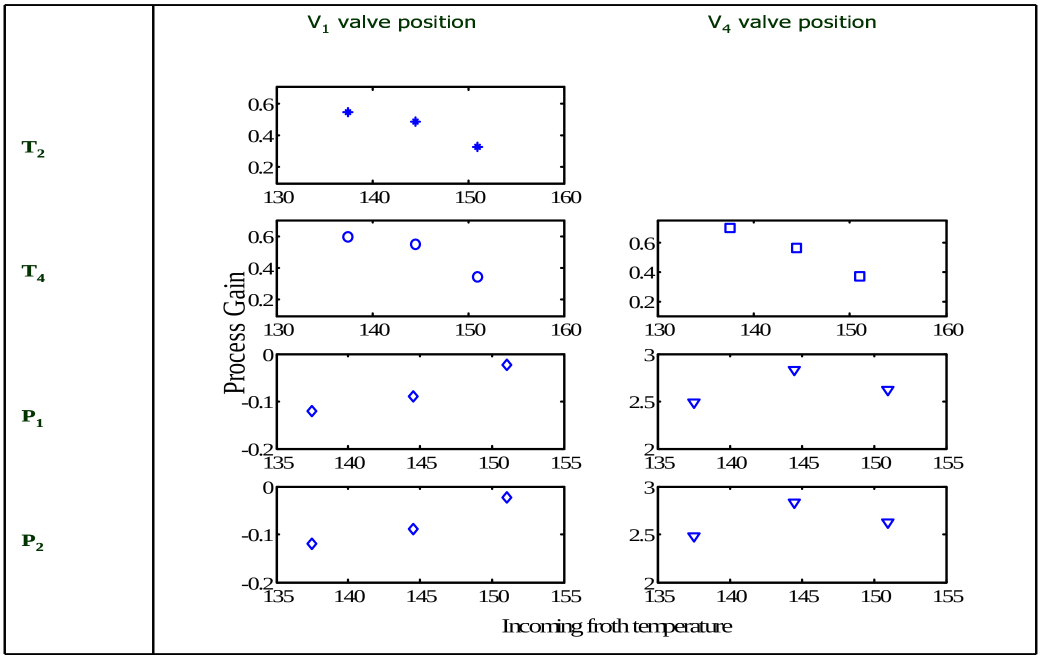


Figure 6.13: Process gain variation with incoming froth feed temperature

6.4 Gain scheduling MPC

Since the oil sands extraction process is a continuous process, a good temperature control in the froth heater process results in favorable conditions in the later stages of the process. Stabilization of the exit froth temperature and reduction in the steam valve movement results in significant steam energy consumption and thus huge energy savings.

Multiloop PID controllers with feedforward action for incoming froth temperature will not be able to handle this highly nonlinear process and the interactions between the two heaters. In contrast, MPC can handle the process constraints and the interactions in a much better way.

The design of a gain scheduling controller involves the following steps:

1. A set of LTI approximations of a nonlinear plant at constant operating conditions, parameterized by constant values of the scheduling variable ρ, is computed. It is assumed that the scheduling variable is measured.

2. LTI controllers corresponding to the derived set of local LTI models are designed to achieve required closed-loop performance and robustness. These controllers are also parameterized by the same scheduling variable, ρ.

3. Design the scheduling algorithm.

4. Implement the family of LTI controllers such that the controller coefficients are scheduled based on measured value of the scheduling variable.

The linear models are already built in Section 6.3. Based on the models developed at different operating conditions, a set of LTI controllers $\Pi(\rho)$ is designed, which is parameterized by the same scheduling variable ρ. The same operating conditions as defined in the modeling $\rho \in \Xi$ are used. The parameterized set of linear controllers is defined by:

$$\Pi(\rho) : \{\mathbf{G_c}(s, \rho)\} \quad \rho \in \Xi \tag{6.4}$$

$\mathbf{G_c}$ represents the controller, in our case $\mathbf{G_c}$ is a model predictive controller. If required, the controllers $\Pi(\rho)$ can be designed to provide different closed-loop dynamics at different operating points.

In case of discrete scheduling, the implementation of the LTI controllers involves the design of a scheduled selection procedure that is applied to the set of LTI controllers, rather than the design of a family of scheduled controllers. This comes down to the definition of regions for which members of the set of LTI controllers are valid. However, discontinuities (jumps) may appear in the controlled output, or in the controller coefficients in case of discrete scheduling of the coefficients rather than the total controller. In practice, ad hoc interpolation of the local point controllers is adopted to arrive at a gain-scheduled controller.

