

# UC Merced

## UC Merced Electronic Theses and Dissertations

### Title

SELF OPTIMIZING CONTROL STRATEGIES FOR THERMAL PROCESSES AND RF IMPEDANCE MATCHING

### Permalink

<https://escholarship.org/uc/item/3wf2j6qc>

### Author

Dehghan, Sina

### Publication Date

2021

### Copyright Information

This work is made available under the terms of a Creative Commons Attribution-NonCommercial-NoDerivatives License, available at <https://creativecommons.org/licenses/by-nc-nd/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA, MERCED

**SELF OPTIMIZING CONTROL STRATEGIES FOR  
THERMAL PROCESSES AND RF IMPEDANCE  
MATCHING**

by

Sina Dehghan

A thesis submitted in partial satisfaction of the  
requirements for the degree of  
Doctor of Philosophy

in

Mechanical Engineering

Committee in charge:  
Dr. YangQuan Chen, Advisor  
Dr. Jian-Qiao Sun  
Dr. Gerardo C. Diaz  
Dr. Ricardo Pintro de Castro

©2021 Sina Dehghan

©2021 Sina Dehghan  
All rights are reserved.

The thesis of Sina Dehghan is approved:

---

Dr. YangQuan Chen

Date

---

Dr. Jian-Qiao Sun

Date

---

Dr. Gerardo Diaz

Date

---

Dr. Ricardo Pinto de Castro

Date

University of California, Merced

©2021 Sina Dehghan

Dedicated to my parents, and my dearest wife, Arghavan.

## ACKNOWLEDGEMENTS

First and foremost I am extremely grateful to my PhD advisor, Prof. YangQuan Chen for his continuous support, invaluable advice, patience, and the great learning environment he has created in MESA Lab that offers huge practical experience through industry collaborations. Moreover, I am grateful to the rest of my dissertation committee: Dr. Gerardo Diaz, Dr. Jian-Qiao Sun, and Dr. Ricardo de Castro for their insightful comments and proofreading this dissertation. I would also like to thank all my labmates with whom I had the honor of collaborating with through this period; particularly Mr. Jairo Viola whose kindness and positive character has made the long days of working together enjoyable.

I am deeply grateful to my parents for inspiring me from the very young age to get into the path I am in and to enable me to do so through their continuous support. Finally, I would like to express my gratitude to my wife, Arghavan; she not only set a perfect example for me as she graduated from her PhD, but it would be impossible for me to complete my study without her endless encouragement and support.

# CURRICULUM VITAE

## Education

Ph.D. in Mechanical Engineering, University of California, Merced, 2021.  
B.S. in Mechanical Engineering, Isfahan University of Technology, 2013.

## Work Experience

- Full-Time Control Engineer at Lam Research Corporation, Deposition Product Group, Tualatin, Oregon. Main Responsibility: Providing Solutions for RF Sub-Systems Related Control Problems.
  - Control System Design and Support for Active Impedance Matching and Power Distribution in a Dual-Frequency Four-Station Deposition Tool.
- Electrical Engineer Summer Intern at Lam Research, Etching Product Group, Fremont, California. Topic: Advanced Control of a Single Load RF Matching Network for an Etching Process.
  - Developed MATLAB GUI and Tools for Data Analysis.
  - Developed a Gradient-Based Control Algorithm for RF Impedance Matching Problem for an Etching Process, Resulted in a Patent Submission.

## Publications

- Dehghan, S., Viola, J., Chen, Y., “Preview Control Based on RIOTS MPC and  $H_\infty$  for A Thermal Hardware in the Loop System”. Modeling, Estimation and Control Conference 2021. Online and UT Austin, USA, October 2021.
- Dehghan S, Zhao T, Chen Y, Homayouni T. Model Predictive Control of General Fractional Order Systems Using a General Purpose Optimal Control Problem Solver: RIOTS. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference 2019 Aug 18 (Vol. 59292, p. V009T12A014). American Society of Mechanical Engineers.

- Viola J, Radici A, Dehghan S, Chen Y. Low-cost real-time vision platform for spatial temperature control research education developments. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference 2019 Aug 18 (Vol. 59292, p. V009T12A030). American Society of Mechanical Engineers.
- Viola J, Dehghan S, Chen Y. Embedded RIOTS: Model Predictive Control Towards Edge. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference 2019 Aug 18 (Vol. 59292, p. V009T12A001). American Society of Mechanical Engineers.
- Zhao Y, Li Y, Dehghan S, Chen Y. Learning feedforward control of a one-stage refrigeration system. IEEE Access. 2019 May 10;7:64120-6.
- Muñoz-Hernández A, Dehghan S, Diaz G. Physical (Steam) Activation of Post-Gasification Biochar Derived From Peach Pits. In ASME International Mechanical Engineering Congress and Exposition 2018 Nov 9 (Vol. 52071, p. V06AT08A003). American Society of Mechanical Engineers.
- Dehghan S, Zhao T, Zhao Y, Yuan J, Ates A, Chen Y. PID2018 Benchmark Challenge: Model Predictive Control With Conditional Integral Control Using A General Purpose Optimal Control Problem Solver–RIOTS. IFAC-PapersOnLine. 2018 Jan 1;51(4):882-7.
- Yuan J, Ates A, Dehghan S, Zhao Y, Fei S, Chen Y. PID2018 Benchmark Challenge: Model-based Feedforward Compensator with A Conditional Integrator. IFAC-PapersOnLine. 2018 Jan 1;51(4):888-93.
- Ates A, Yuan J, Dehghan S, Zhao Y, Yeroglu C, Chen Y. PID2018 benchmark challenge: Multi-objective stochastic optimization algorithm. IFAC-PapersOnLine. 2018 Jan 1;51(4):877-81.
- Zhao Y, Dehghan S, Ates A, Yuan J, Zhou F, Li Y, Chen Y. PID2018 Benchmark Challenge: learning feedforward control. IFAC-PapersOnLine. 2018 Jan 1;51(4):663-8.
- Li Z, Liu L, Dehghan S, Chen Y, Xue D. A review and evaluation of numerical tools for fractional calculus and fractional order controls. International Journal of Control. 2017 Jun 3;90(6):1165-81.

#### Oral Presentations

- “Preview Control Based on RIOTS MPC and  $H_\infty$  for A Thermal Hardware in the Loop System”, *MECC 2021 online conference*, October 2021.

- “Study of the TCP-Match behavior during ignition and some solution proposals to improve the transient behavior”, *MESA Lab Research Seminar Series at UC Merced*, September 2019.
- “Model Predictive Control of General Fractional Order Systems Using a General-Purpose Optimal Control Problem Solver, RIOTS”, *ASME 2019 IDETC/CIE, Anaheim Convention Center*, August 2019.
- “Model Predictive Control of General Fractional Order Systems Using a General-Purpose Optimal Control Problem Solver, RIOTS”, *MESA Lab Research Seminar Series at UC Merced*, July 2019.
- “Analysis and Improvement of Kiyo TCCT Match Control”. *Lam Research Corporation, Fremont*, August 2018.
- “PID2018 Benchmark Challenge: Model Predictive Control Using A General-Purpose Optimal Control Problem Solver - RIOTS.” *CITRIS/CPAR Control Theory and Automation Symposium, UC Santa Cruz*, April 2018.
- “MESA-Box App: Inverted Pendulum”, *MESA Lab Research Seminar Series at UC Merced*, August 2015.

# TABLE OF CONTENTS

<b>ACKNOWLEDGEMENTS</b> . . . . .	<b>v</b>
<b>CURRICULUM VITAE</b> . . . . .	<b>vi</b>
<b>LIST OF FIGURES</b> . . . . .	<b>xiii</b>
<b>LIST OF TABLES</b> . . . . .	<b>xx</b>
<b>ABSTRACT</b> . . . . .	<b>xxi</b>

## Chapter

<b>1 INTRODUCTION</b> . . . . .	<b>1</b>
1.1 Overview . . . . .	1
1.2 Self Optimizing Control - A General Framework . . . . .	2
1.2.1 History . . . . .	2
1.2.2 The Proposed Framework . . . . .	2
1.2.3 Strategies for Realization . . . . .	3
1.3 Dissertation Outline . . . . .	4
<b>2 ITERATIVE LEARNING CONTROL FOR A LAB-SCALE PHYSICAL (STEAM) CARBON ACTIVATION PLANT</b> . . . . .	<b>6</b>
2.1 Introduction . . . . .	6
2.2 Problem Description . . . . .	6
2.3 Modeling and Simulation . . . . .	8
2.4 Proposed Solution . . . . .	10
2.4.1 Iterative Learning of Reference . . . . .	10
2.4.2 Control Algorithm . . . . .	10
2.4.3 A Control Toolbox . . . . .	11
2.5 Results and Discussions . . . . .	12