A class of control algorithms that use dynamic process models to incorporate process knowledge into the controller structure are known as predictive control algorithms. The Generalized Predictive Control (GPC) method proposed by Clarke et al. (1987) is a reasonable representative of model based predictive control (MPC) methods and one of the most general way of posing the process control problem in time domain. Model predictive controllers are good for highly complex and interconnected dynamic systems. In contrast to a standard PID controller, MPC calculates an optimal control action by taking into account process dynamic behavior and operational constraints (Qin and Badgwell, 2003; Rawlings, 2000). MPC can be an effective tool to design a gain scheduling policy for these systems. In fact, bounds in parameter variations can be explicitly considered in the optimization step of MPC, guaranteeing stability, robustness and performance.

Section 5.4.2 in Chapter 5 presented some discussion on MPC. In the overall operation of the process, especially for this process, MPC presents significant advantages over other multi-variable controllers which primarily include handling the interactions and the flexible constraint handling capacity. Also, for this process we have only two MVs compared to four CVs. The MPC algorithm is modified to allow set point tracking for T_2 and T_4 while do a constraint control for P_1 and P_2. Only if either P_1 or P_2 violates the constraint, the objective function penalizes the violation and MVs move, otherwise MVs change only to maintain T_2 and T_4 at their respective set points. Again, the real economic benefits are more prominent when real dynamic control allows the set points to be moved closer to the constraints without violating them; this is something we expect to achieve in the froth heater process by using MPC.

6.5 Control strategy applied in simulations

A number of off-line simulation studies were undertaken to identify suitable parameters for the controllers at the three operating conditions and get the required closed loop performance. The controllers have to be tested and implemented at the extraction unit at SUNCOR Energy Inc., Fort McMurrary, Albert, Canada but so far we have simulation results and which are presented here. Honeywell/Matrikon's Controller performance optimizer (CPO) is used to do offline simulations for MPC.

6.5.1 MIMO gain scheduling MPC

Using MPC, we added process constraints in all the MVs and CVs. Since we have less MVs compared to CVs, only the temperature loops, T_2 and T_4 are kept under set point control while range control is carried out for the pressure loops P_1 and P_2. The mode of operation is defined using the scheduling variable measurement and they are defined as: '0' for operating condition-1, '1' for operating condition-2 and '2' for operating condition-3.

Fig. 6.14 presents simulation results for the gain scheduling MPC using the set of identified models. The scheduling variable is T_{in}. For this simulation, no plant-model mismatch is assumed that is the plant is assumed to be the same as the process model. The three MPC controllers are tuned to provide sufficient robustness locally around their operating conditions. If the incoming froth is at a lower temperature, the set point of T_2 is also lowered to allow equal steam usage from both the heaters. The set point for T_4 and the incoming froth temperature, T_{in}, is used to calculate the set point for T_2. This also helps maintain the pressure measurements, P_1 and P_2, in the two steam lines. As expected we get perfect set point tracking.

Although scheduling from one controller to another is discrete, it has been kept smooth and there is no abrupt jump in the MVs. However, for simulation, since the process is represented as a set of models, when there is a change in operating conditions, there is a switch from one model to another, which sometimes causes sudden jump in CVs. This however will not be encountered for testing using real process which is continuous. This can even be alleviated by using a weighted sum of MVs from the three models.

The industrial data set for current operations which uses a couple of localized PID controllers is shown in Fig. 6.15. This data is from real current operations. In current operations, P_1 is controlled by manipulating valve V_1 position and P_2 is controlled by manipulating valve V_4 position. The temperature loops for T_2 and T_4 are run in the manual mode. As can be observed from the figure, valves V_1 and V_4 are used to maintain pressures as well as do some control of the temperature.

The offline simulation uses 'process' to be the same as the 'model'; we tried to generate results using the gain scheduling MPC controller on how it would have performed had it been online. The perfect model conditions are assumed and the real industrial data for T_{in} (Fig. 6.14 (c)) is used to simulate different operating conditions. Since in our control scheme, we have scheduling variable (T_{in}) which takes discrete values, we defined the range for T_{in} which corresponds to a particular operating condition and the controller at that operating condition is used for control purpose. Also, T_{in} is filtered to remove high frequency noise. As can be seen from Fig. 6.16, the designed gain scheduling MPC handles the switching from OC-2 to OC-3 really well. There is smooth transition of MV from one controller to another and we obtain perfect set point tracking. The pressure transmitter readings P_1 and P_2 in simulations are way off compared to the real industrial settings probably because in current operations these transmitters have been re-calibrated again.

Fig. 6.17 presents another case using the industrial data for T_{in}. For this study, change in the set point for T_2 does not allow MVs to move since the pressure P_1 hits the upper constraint. Maintaining the pressure within the limits is higher priority compared to set point tracking for the outlet temperature from the heater. These simulation results are encouraging and present an excellent opportunity to do some real testing on the industrial setup. The comparison of the results from the gain scheduling MPC with the current operations would justify the need for this multivariable application.

6.6 Economic benefits and Conclusions

MPC can exploit the interactions and the variation in plant conditions and can help reduce the variance of the froth temperature and pressure in the presence of changing operating conditions compared to using a couple of PID controllers. If the variability of manipulated variables can be reduced, it results in reduction in steam usage, leading to steam energy savings; and good temperature control results in smoother operation in the downstream.

In this Chapter, a case study application of identification and model based predictive control design for the bitumen froth steam heater (FHC) process is presented. The process is a multivariable application example and presents the challenge of regulating more controlled variables than manipulating variables, that is there are not enough degrees of freedom to asymptotically regulate all the controlled variables. A gain scheduling model predictive control (MPC) using linear models is designed and tested in simulations. The advantage of the multivariable scheme compared to current operations which uses localized PID controllers is very obvious. The industrial testing would help in establishing the benefits of using gain scheduling controllers for this application.

Bibliography

J. Masliyah, Z. Zhou, Z. Xu, J. Czarnechki and H. Hamza. Understanding water-based bitumen extraction from Athabasca oil sands. *The Canadian Journal of Chemical Engineering*, vol. 82, pp. 628-654, 2004.

K. A. Clark. Hot-water separation of Alberta bituminous sand. *Transactions of Canadian Institute of Mining and Metallurgy* , vol. 47, pp. 257-274, 1944.

M. A. Henson. Nonlinear model predictive control: Current status and future directions. *Computers and Chemical Engineering*, vol. 23(2), pp. 187-202, 1998.

S.J. Qin and T.A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, vol. 11, pp. 733764, 2003.

J. Rawlings. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, vol. 20, pp. 38-52, 2000.

M. Morari and J. H. Lee. Model predictive control: past, Present and Future. *Computers and Chemical Engineering*, vol. 23, pp. 667-682, 1999.

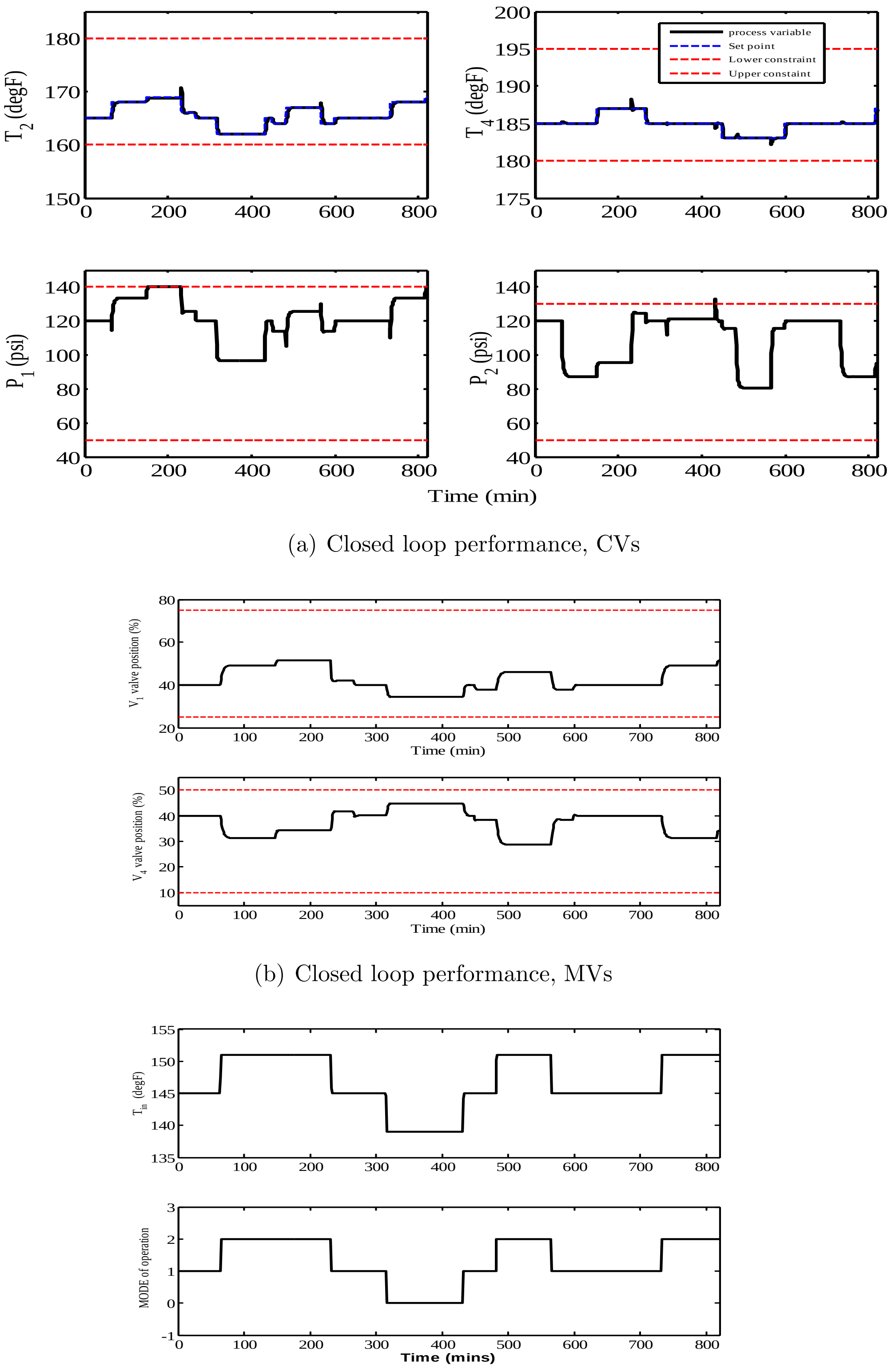
D. J. Leith and W. Leithead. Survey of gain-scheduling analysis and design. *Int. J. Control*, vol. 73, pp. 1001-1025, 2000.