2.6	Conclusion . . . . .	13
<b>3</b>	<b>PLUG AND PLAY PLANT-READY MPC TOOLBOX, RMPC . . . . .</b>	<b>14</b>
3.1	Introduction . . . . .	14
3.2	Model Predictive Control (MPC) Formulation . . . . .	15
3.3	Recursive Integration Optimal Trajectory Solver (RIOTS) . . . . .	16
3.3.1	sys_acti.m . . . . .	16
3.3.2	sys_init.m . . . . .	17
3.3.3	sys_h.m . . . . .	17
3.3.4	sys_g.m . . . . .	17
3.3.5	sys_l.m . . . . .	17
3.3.6	sys_Dh.m, sys_Dl.m, sys_Dg.m . . . . .	17
3.4	RMPC Toolbox . . . . .	18
3.5	Examples of RMPC Implementation . . . . .	19
3.5.1	RMPC for PID 2018 Benchmark Challenge . . . . .	20
3.5.1.1	Problem Description . . . . .	20
3.5.1.2	Implementation Details . . . . .	21
3.5.1.3	Results and Discussion . . . . .	22
3.5.2	RMPC for General Fractional Order Systems . . . . .	25
3.5.2.1	Introduction . . . . .	25
3.5.2.2	Example 1: Comparison of RMPC for a Multi Fractional Order System Using Oustaloup Approximation and Time-Response-Invariant Approximation . . . . .	26
3.5.2.3	Example 2: Application of RMPC for a MIMO Commensurate Fractional Order System Based on Time-Response-Invariant Approximation Method. . . . .	35
3.6	RIOTS MPC with Preview for a Thermal Hardware-in-the-Loop System . . . . .	40
3.6.1	Introduction . . . . .	40
3.6.2	RMPC with Preview . . . . .	41

3.6.3	Case Study: A Peltier System . . . . .	42
3.6.3.1	MESA Lab Edge Computing Based Peltier Box . . . . .	42
3.6.3.2	System Identification . . . . .	43
3.6.4	Controller Implementation . . . . .	44
3.6.5	Results and Discussion . . . . .	46
3.7	Conclusion . . . . .	53
<b>4</b>	<b>SELF OPTIMIZING CONTROL OF AN RF MATCHING NETWORK FOR SEMICONDUCTOR MANUFACTURING APPLICATIONS . . . . .</b>	<b>55</b>
4.1	Introduction . . . . .	55
4.2	Problem Description . . . . .	56
4.2.1	Impedance Matching . . . . .	56
4.2.2	The Matching Network . . . . .	57
4.2.2.1	Reflection Coefficient Definition and Formulation . . . . .	58

4.2.2.2	Motorized Capacitors . . . . .	59
4.2.3	State-Space Formulation . . . . .	60
4.2.3.1	Phase-Portrait Analysis . . . . .	61
4.2.4	Sensor Configuration . . . . .	62
4.3	Decoupled Proportional Control . . . . .	63
4.3.1	Stability and Robustness . . . . .	64
4.4	Model Reference Algebraic Controller . . . . .	67
4.4.1	Algebraic Controller without Model Reference . . . . .	67
4.4.1.1	Stability and Robustness . . . . .	69
4.4.2	Algebraic Controller with Model Reference . . . . .	72
4.4.2.1	Stability and Robustness . . . . .	74
4.5	Gradient-Based Controller . . . . .	76
4.5.0.1	Stability and Robustness . . . . .	78
4.6	Further Discussions . . . . .	82
4.7	Conclusion . . . . .	87
<b>5</b>	<b>CONCLUDING REMARKS . . . . .</b>	<b>89</b>
5.1	Conclusion . . . . .	89
5.2	Summary of Dissertation Contributions . . . . .	90
5.3	Future Work . . . . .	90
	<b>BIBLIOGRAPHY . . . . .</b>	<b>92</b>

## LIST OF FIGURES

1.1	General schematic of the proposed self optimizing control framework. . . . .	3
2.1	Schematic representation of the physical (steam) biochar activation experimental setup. . . . .	7
2.2	Furnace and reactor temperature during biochar placement and activation process without ILC. . . . .	8
2.3	Furnace and reactor temperature profiles for an experiment consisting of several step changes at different temperature ranges. . . . .	9
2.4	The Matlab GUI: visualizes recorded data, visualizes the utilized reference for the recipe run. . . . .	12
2.5	Comparison of reactor temperature for constant reference temperature and after two iterations of learning. . . . .	13
2.6	Furnace reference temperature evolution through iterations of learning. . . . .	13
3.1	RMPC Simulink scheme. The plant could be either SISO or MIMO. . . . .	19
3.2	Schematic picture of one-compression-stage, one-load-demand vapour-compression refrigeration cycle [46]. . . . .	20
3.3	The original PID2018 Benchmark Challenge Simulink Model. . . . .	21
3.4	Manipulated variables comparison for multivariable PID control system versus RMPC . . . . .	23
3.5	Controlled variables comparison for multivariable PID control system versus RMPC . . . . .	23

3.6	Comparison between the PID18 benchmark challenge control strategies [47]. . . . .	24
3.7	Step response comparison of models obtained using Oustaloup method with different orders. . . . .	27
3.8	Error between step response of the fractional order system and approximated models using Oustaloup method. . . . .	28
3.9	Frequency response comparison of models obtained using Oustaloup method with different orders. . . . .	29
3.10	Step response comparison of models obtained using time-response-invariant approximation method with different orders. . . . .	30
3.11	Error between step response of the fractional order system and approximated models using time-response-invariant approximation method. . . . .	31
3.12	Frequency response comparison of models obtained using time-response-invariant approximation method with different orders. . . . .	32
3.13	Closed loop response of the fractional order system in example 1 under RMPC. . . . .	34
3.14	Control effort of RMPC controller for example 1. . . . .	35
3.15	Step response comparison of models obtained using time-response-invariant approximation and Oustaloup method with different orders. . . . .	36
3.16	Error between step response of the fractional order system and approximated models using time-response-invariant approximation method and Oustaloup method. . . . .	37
3.17	Frequency comparison of models obtained using time-response-invariant approximation method and Oustaloup method with different orders. . . . .	38
3.18	Closed loop response of the fractional order system in example 2 under RMPC. . . . .	39

3.19	Control effort of RMPC controller for example 2. the top curve shows the input for first output, $y_1$ , following the set-point $r_1 = 10$ , and the bottom curves shows the input for the second output, $y_2$ , following the set-point $r_2 = 2$ . . . . .	40
3.20	MESA Lab Peltier temperature platform . . . . .	43
3.21	Identified model for Peltier system and fitting . . . . .	44
3.22	RMPC with preview controller . . . . .	46
3.23	Plant output under RMPC with Preview for different preview horizons for set-point $T = 60^\circ\text{C}$ . . . . .	47
3.24	Control action for plant under RMPC with Preview for different preview horizons for set-point $T = 60^\circ\text{C}$ . . . . .	47
3.25	Plant output under RMPC with Preview for different preview horizons for set-point $T = 45^\circ\text{C}$ . . . . .	48
3.26	Plant output under RMPC with Preview for different preview horizons for set-point $T = 10^\circ\text{C}$ . . . . .	49
3.27	Plant output under RMPC for different prediction horizons. . . . .	50
3.28	Plant output under RMPC for different control penalization weight, $W_u$ . . . . .	51
3.29	Control action of RMPC for different control penalization weight, $W_u$ . . . . .	52
3.30	Plant output under RMPC for different observer gains. . . . .	52
3.31	Control action of RMPC for different observer gains. . . . .	53
4.1	Circuit diagram of high-power RF plasma drive with an L-type matching network considered in this work. . . . .	57
4.2	Reflection percentage ( $100\ \Gamma\ ^2$ ) over Load and Tune capacitance space. . . . .	59

4.3	Single-sensor impedance matching control configuration for conventional control approaches. . . . .	62
4.4	Double-sensor impedance matching control configuration for the proposed SOC approaches. . . . .	63
4.5	Schematic of conventional decoupled proportional control system. .	63
4.6	Reflection coefficient based phase-portrait plot for decoupled proportional control for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.	65
4.7	Capacitor position based phase-portrait plot for decoupled proportional control for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.	66
4.8	Flowchart of voltage command generation for the lower level motion controller by the high level matching controller, after desired/commanded capacitor positions are calculated. $\delta P_{th}$ is the maximum position change possible in a controller loop time. As long as at least one of the capacitors is further from the commanded position than this threshold ( $\delta P_{th}$ ), then capacitors are moved with maximum speed while maintaining the direction in the capacitor position space; after that system switches into fine tuning mode with the gain $K_{fine}$ , to avoid overshoot or potential oscillation. . .	68

4.9	Reflection coefficient based phase-portrait plot for the Algebraic Control for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.	69
4.10	Capacitor position based phase-portrait plot for the Algebraic Control for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.	70
4.11	Reflection coefficient based phase-portrait plot for Algebraic Control for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.	71
4.12	Capacitor position based phase-portrait plot for the Algebraic Control for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.	72
4.13	Block diagram of RF impedance matching with the Model Reference Algebraic Controller.	73
4.14	In-detail schematic of reactance trajectory generation flow in the Model Reference Algebraic Controller.	74

- 4.15 Reflection coefficient based phase-portrait plot for the Algebraic Control with model reference for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$  with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories. . . . . 75
- 4.16 Capacitor position based phase-portrait plot for the Algebraic Control with model reference for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$  with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories. . . . . 76
- 4.17 Detailed schematic of the Gradient-Based Controller.  $RP_{th}$  is the reflection percentage threshold, and  $K_{CL}$  and  $K_{CT}$  are the gains deciding the gradient descent step size when reflection percentage drops below the threshold. . . . . 77
- 4.18 Reflection coefficient based phase-portrait plot for Gradient-Based Controller for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories. 79
- 4.19 Capacitor position based phase-portrait plot for Gradient-Based Controller for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories. 80

4.20	Reflection coefficient based phase-portrait plot for Gradient-Based Controller for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories. . . . .	81
4.21	Capacitor position based phase-portrait plot for Gradient-Based Controller for the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories. . . . .	82
4.22	Time trace plot for the decoupled proportional controller at the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ , with capacitor position initial condition of $P_{C_L} = 60 \%$ and $P_{C_T} = 15 \%$ . The top plot shows the simulated reflected and forward power. The middle plot shows the Load and Tune capacitor positions. The bottom plot shows the control action which is the voltage input to the low-level motion controllers for Load and Tune capacitors. . . . .	84
4.23	Time trace plot for the Model Reference Algebraic Controller at the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ , with capacitor position initial condition of $P_{C_L} = 60 \%$ and $P_{C_T} = 15 \%$ . The top plot shows the simulated reflected and forward power. The middle plot shows the Load and Tune capacitor positions. The bottom plot shows the control action which is the voltage input to the low-level motion controllers for Load and Tune capacitors. . . . .	85
4.24	Time trace plot for the Gradient-Based controller at the constant load $R_L = 5 \Omega$ and $X_L = 50 \Omega$ , with capacitor position initial condition of $P_{C_L} = 60 \%$ and $P_{C_T} = 15 \%$ . The top plot shows the simulated reflected and forward power. The middle plot shows the Load and Tune capacitor positions. The bottom plot shows the control action which is the voltage input to the low-level motion controllers for Load and Tune capacitors. . . . .	86

## LIST OF TABLES

2.1	Estimated 1st order plus delay model parameters for different temperature ranges. . . . .	10
3.1	Initial operating values for the manipulated and controlled variables	21
3.2	Comparison of computational time. . . . .	33
3.3	RMPC with preview, parameters . . . . .	45

## ABSTRACT

The increased attention towards sustainability, and the rapid advancement in technology in recent years, has introduced an unprecedented need towards more optimal process control. On the other hand, the increase in computing power per chip following the Moore's law, and the decrease in computing power cost following the price-performance Moore's law, enables implementation of more computationally demanding control algorithms that their application was deemed impractical in the recent past. This work presents a framework called self optimizing control (SOC) that attempts to address the new requirements for process control by pursuing the attainment and sustainability of optimal performance. Then, several control strategies that enable the realization of the SOC framework are introduced and implemented to solve control problems associated to thermal processing and radio frequency (RF) impedance matching. An iterative learning control (ILC) approach is used to control an experimental carbon activation plant. A plug and play model predictive control (MPC) toolbox is developed that allows for straightforward implementation of embedded MPC for process control; this toolbox is then used to solve control problems for several experimental and simulation-based examples associated to thermal processes. Finally, a double sensor configuration is proposed to solve RF impedance matching in the context of semiconductor manufacturing as opposed to the conventional single sensor configuration to achieve robust optimal matching performance. A model-reference adaptive control along with a gradient based reflection minimization approach are developed to control an L-type matching network based on the double sensor configuration.

# Chapter 1

## INTRODUCTION

### 1.1 Overview

With sustainability becoming a pervasive trend [1], and the rapid advancement in technology in recent years, advanced and optimal process control is more than ever pursued. While sustainability is closely tied to process control design through attainment of optimal energy management [2], the increased process complexity for development of new technology calls for smart, precise, and optimal process control design [3]. This dissertation showcases implementation of some advanced control techniques for problems in thermal processes and radio frequency (RF) impedance matching to address the mentioned new requirements.

The term “thermal process” was used by Ahrendt and Taplin [1] to describe processes in which one of the principal objectives was the addition or subtraction of heat; the control systems for such processes were called “thermal controllers”. While a variety of control strategies for control of different thermal processes has been studied in the literature; industry relies mainly on on-off control methods [4] and proportional integral derivative (PID) controllers due to their simplicity [5]. However, with the current advances in technology, new applications are emerging where precision, uniformity, and optimality in terms of time and energy are becoming more and more important in control of thermal processes.

One important application of impedance matching is in chip manufacturing processes in the semiconductor industry where so many factors/process conditions such as mass flow rate of difference gas species, the substrate temperature, plasma power can affect the electrical impedance of the system [6]. Therefore, an active impedance matching is necessary for maximum power transfer to the system i.e minimum power reflection from the system [3]. As the Semiconductor industry is growing faster than ever by following Moor’s law [7] and moving towards feature sizes as small as 10nm on electronic chips, the demand for shorter plasma processing time becomes higher than ever [8]. Shorter process time means that plasma instabilities become a more significant portion of the process and as a result more precise RF matching control with shorter time-to-match becomes more and more necessary. Luckily, availability of computational power and sensing technologies at lower cost [9] allows us to take advantage of more complex control systems for impedance matching to meet those new requirements.

In this dissertation a self optimizing control (SOC) framework is proposed to address the need for advanced and optimal process control and some control strategies that enable the attainment of this framework are implemented for experimental and simulation-based examples in thermal processes and RF impedance matching. The rest of this chapter is focused on self optimizing control by exploring the history of application of the term in literature, defining the framework proposed in this dissertation and reviewing the control strategies that allow for realization of the proposed framework. Finally an outline explaining the general content of the upcoming chapters is provided.

## **1.2 Self Optimizing Control - A General Framework**

### **1.2.1 History**

The term self optimizing control has been used in control literature for several different concepts. In older literature, self optimizing control is used as an alternative for adaptive control [10]. In 2000, Skogestad introduced a new concept for SOC [11] in which the original optimal control problem is transformed to a feedback control problem by optimally choosing controlled variables of the control system such that an acceptable operation is obtained with constant set-points for those controlled variables. However, SOC in this sense is mostly studied for chemical plants where selection of controlled variables are more relevant. In the field related to heating and cooling systems, SOC has been used to refer to optimization based control algorithms that do not require a model such as extremum seeking control (ESC) [12, 13]. In 1995, Li [14] introduced an SOC framework for intelligent machines, particularly for an exercise machine, where SOC allows for online adjustment of what optimal behavior is for the machine and also adaptively controls the plant to realize that optimal behavior.

### **1.2.2 The Proposed Framework**

The self optimizing control framework proposed in this work is based on seeking accuracy and optimality, online self-adjustment, and finally learning from the operating history of the system considering that thermal processing often involves repetitive operations. We consider any control strategy that retains the following characteristics an SOC:

- 1- A feedback or feedforward control law that satisfies a certain optimal criterion.
- 2- Online adjustment of parameters to pursue optimality in presence of disturbances or variations in system model.
- 3- Learn from history of system operation to update controller parameters and the criterion.

Figure 1.1 shows a general schematic for the introduced SOC framework. To clarify the difference between the last two items, consider impedance matching problem for RF plasma in semiconductor industry. While the adaptive controller is responsible for optimal control of the matching network considering the varying plasma load, the data collection and processing unit allows for learning optimal initial conditions to initialize the match network (before feedback controller start playing a role) for next wafer processing for a specific recipe based on the previous runs of wafer processing for that recipe. The importance of the third item is in the fact that technology nowadays has allowed for abundance of measured data for different processes.

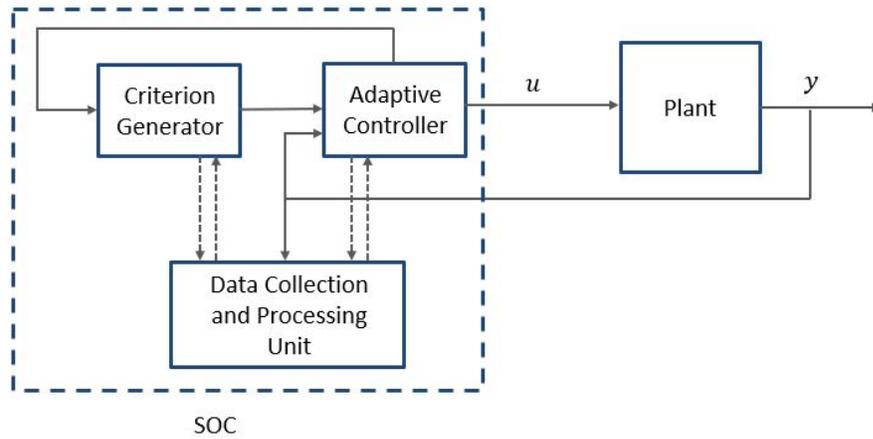


Figure 1.1: General schematic of the proposed self optimizing control framework.

### 1.2.3 Strategies for Realization

Some control strategies that should be considered for realizing the SOC framework can be listed as model predictive control (MPC), iterative learning control (ILC), model reference adaptive control (MRAC), extremum seeking control (ESC), repetitive control (RC), neural networks, etc.