L. Gaston, D. N. Madge, W. L. Strand, I. Noble, W. N. Garner and M. Lam. Bituminous froth inline steam injection processing. *U.S. Patent 7,914,670 B2*, issued March 29, 2011.

J. Shamma and D. Xiong. Set-valued methods for linear parameter varying systems. *Automatica*, vol. 35, pp. 1081-1089, 1999.

L. Ljung. System identification: Theory for the users. *Prentice-Hall*, 2d ed., New Jersey, 1999.

D. W. Clarke, C. Mohtadi and P.S. Tuffs. Generalized predictive control- Part I. The basic algorithm. *Automatica*, vol. 23(2), pp. 137148, 1987.

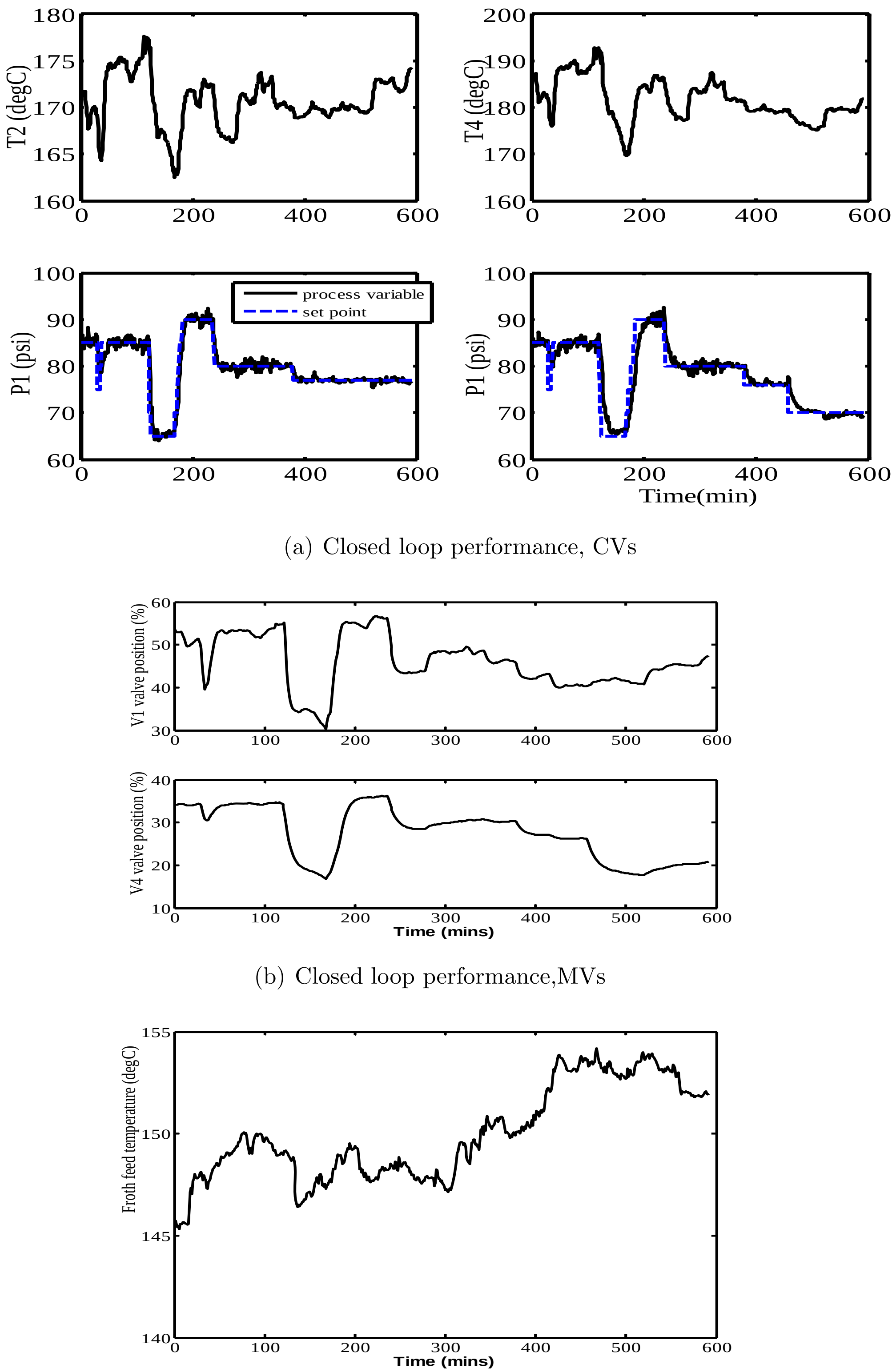


(a) Closed loop performance, CVs

(b) Closed loop performance, MVs

(c) Scheduling variable (T_{in}) and mode of operation

Figure 6.14: MIMO MPC for froth heater process-simulation 1

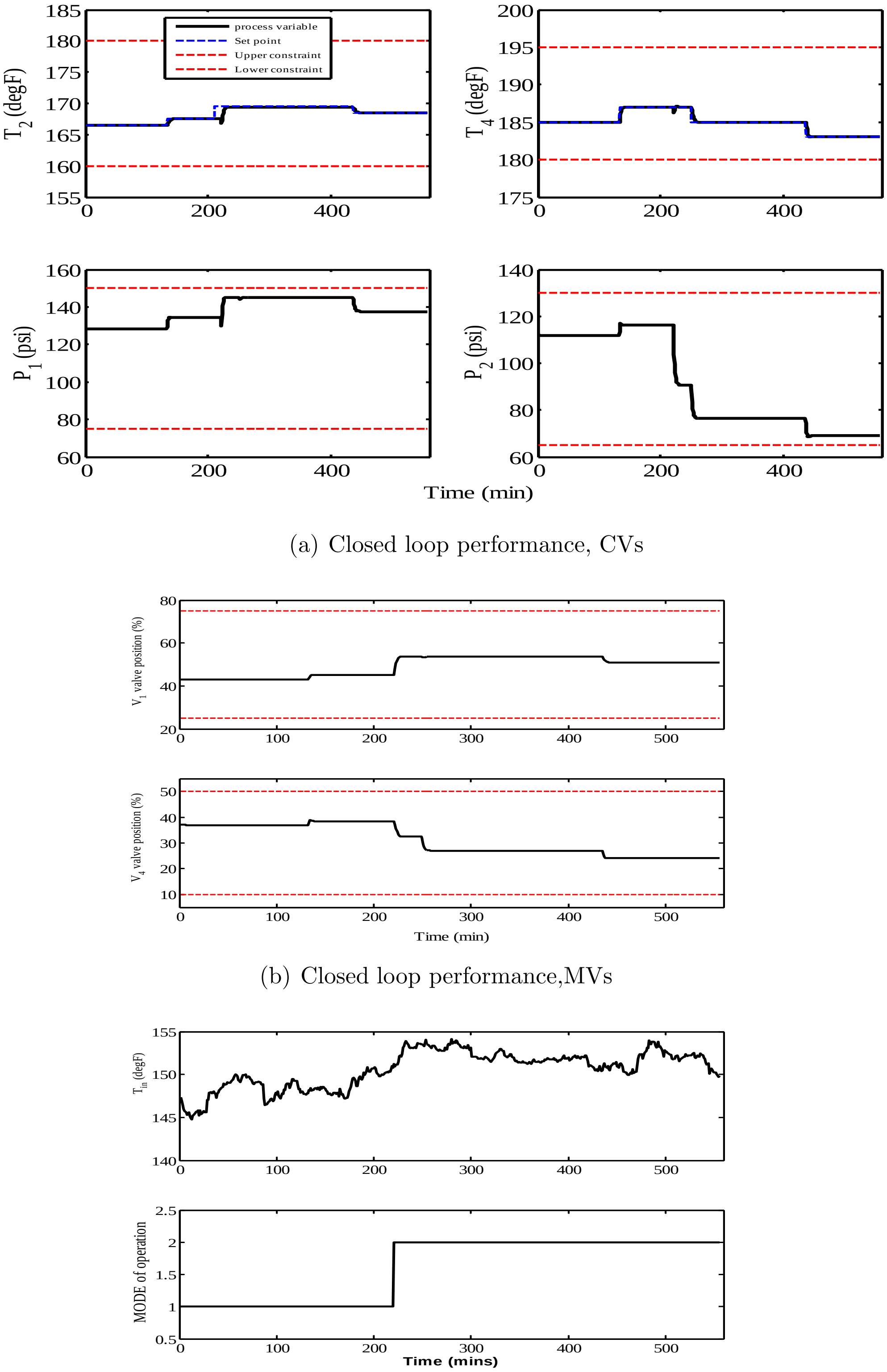


(a) Closed loop performance, CVs

(b) Closed loop performance,MVs

(c) Scheduling variable (T_{in})

Figure 6.15: Current operations for FHC: Industrial data

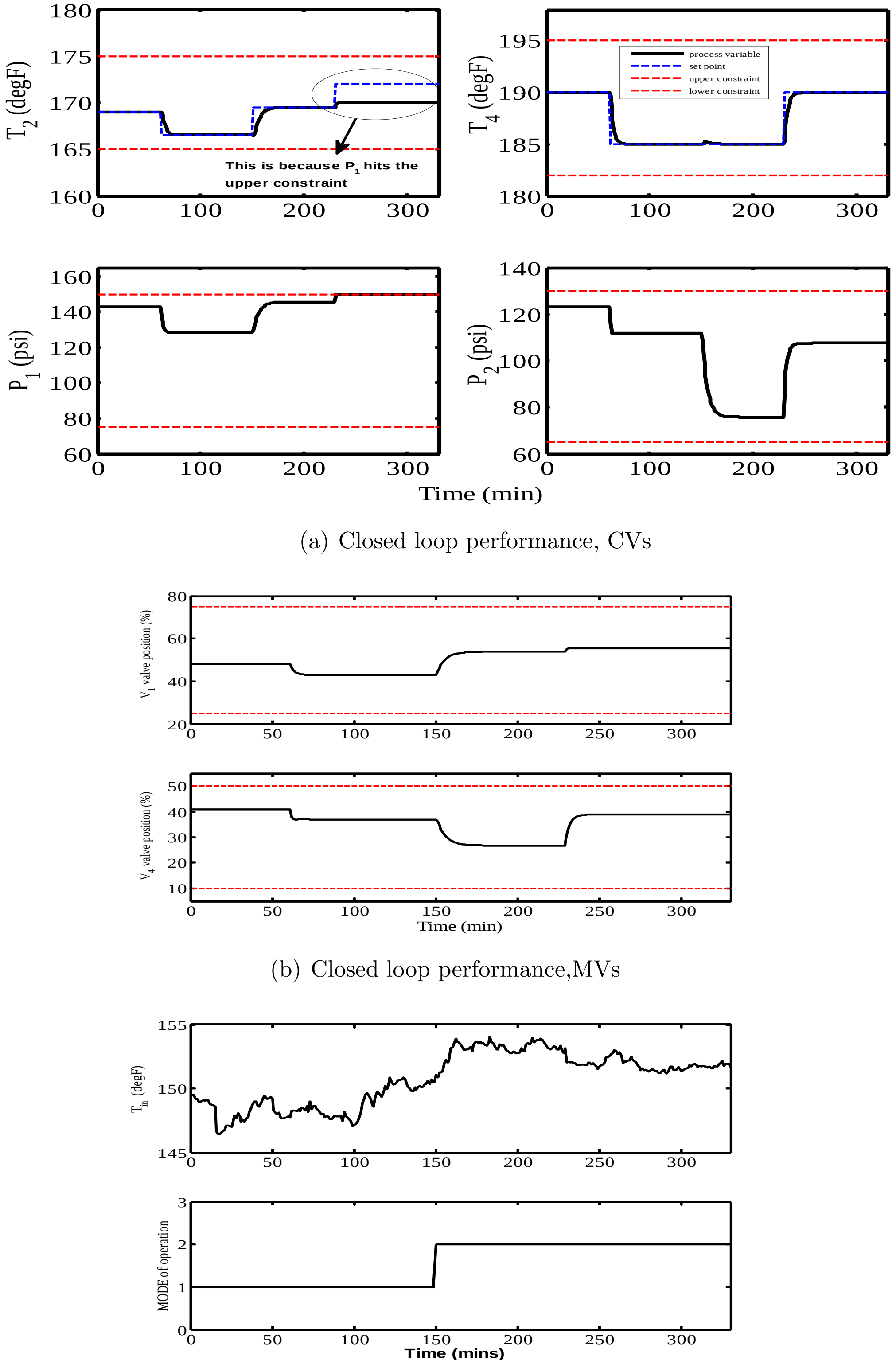


(a) Closed loop performance, CVs

(b) Closed loop performance,MVs

(c) Scheduling variable (T_{in}) and mode of operation

Figure 6.16: MIMO MPC for froth heater process-simulation 2



(a) Closed loop performance, CVs

(b) Closed loop performance,MVs

(c) Scheduling variable (T_{in}) and mode of operation

Figure 6.17: MIMO MPC for froth heater process-simulation 3

Chapter 7

Concluding Remarks and Future Work

7.1 Major Thesis Contributions

This thesis is concerned with the identification and control of chemical processes and the main contributions of this thesis are summarized below:

- **Chapter 2** presented a continuous-time identification algorithm for commensurate fractional order models with time delays. The proposed method was developed for the following cases: (a) using step input excitation, (b) using RBS/PRBS or any other kind of input signal excitation. The proposed algorithm estimates the time delay along with constant model parameters in an iterative way in the inner loop and the unknown fractional order in the outer loop. The efficacy of the proposed algorithm was demonstrated by performing Monte Carlo simulation analysis on the examples. The proposed algorithm was also applied in simulations on a fractional order system of a classical wall heat transfer problem, which is described by fractional behavior. Using the proposed algorithm, this process was represented as a parsimonious fractional order model with a delay term.

- An algorithm for tuning fractional order PI controller using the Bode's ideal transfer function for fractional order and integer order systems was presented in **Chapter 3**. A control strategy based on a reference model, whose open-loop transfer function is given by the Bode's ideal transfer function was proposed where the parameters of the controller are estimated by formulating a constrained non-linear optimization problem. Global optimization tool, particle swarm optimization, was used to solve the optimization problem.

Simulation studies were presented and analyzed to illustrate the effectiveness and efficacy of the proposed algorithm. The proposed algorithm was also applied in simulation to design an FO-PI controller for the classical wall problem. Also, the proposed method is extended to examples for lag/delay dominant processes. The efficacy of the controller tuned using the proposed algorithm was also demonstrated on an experimental setup of a continuous stirred tank heater process. For CSTH, the FO-PI controller designed using the proposed algorithm was compared with some of the standard methods available in literature for designing FO-PI as well as traditional PI controllers and was shown to give better closed loop performance than other tuning methods.

- The application of the identification method proposed in Chapter-2, was applied on distributed parameter systems in **Chapter 4** . Three application studies were considered for the following single-input single-output DPS examples: (a) a simulated heat transfer wall problem, (b) an industrial scale froth heater, and (c) a laboratory scale heater mixer plus pipe flow setup. Both simulation and experimental studies are carried out to build fractional order models. The unknown DPS was treated as a black box and both fractional order as well as continuous-time rational order (integer) models were identified from input-output perturbation data. For the applications presented, fractional order models with time delay were found to be very effective to deal with infinite order distributed parameter systems.

- **Chapter 5** presented a case study application of identification and model based predictive control design for the interface level control in the separation cell process, found in the oil sands industry. Handling large dead time and other process constraints using internal model control and model predictive control was discussed. Both IMC and MPC were designed, implemented and tested in real time on the industrial separation cell to observe the benefits of using advanced process control schemes in the extraction unit. Using the real testing data on the industrial setup, it was shown that both IMC and MPC schemes provide significant benefits over the the current operations which uses PID (plus manual) control. These benefits included significant reduction in the variance of the interface level and underflow pump movement, further resulting in the pump energy savings and smoother operation downstream.

- **Chapter 6** presented the second case study application of identification and gain scheduling model predictive control design for the bitumen froth steam heater process. The process represented a non-square multi-variable problem with more controlled variables than manipulating variables and a gain scheduling MPC using linear models was designed and tested in simulations. The simulation results were very encouraging compared to current operation which uses a number of local PID controllers. Industrial implementation of the gain scheduling MPC could help in reducing the variance of the exit froth temperature and steam usage, resulting in steam energy savings.

7.2 Directions for Future Work

There is still scope for further exploration on the work presented in this thesis and in this section, I would like to share our perspective on the fields and directions that are worthy of future investigation.

1. In Chapter 2, we discussed the identification method for commensurate order fractional order models with time delay. Although, for most of the fractional order systems, this type of transfer functional form is encountered, the limitation of this representation is that all fractional powers in the model are integer multiples of a single real number. Extending the proposed algorithm to non-commensurate models is not very straight forward as it further complicates the formulation, especially the outer nonlinear loop; but at the same time the advantage is that each fractional powers is chosen independently. The discussion to non-commensurate models would be worth exploring in future where the parameters from commensurate order models could be used as good initial guesses for these models.

2. From theory and as seen from analytical expressions in Chapter 4, we know that the diffusive interface examples like classical heat conduction in a wall or metal rod are ideal examples of fractional order systems. In Chapter 2 and Chapter 3, the proposed algorithm has only been applied in simulation on such examples. The interesting future perspective would be to carry out experimental evaluation of the proposed identification algorithm by designing an experimental setup for heat conduction in a wall or metal rod. If a laboratory scale process for fractional order process is available, other areas such as model based control design could be explored. Fractional order controllers could be designed and tested on the setup and could be compared with some

of the existing methods to design fractional order controllers. In addition to this, empirical relations could be derived to relate the tuning parameters of FPI controllers with the process model for both fractional and integer order systems.