This dissertation has focused more on MPC, ILC, and MRAC based on the problems in hand. Model predictive control (MPC) is a very good candidate for the feedback control method to implement SOC for thermal processes considering that the dynamics for these processes are not too fast. Model Predictive Control (MPC) is an advanced control strategy that is one of the most widely implemented in industry [15, 16]. Originally developed for petroleum refineries [17, 18], MPC now has a wide variety of applications in different fields such as chemical processes in pharmaceutical applications and power plants [19–22]. The reason for this success attributes to its inherent ability to handle multivariable control problems,

the ability to handle constraints, delay, non-minimum phase issue, and open-loop unstable systems [23, 24].

Iterative learning control (ILC) is based on the notion that the performance of a system that executes the same task multiple times can be improved by learning from previous executions. ILC is one of the most effective strategies to learn from operating history of a system when the system involves repetitive operations [25]. This makes ILC a very good candidate for realizing the third characteristic of proposed SOC. ILC has been used in wide variety of applications including rapid thermal processing [26, 27]. ILC has also found application to systems that do not have identical repetition. For instance, in [28] an underwater robot uses similar motions in all of its tasks but with different task-dependent speeds.

Model reference adaptive control utilizes the model of the system to dynamically adjust control system variables by comparing the output of the plant with response of the model to the same inputs [29]. MRAC provides observation on the amount of model mismatch and allows for an adjustment mechanism that enables controllers to deal with nonlinear actuators, model uncertainties, variations in character of disturbances, etc. [30]. MRAC has been used for different applications in the industry from aircraft manufacturing to biomedical field [31].

### 1.3 Dissertation Outline

The rest of this dissertation is structured as follows. In Chapter 2, an algorithm based on ILC technique is developed to achieve uniform temperature over time for activation process of biochar in a physical (steam) carbon activation plant. Experiments based on the algorithm are performed and results are shown. Moreover, a Matlab toolbox with graphical user interface (GUI) is developed to help with visualization of recorded data and execution of the algorithm.

In Chapter 3, a plug and play model predictive control toolbox called RMPC is developed that allows for straightforward implementation of MPC for either SISO or MIMO systems; the operator only needs to provide an identified state-space model for the system. Then effectiveness of developed method is demonstrated on PID 2018 benchmark problem. Moreover, RMPC toolbox is equipped with time-response-invariant approximation method to handle control of MIMO multi-fractional order systems. Finally, considering that in many process control applications, the future reference information is known, the RMPC package is modified to incorporate reference preview information into the control scheme. RMPC with preview is applied to control a hardware-in-the-loop (HIL) thermal system.

In Chapter 4, self optimizing control of an L-type impedance matching network motivated by application in semiconductor manufacturing industry is explored.

A double sensor configuration is proposed to solve RF impedance matching as opposed to the conventional single sensor configuration to achieve robust optimal matching performance. A model reference adaptive control along with a gradient-based reflection minimization approach are developed, their stability and performance are studied, and their performance is compared to a conventional decouples PID control approach.

Finally, in Chapter 5, contributions made in this dissertation are summarized, and opportunities for future efforts are suggested.

## Chapter 2

# ITERATIVE LEARNING CONTROL FOR A LAB-SCALE PHYSICAL (STEAM) CARBON ACTIVATION PLANT

### 2.1 Introduction

Physical carbon activation is a process through which coal, biomass, or biochar, is treated with steam or carbon dioxide in a reactor at temperatures in the neighborhood of  $800^{\circ}\text{C}$  in order to increase the overall porosity, especially micropores which range from Angstroms to nanometers in size, of the material and turn it into what is called activated carbon.

This project, focuses on temperature control for a lab-scale physical activation plant [32] to achieve the desired reactor temperature and maintain it during the activation process. First the system behaviour is studied and a model for reactor temperature is obtained. Then a control strategy based on iterative learning of reference furnace temperature is developed to achieve the required reactor temperature behaviour.

### 2.2 Problem Description

Figure 2.1 shows the schematic of the plant where the reactor is placed inside a thick metal tube, inside a tube furnace whose temperature is controlled with a commercial PID controller. It is important to notice the difference between furnace temperature and reactor temperature since the furnace temperature is controlled by the PID while there is a substantial dynamics involved in how reactor temperature follows the furnace temperature as explained later. The reactor temperature is measured using a thermocouple attached to the end-cap of the reactor, while, the thermocouple for furnace temperature is placed inside the furnace and outside of the metal tube.

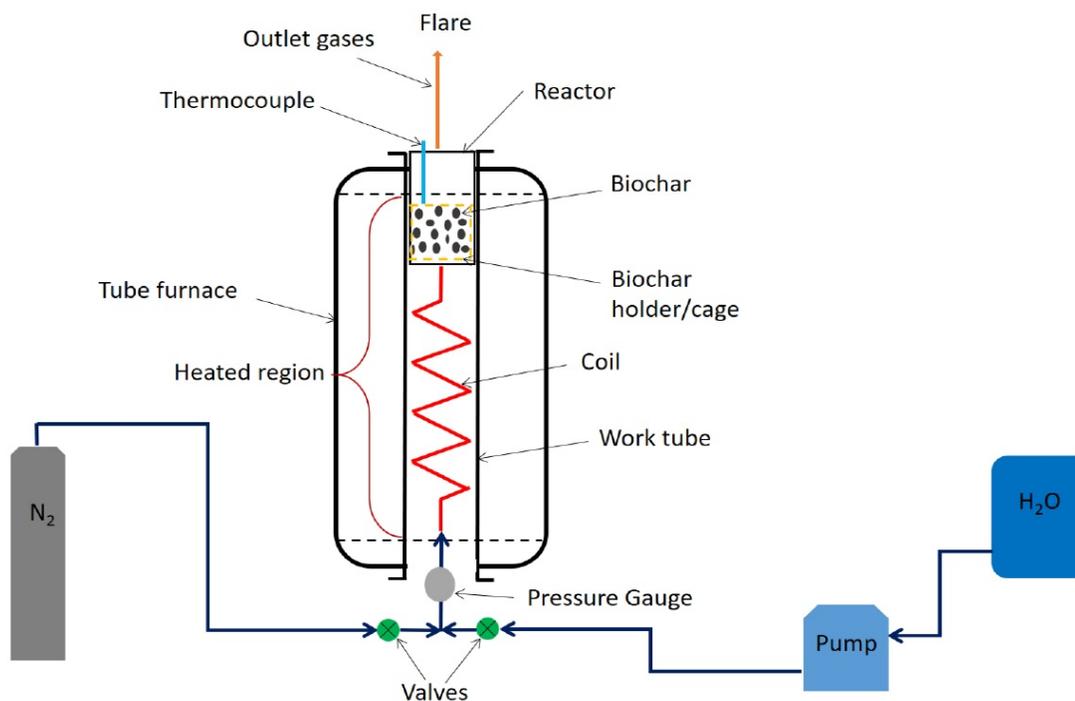


Figure 2.1: Schematic representation of the physical (steam) biochar activation experimental setup.

In order to mimic the large scale physical activation procedures, biochar sample with ambient temperature needs to be placed in the heated ( $800^{\circ}\text{C}$ ) reactor to experience a very high rate of heating. Thus, a testing procedure based on furnace temperature (without particular control over reactor temperature) is as follows: The furnace is preset and heats up to the activation temperature. Once the furnace reaches the activation temperature, the water pump is turned on at the desired flow rate to initiate the production of superheated steam. While the furnace temperature closely follows the preset reference and reaches steady state within seconds, enough time should be given to reactor for its temperature to reach steady-state as well (around 30 minutes). The lag in reactor temperature happens due to the large thermal inertia of the metal tube and reactor, moreover, the heat loss to the steam flow through the system and ambient results in an offset between reactor and furnace temperature. At this time, the water pump is turned off, the reactor end-cap is taken off, the biochar sample is quickly placed inside the reactor. Then the reactor is closed and the pump is switched back on. All these steps are performed in approximately 30 seconds. Since the thermocouple is attached to the end-cap, which is removed temporarily while inserting the biochar into the reactor, the temperature reading falls well below the actual reactor temperature. As soon as

the thermocouple is re-inserted, the temperature reading goes back up to the reactor temperature pretty fast. However, because of the biochar placement procedure, the reactor experiences a temperature drop (ignoring the fake drop in thermocouple reading) anyway. Figure 2.2 shows the reactor and furnace temperature profiles during the biochar placement and activation process when ILC is not used to adjust the reactor temperature. It can be observed from this figure that reactor temperature is constantly increasing during the activation process and is not constant.