3. For DPS we know that we obtain irrational transfer function, the model parameters are functions of some spatial variables. The area of continuous-time model identification for DPS has not been significantly explored. The fractional order models could provide a parsimonious representation for these systems and could tremendously help in designing the model based controllers. The algorithms for modeling using non-commensurate type models would be worth exploring in future for other DPS.

4. In oil sands industry, the APC applications are relatively new and provide tremendous opportunity to apply developed theory and tailor them as the requirement demands. The application of MPC could improve process efficiency by handling the constraints, process interactions and combining feed forward plus feedback control in the extraction units. Additional information from the local controllers could be added into the MPC setup to design a MIMO application. MIMO MPC testing on the SepCell by incorporating additional CVs and MVs would make the control problem more challenging. In the future, a MIMO MPC industrial testing on SepCell can be studied. Once MIMO MPC is implemented, we can explore how to optimally chose the set point of this controller.

5. The benefits of multivariable MPC for the froth heater process are very obvious. If the approval for online testing can be obtained, gain scheduling MPC testing on the real industrial setup for the froth heater process and its comparison with the current operations would be the next immediate task. A comparative study could then easily establish the benefits and the need for APC application in the oil sands industry.

7.3 Concluding Remarks

Overall, this thesis has presented results on identification and model based control design case studies on fractional order systems, distributed parameter systems and two industrial processes. The focus of the thesis has been that identification and control go hand-in-hand. The model lies at the heart of the controller; we can expect a high performance from a controller using a high fidelity model. New methods and new tools for building models and design control schemes will always be required to make life easier for process control engineers and practitioners. Also, with so much developed theory in process control, the application of these ideas on real complex chemical processes and tailoring them for the industrial need will have to be taken into account. Applied research of the existing and new methods in the area of model based control design is essential to improve current process operations and bring theory much closer to practice.

Appendix A

Particle Swarm Optimization

In PSO, suppose we have N particles in a D-dimensional search space, and each particle position be represented by a position vector $X_i = (x_{i1}, x_{i2}, ..., x_{iD})$, and a velocity represented by a velocity vector $V_i = (v_{i1}, v_{i2}, ..., v_{iD})$. Each particle remembers its best position (pbest) so far in a vector $P_i = (p_{i1}, p_{i2}, ..., p_{iD})$. Also, the best solution achieved so far by the whole swarm (gbest) is represented as $P_g = (p_{g1}, p_{g2}, ..., p_{gD})$. At each time step, each particle moves toward the pbest and gbest locations. The fitness function evaluates the performance of particles to determine whether the best fitting solution is achieved. The particles are updated according to the following equations:

$$\begin{aligned} v_{id}^{(k+1)} &= v_{id}^{(k)} + c_1 r_1 (p_{id} - x_{id}^{(k)}) + c_2 r_2 (p_{gd} - x_{id}^{(k)}) \\ x_{id}^{(k+1)} &= x_{id}^{(k)} + v_{id}^{(k+1)} \quad \forall \;\; d = 1, 2, ...D \end{aligned} \tag{A.1}$$

where: c_1 and c_2 are two positive constants, also known as acceleration constants; r_1 and r_2 are random numbers in the range of [0,1] and 'k' defines the current iteration count (k_{max} being maximum iteration count).

This type of PSO algorithm had shortcomings such as velocity control mechanism and no limits on the positions of the particles. An improved PSO was proposed to solve many shortcomings of the old PSO algorithm. The updated position and the velocity using the modified version of PSO (Liu et al. (2004)) are manipulated as

$$\begin{aligned} v_{id}^{(k+1)} &= w v_{id}^{(k)} + c_1 r_1 (p_{id} - x_{id}^{(k)}) + c_2 r_2 (p_{gd} - x_{id}^{(k)}) \\ x_{id}^{(k+1)} &= (1 - mc) x_{id}^{(k)} + (mc) v_{id}^{(k+1)} \quad \forall \;\; d = 1, 2, ...D \end{aligned} \tag{A.2}$$

where: 'mc' is the momentum factor (0< mc <1) and 'w' is the inertial factor.

The inertia factor, w, in general can be set according to the following equation:

$$w = w_{min} + \frac{(w_{max} - w_{min})}{k_{max}}(k_{max} - k) \tag{A.3}$$

This new improved PSO algorithm ensures that the particles are in the limits of the defined search space without checking the boundary at every iteration. The maximum velocity is set to 'max(X_i)' and minimum velocity is set to 'min(X_i)'.