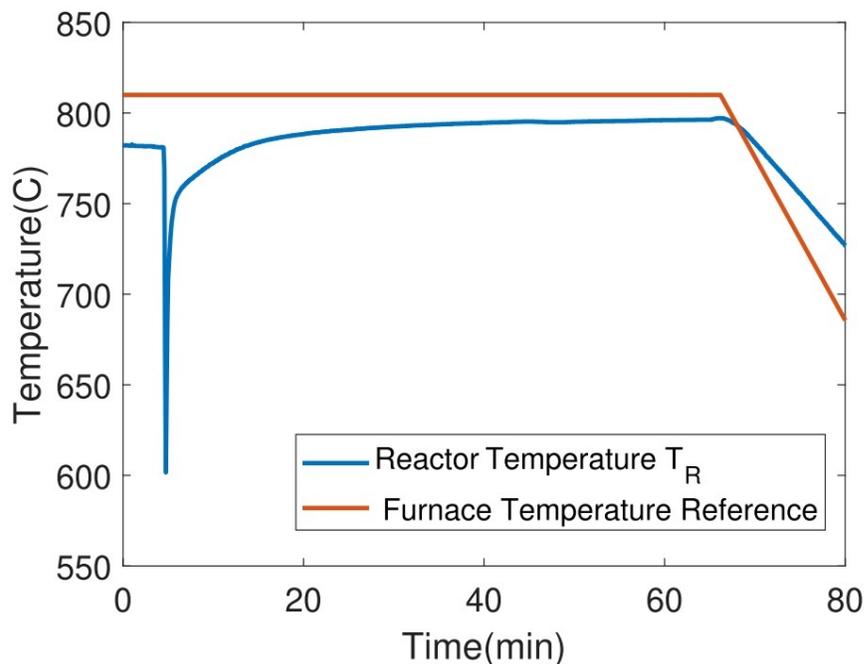


Figure 2.2: Furnace and reactor temperature during biochar placement and activation process without ILC.

Upon completion of the activation process, the water flow is stopped. At this time nitrogen flow is initiated to avoid any further chemical reactions and to help the cool-down process. The sample is taken out of the reactor at low temperatures to avoid instantaneous combustion of the biochar.

The objective of this work is to obtain a constant temperature during activation process.

### 2.3 Modeling and Simulation

Considering the large temperature operation range of the system (ambient to more than 800°C), nonlinear behaviour is quite expected. So, an experiment consisting of a series of step changes in different temperature ranges is designed to

study the system behaviour and non-linearity as shown in Fig. 2.3. Then considering a 1st order plus delay model:

$$G(s) = \frac{K}{\tau s + 1} e^{-Ls}, \quad (2.1)$$

the parameters, i.e., gain  $K$ , time constant  $\tau$ , and delay  $L$  are estimated for each step change. Table 2.1 shows the estimated parameters for different temperature ranges. According to the table, as the temperature increases, the dynamics of the system becomes faster and the delay decreases significantly. Moreover, the reduction in gain means that in higher temperatures the offset between reactor temperature and furnace temperature increases.

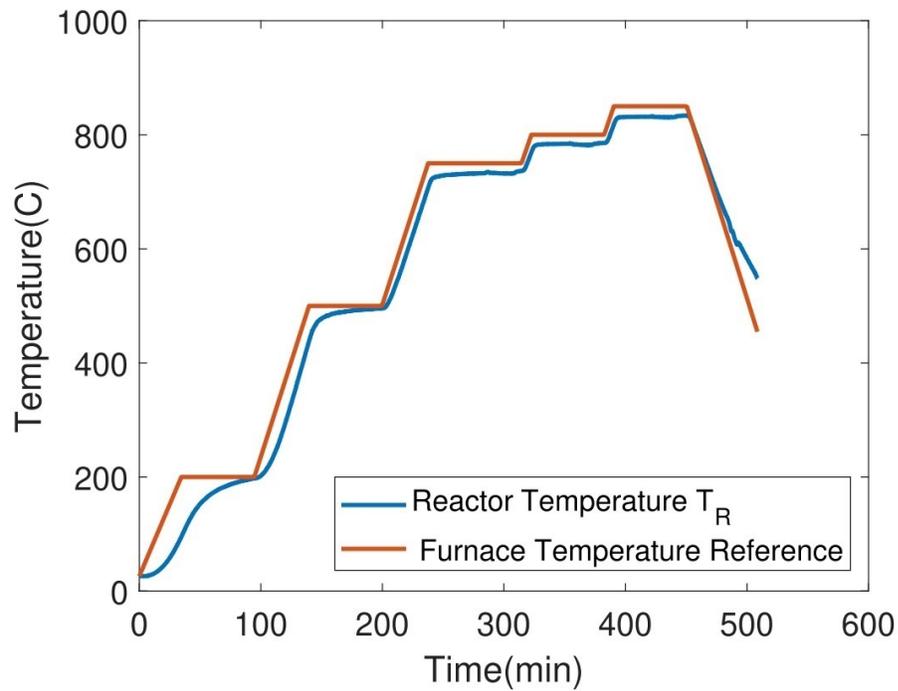


Figure 2.3: Furnace and reactor temperature profiles for an experiment consisting of several step changes at different temperature ranges.

Table 2.1: Estimated 1st order plus delay model parameters for different temperature ranges.

Temperature Range (°C)	K	L (seconds)	$\tau$ (seconds)
0-200	0.9981	395	1092
200-500	0.9845	239	312
500-750	0.9452	78	201
750-800	0.9383	55	105
800-850	0.9212	50	89

## 2.4 Proposed Solution

### 2.4.1 Iterative Learning of Reference

probably the first idea that comes to mind to control the reactor temperature during the activation process would be to design a feedback control with feedback from reactor temperature instead of furnace temperature. However, aside from the fact that the furnace in this case is a closed commercial product and to do so requires extra resources, more importantly, considering the large time constant and delay of the system even in high temperature (Table 2.1), with a feedback control it would still take a few minutes for the reactor temperature to reach steady state after the disturbance due to biochar placement into the reactor.

The solution proposed here is to achieve a reactor temperature before biochar placement in a way that the temperature drop due to biochar placement would bring the temperature right to the desired activation temperature and to keep the reactor temperature at that temperature for the rest of activation process. This can be achieved by the knowledge acquired from system identification and utilizing of iterative learning control technique to learn a profile for furnace temperature so that reactor temperature would stay steady during the activation process. Moreover, this solution does not require any extra resources in terms of hardware development. However, the limitation of this method is in that for any particular recipe, a few experiment runs are required to obtain an optimal solution.

### 2.4.2 Control Algorithm

Considering a particular recipe, i.e., the biochar sample used, the desired treatment temperature  $T_d$ , the treatment time, and the steam flow rate, the procedure steps to achieve a uniform and desired treatment temperature using the proposed solution is as follows:

- Use the simulation model to estimate the furnace temperature  $T_{F_d}$ , that results in desired steady-state reactor temperature  $T_{R_d}$ .

- Run a test during which furnace temperature reaches the steady state  $T_{Fd}$  and is kept constant during biochar placement and activation and record reactor and furnace temperature.
- Obtain the error signal:

$$e^n(t) = T_{Rd} - T_R^n(t), \quad (2.2)$$

where  $T_R^n(t)$  is the recorded reactor temperature after biochar placement (ignoring the fake drop due to exposure of the thermocouple to ambient temperature) for the  $n$ th iteration.

- Update the reference for furnace temperature for the next experiment as follows:

$$T_F^{n+1}(t) = T_F^n(t) + K_{ILC}e^n(t), \quad (2.3)$$

where  $T_F^n(t)$  and  $T_F^{n+1}(t)$  are the furnace temperature profile for previous and next run during biochar placement process preparation and activation process respectively.  $K_{ILC}$  is the learning gain.

- Run a test during which the furnace temperature reaches the first value in the time series  $T_F^{n+1}(t)$  and stays constant at that value for enough time before biochar placement (the time required for reactor to reach steady-state). The furnace temperature starts following the learned profile,  $T_F^{n+1}(t)$ ,  $t_a$  seconds in advance of biochar placement process. The advance time is obtained by subtracting the time required for biochar placement,  $t_{bp}$ , from the delay estimated at the specified temperature range,  $L$  (Table 2.1):

$$t_a = L - t_{bp}. \quad (2.4)$$

Please note that since the furnace only accepts linear reference input for temperature, the learned profile needs to be estimated into several linear sections.

- Repeat steps 2 to 5 to achieve a uniform reactor temperature during activation process.

### 2.4.3 A Control Toolbox

A preliminary Matlab toolbox is developed to allow for visualization and analysis of recorded data, and to generate reference trajectory based on the explained algorithm for the operator to enter into furnace controller. Figure 2.4 shows one of the sub-applications of the toolbox as an example.

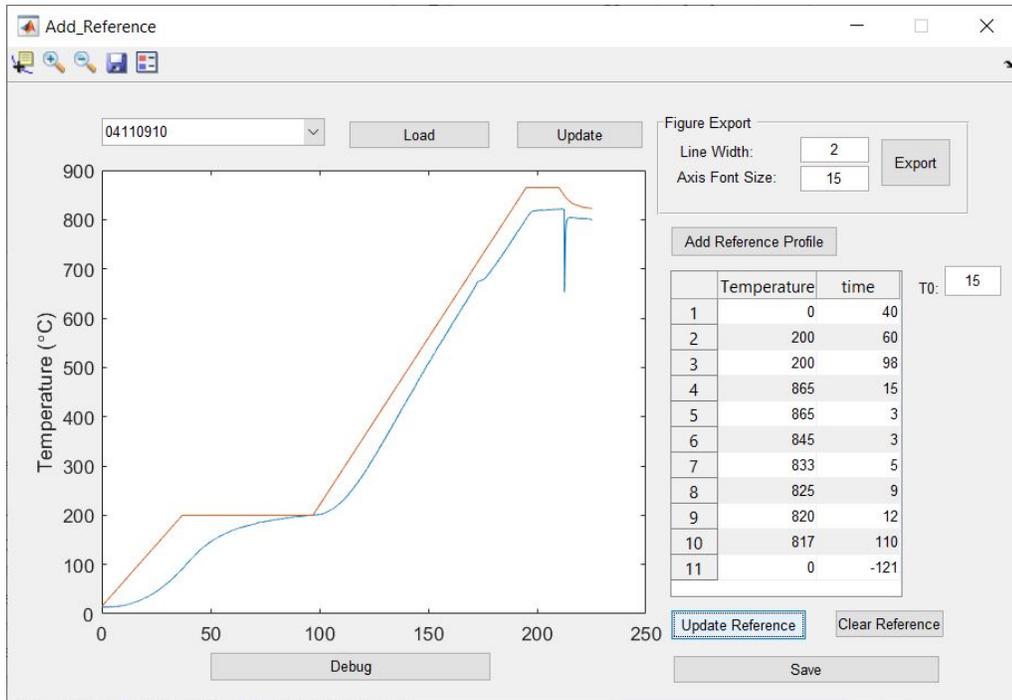


Figure 2.4: The Matlab GUI: visualizes recorded data, visualizes the utilized reference for the recipe run.

## 2.5 Results and Discussions

Figure 2.6 shows evolution of furnace temperature reference through iterations of introduced algorithm for an example recipe and Fig. 2.5 shows the resulted reactor temperature. It is observed from Fig. 2.5 that without learning, reactor temperature goes through around 50°C of change during the treatment while only after two iterations of learning a fairly uniform temperature profile is achieved.

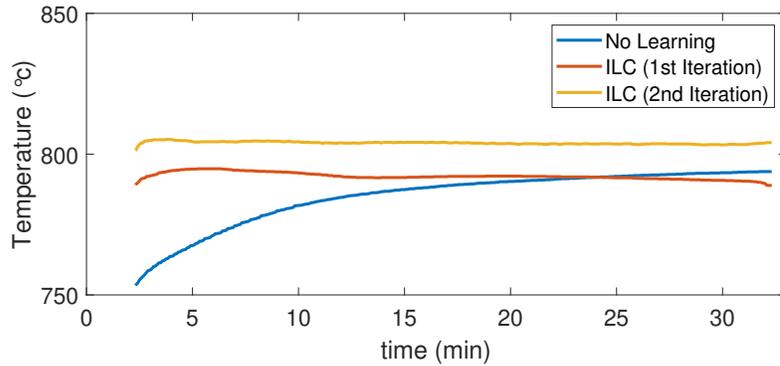


Figure 2.5: Comparison of reactor temperature for constant reference temperature and after two iterations of learning.

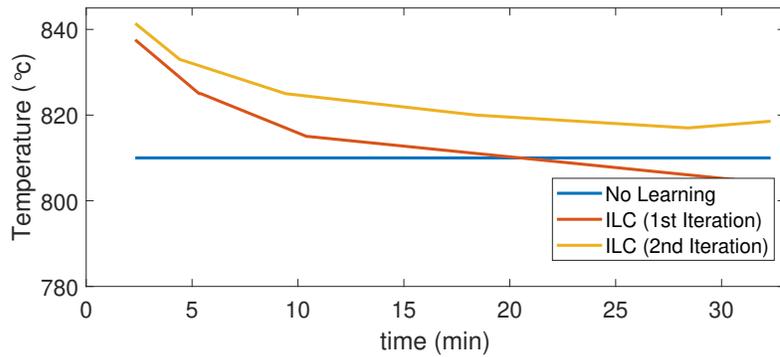


Figure 2.6: Furnace reference temperature evolution through iterations of learning.

## 2.6 Conclusion

In this project, an iterative learning control (ILC) approach was used to provide a solution to the problem of controlling the reactor temperature in a particular physical carbon activation plant. The objective was to achieve uniform reactor temperature during the carbon processing time, without modifying the internal commercial temperature controller that controls the furnace temperature. The proposed ILC approach is based on feeding a desired reference temperature profile to the internal furnace temperature controller and it successfully achieves a uniform reactor temperature during the carbon processing time in fewer than 4 iterations. A Matlab tool with GUI was also designed to help with analyzing the data from previous runs and applying the ILC rule.

## Chapter 3

# PLUG AND PLAY PLANT-READY MPC TOOLBOX, RMPC

### 3.1 Introduction

In this chapter an MPC control toolbox, based on Recursive Integration Optimal Trajectory Solver (RIOTS) is introduced. RIOTS is a Matlab toolbox for solving large class of linear and nonlinear optimal control problems subjected to trajectory and end-point constraints on control inputs as well as states [33, 34]. Tricaud and Chen proposed [35] using RIOTS as optimal control problem solver for MPC and compared its performance to control a DC motor model with MPC Toolbox of Matlab. Zhao, Li, and Chen [36] used MPC based on RIOTS to control SISO and MIMO fractional order models utilizing Oustaloup's approximation method to approximate the fractional order model with an integer order model. Moreover, Li [37] in his PhD dissertation used MPC based on RIOTS to control a two-by-two thermal system.

The contribution of this work can be summarized in:

- Development of MPC based on RIOTS into a toolbox called RMPC allowing for straight forward application of this method for SISO and MIMO systems; the user only needs to provide a model for the system and tune the knobs.
- Using RMPC to obtain the best reported results for PID benchmark 2018 challenge [38].
- Proposing a scheme for control of general MIMO multi-fractional-order systems using RMPC based on time-response-invariant approximation of fractional order system [39].
- Embedded application of RMPC with preview to control a SISO thermal HIL system [40].

The rest of this chapter is structured as follows. First a brief introduction to theory of MPC is given and necessary formulation is presented. RIOTS toolbox is introduced in the third section. Then general implementation of RMPC is shown in Section 3.4 and details of implementation for mentioned contributions and some of the results are given in fifth and sixth sections.

### 3.2 Model Predictive Control (MPC) Formulation

Since 1970's when MPC was established, different types of MPC have been proposed such as GPC (Generalized Predictive Control) [41], DMC (Dynamic Matrix Control) [17, 42], EHAC (Extended Horizon Adaptive Control) [43, 44], etc. All MPC schemes are based on three similar components: a predictive part based on a plant model, an optimization part based on a defined cost function, and a receding horizon part which updates the control input at every step. State-space model is commonly used in MPC systems because it can be easily applied to multivariable cases. Equation (3.1) shows the state-space plant model considered here.

$$\begin{cases} x(i+1) = Ax(i) + Bu(i) + w(i) \\ y(i) = Cx(i) + Du(i) + v(i) \end{cases}, \quad (3.1)$$

where  $x(i) \in R^{n_x}$  is the state,  $u(i) \in R^{n_u}$  is the input, and  $y(i) \in R^{n_y}$  is the output.  $w(i)$  and  $v(i)$  are the state noise and measurement noise, respectively, which are both assumed to be Gaussian with zero mean. The plant model, the centerpiece of MPC, allows the predictions to be calculated based on information up to the current time  $i$ . The state estimate at  $k+1$  time-step in the future,  $\hat{x}(i+k+1|i)$ , can be calculated as follows.

$$\begin{cases} \hat{x}(i+k+1|i) = A\hat{x}(i+k|i) + Bu(i+k|i) \\ \hat{y}(i+k|i) = C\hat{x}(i+k|i) + Du(i+k|i) \end{cases}, \quad (3.2)$$

where  $\hat{y}(i+k|i)$  is estimate of system output at  $i+k$ . The following equation shows the simplest form of objective function, typically considered in model predictive control:

$$J = \sum_{k=1}^{N_p} [\hat{y}(i+k|i) - r(i+k)]^T W_y [\hat{y}(i+k|i) - r(i+k)], \quad (3.3)$$

where  $W_y \in R^{n_y \times n_y}$  is a positive definite weighting matrix and  $r(i+k)$  is the reference input at instant  $i+k$ .  $N_p$ , which is called the prediction horizon, is the number of samples, for which, the error between future reference and future output estimate is considered in the objective function. Index  $J$  in (3.3) can be applied to both SISO and MIMO systems. Considering the introduced objective function, MPC repetitively solves an optimal control problem described as:

$$\min_{(u(i+1), u(i+2), \dots, u(i+N_u))} J, \quad (3.4)$$

subject to the system dynamics described in Eqn. (3.2), and other constraints such as a saturation limit on control input:

$$u_{min} < u(i) < u_{max}. \quad (3.5)$$

$N_u$  in Eqn. (3.4) is the number of future control output samples considered to solve the optimal control problem, and it is called the control horizon. After solving the optimal control problem at each sample time, a series of future control output with  $N_u$  samples is obtained; however, only the very first control output is applied to the system and the rest will be discarded. The optimization is repeated to calculate a new set of control output series using the updated information and all the sequences are brought up to date using the receding horizon concept.

### 3.3 Recursive Integration Optimal Trajectory Solver (RIOTS)

Equation (3.6) shows a general form of optimal control problem (OCP) which can be solved using RIOTS (Recursive Integration Optimal Trajectory Solver) toolbox [33, 34] in Matlab.

$$\min_{(u, \xi) \in L_\infty^m \times R^n} f(u, \xi) = g_o(\xi, x(b)) + \int_a^b l_o(t, x, u) dt \quad (3.6)$$

subject to:

$$\begin{aligned} \dot{x} &= h(t, x, u), x(a) = \xi, t \in [a \ b] \\ u_{min}^j(t) &< u^j(t) < u_{max}^j(t) \\ \xi_{min}^j(t) &< \xi^j(t) < \xi_{max}^j(t) \\ l_{ti}^v(t, x(t), u(t)) &\leq 0, v \in Q_{ti} \\ g_{ei}^v(\xi, x(b)) &\leq 0, v \in Q_{ei} \\ g_{ee}^v(\xi, x(b)) &= 0, v \in Q_{ee} \end{aligned} \quad (3.7)$$

where  $x(t) \in R^{n_x}$ ,  $u(t) \in R^{n_u}$ ,  $g : R^{n_x} \times R^{n_x} \rightarrow R$ ,  $l : R \times R^{n_x} \times R^{n_u} \rightarrow R$ ,  $h : R \times R^{n_x} \times R^{n_u} \rightarrow R^{n_x}$ .

The subscripts  $o$ ,  $ti$ ,  $ei$  and  $ee$  on the functions  $g(\cdot, \cdot)$  and  $l(\cdot, \cdot, \cdot)$  respectively refer to, objective function, trajectory constraint, endpoint inequality constraint, and endpoint equality constraint. Based on the nature of the optimal control problem, RIOTS can be used for optimization over both the optimal control  $u$  and one or more optimal initial state  $\xi$ .

In order for a user to implement RIOTS toolbox, the information regarding the problem and the model should be provided by the user using the following functions:

#### 3.3.1 sys\_acti.m

This function is called before all other functions and is used to preliminary setup such as loading some required parameters.

### 3.3.2 `sys_init.m`

The information about control problem such as number of model states, number of constraints, etc. is provided in this function. Moreover, parameters that need to be passed from Matlab to be used in the rest of functions during the run-time are introduced here.

### 3.3.3 `sys_h.m`

The right hand side of the differential equations describing the system dynamics,  $h(t, x, u)$ , is provided in this function.

### 3.3.4 `sys_g.m`

The endpoint cost function  $g_o(\xi, x(b))$  and the endpoint inequality and equality constraints  $g_{ei}^v(\xi, x(b))$  and  $g_{ee}^v(\xi, x(b))$  are provided in this function.

### 3.3.5 `sys_l.m`

Computation of the integrands of objective cost functions  $l_o(t, x, u)$  and the values of state trajectory constraints  $l_{ii}^v(t, x(t), u(t))$  are done in this function.

### 3.3.6 `sys_Dh.m`, `sys_Dl.m`, `sys_Dg.m`

These functions provide the derivatives of the functions above with respect to the arguments  $x, u$ . This step can be skipped as RIOTS can also run without providing these derivatives. More information about the above functions and their syntax is available in [33].

After providing the problem description files, the user can use the RIOTS by calling the command

$$\begin{aligned} [\mathbf{u}, \mathbf{x}, \mathbf{f}] = \text{riots}([\mathbf{x0}, \{\text{fixed}, \{\mathbf{x0}_{\min}, \mathbf{x0}_{\max}\}\}], \\ \mathbf{u0}, \mathbf{t}, \mathbf{u}_{\min}, \mathbf{u}_{\max}, \text{params}, \\ [\text{miter}, \{\text{var}, \{\text{fd}\}\}], \text{ialg}, \\ \{[\text{eps}, \text{epsneq}, \text{objreq}, \text{bigbnd}]\} \\ \{\text{scaling}\}, \{\text{disp}\}, \{\text{lambdal}\}) \end{aligned} \quad (3.8)$$

where

- $\mathbf{x0}$ : initial value of the state  $x$ .
- $\mathbf{x0}_{\min}, \mathbf{x0}_{\max}$ : the lower limit and upper limit of initial state.
- $\mathbf{u0}$ : initial value of the control input  $u$ .
- $\mathbf{u}_{\min}, \mathbf{u}_{\max}$ : the lower limit and upper limit of control input.

- `params`: the parameters provided to problem description files.
- `miter`: the maximum number of iterations allowed.
- `var`: it stands for a penalty on the piecewise derivative variation of the control to be added to the objective function.
- `fd`: if a non-zero value is given, the gradients for all functions will be obtained by finite difference approximations without calling `sys_Dh.m`, `sys_Dl.m`, and `sys_Dg.m`.
- `ialg`: it specifies the integration algorithm.

By applying the MPC philosophy, the open-loop optimal control problem solver RIOTS can easily be converted into feedback control system with the receding horizon in MPC.

### 3.4 RMPC Toolbox

The current version of RMPC toolbox is designed to allow for straight forward implementation of MPC for SISO and MIMO systems, when a linear model of the form represented in Eqn. (3.1) is used as the internal model of the controller. The objective function for the optimization problem is:

$$J = (y(N_p) - r(N_p))^T W_g (y(N_p) - r(N_p)) + \int_0^{N_p} ((y(t) - r(t))^T W_l (y(t) - r(t)) + W_u u(t)^T u(t)) dt, \quad (3.9)$$

where  $r$ ,  $y$ , and  $u$  are set-point, output, and input, respectively,  $N_p$  is the prediction horizon,  $W_g$  is the weight for end-point cost,  $W_l$  is the weight for trajectory cost, and  $W_u$  is the weight for penalizing the input. Control horizon would always be equal to prediction horizon for the current version. The optimization problem is subjected to the following constraint on system input:

$$u_{min} < u(t) < u_{max}. \quad (3.10)$$

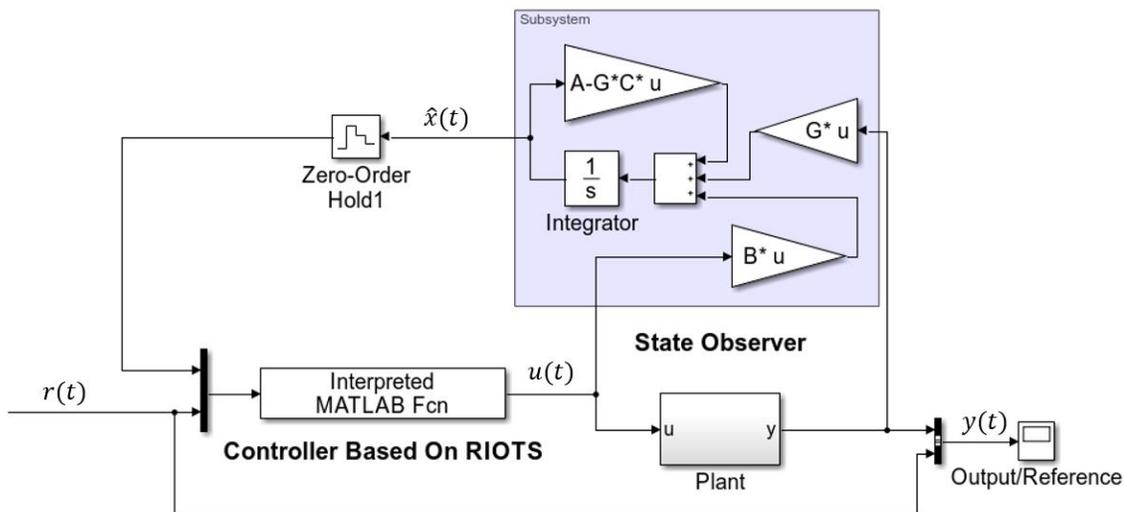


Figure 3.1: RMPC Simulink scheme. The plant could be either SISO or MIMO.

In RMPC, the full-state information of the system is required. Therefore, the states of the system must be either measured directly or estimated using an observer. Since in most cases, the states of the identified model for the plant cannot be directly measured, an observer is required to estimate the full state information of the system at each sample. In this work, the Luenberger [45] observer is employed to estimate the internal model states:

$$\dot{\hat{x}} = (A - GC)\hat{x} + Gy + Bu. \quad (3.11)$$

Here,  $A$ ,  $B$ , and  $C$  are internal model system matrices and the gain matrix,  $G$  is designed to place the eigenvalues of  $A - GC$  at  $k_o[-1, -2, \dots, -n]$ ,  $n$  is the number of states and  $k_o$  is a coefficient to adjust the observers dynamic speed based on the specific problem in hand.

Figure 3.1 shows the Simulink scheme for implementation of RMPC. The users of the toolbox need only to provide an identified state-space model for the plant and tune the parameters,  $N_p$ ,  $W_g$ ,  $W_l$ ,  $W_u$ , and  $k_o$ .

### 3.5 Examples of RMPC Implementation

In this section the details of application of RMPC for three problems are given; all three projects resulted in peer reviewed papers [38–40].



Table 3.1: Initial operating values for the manipulated and controlled variables

Variable	Value	Units
$A_v$	$\cong 48.79$	%
$N$	$\cong 36.45$	Hz
$T_{sec\_evap\_out}$	$\cong -22.15$	$^{\circ}\text{C}$
$TSH$	$\cong 14.65$	$^{\circ}\text{C}$

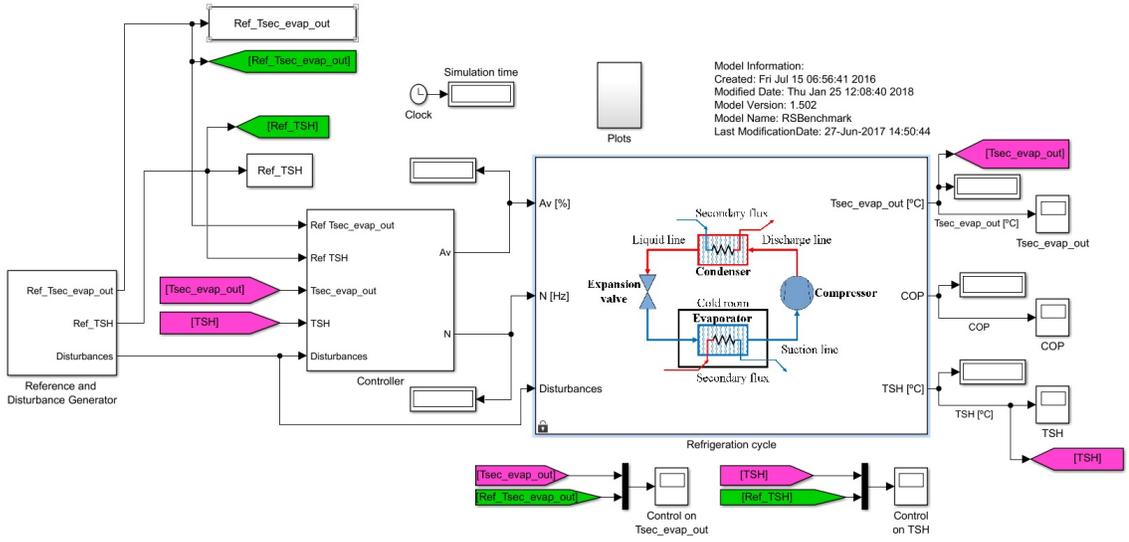


Figure 3.3: The original PID2018 Benchmark Challenge Simulink Model.

### 3.5.1.2 Implementation Details

System Identification toolbox of Matlab is used to obtain a state-space model of the system based on the system step response and a 4th order model is identified to obtain system matrices,  $A$ ,  $B$ ,  $C$ , and  $D$ .

The parameters of the RMPC as introduced in Section 3.4 are chosen as:

$$W_g = 0, W_l = \begin{bmatrix} 2.5 & 0 \\ 0 & 2 \end{bmatrix}, W_u = 0, N_p = 10, K_o = 1.$$

The reason behind setting the weight for end-point cost,  $W_g$ , and the weight for penalizing input,  $W_u$ , equal to zero is that at the time of dealing with this problem, the RMPC toolbox was only developed to handle the trajectory cost. As a result, to compensate for the steady state error, a conditional integrator is added to the control scheme. Introducing the vectors  $r_d$  to be the desired output

vector or the system reference,  $r_{MPC}$  to be the the reference given to the optimal control problem solver, and  $y$  to be the system measured output, then the integral compensator modifies the RMPC reference to compensate the steady state error with the following structure:

$$r_{MPC}(k+1) = r_d(k+1) + K_I \sum_i (r_d(i+1) - y(i)). \quad (3.12)$$

Here the index of summation  $i$  stands for any point in time up to the current time when the error  $r_d(i+1) - y(i)$  is less than some threshold  $e_{th}$ . The threshold for the error is considered to avoid addition of big errors during the transients and therefore not to make the system oscillatory while completely compensating for steady state error. Please note that since the system in the benchmark problem has two outputs,  $r_d$ ,  $r_{MPC}$ , and  $y$  are vectors of size two and  $K_I$ , the integral gain, is a diagonal two-by-two matrix. The integral gain matrix and the error summation threshold values are chosen to be:

$$K_I = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.25 \end{bmatrix}, e_{th} = \begin{bmatrix} 0.05 \\ 0.3 \end{bmatrix}.$$

The input constraint introduced in Eqn. (3.10) is imposed on the optimal control problem solver according to the problem description in Section 3.5.1.1.

### 3.5.1.3 Results and Discussion

The designers of PID 2018 benchmark challenge introduced a multivariable PID controller and required the challengers to compare their results with results of the mentioned controller.

Figures 3.4 and 3.5 show the comparison between control input and system output between the two controllers respectively. Moreover, the designers of the challenge have introduced eight performance indexes and an overall index,  $J$ , to allow for quantitative comparison between two control strategies. Viola, Radici, and Chen collected and listed the published indexes for comparison between different control strategies proposed by challengers and the multivariable PID designed by PID 2018 benchmark challenge designers [47]. A values less than one for different indexes means that the control strategy proposed by the challenger is performing better than the multivariable PID with respect to that index. Figure 3.6 shows the table published by Viola et al. and it can be seen that RMPC shows the best overall performance by obtaining the lowest value for overall index,  $J$ .

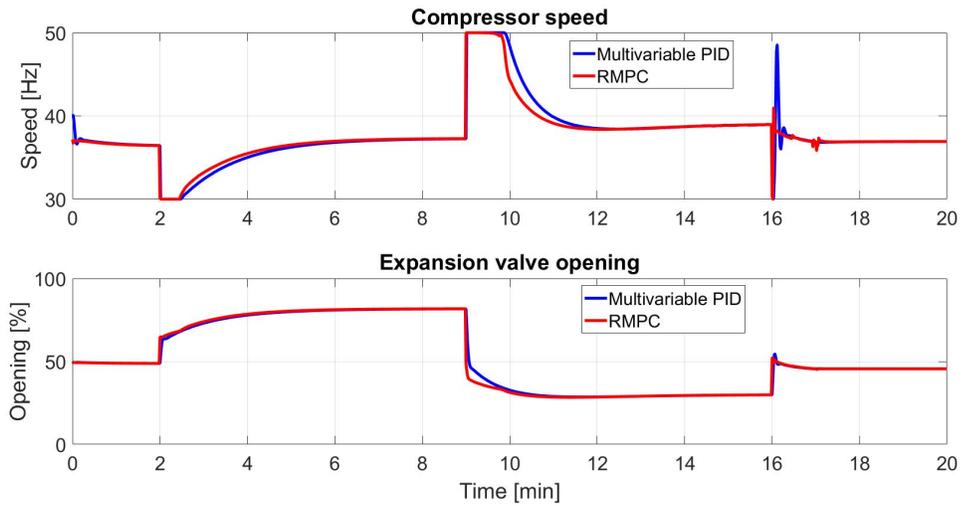


Figure 3.4: Manipulated variables comparison for multivariable PID control system versus RMPC

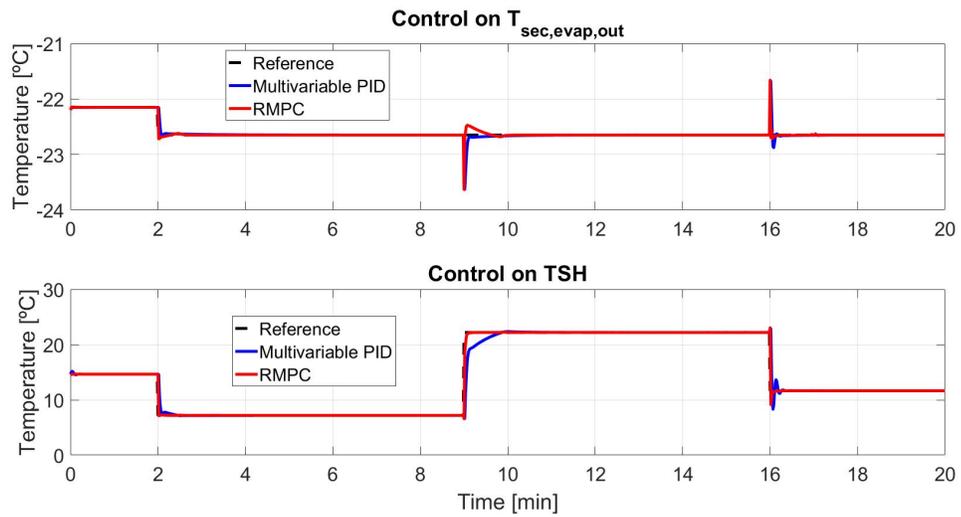


Figure 3.5: Controlled variables comparison for multivariable PID control system versus RMPC

TABLE VI: Comparison between the PID18 benchmark challenge control strategies

Pos	Control technique	First Author	University	Country	RIAE1	RIAE2	RITAE1	RITAE2	RITAE2	RITAE2	RIAVU1	RIAVU2	J
1	RIOTS MPC	Dehghan,S	UC Merced	US	0.2134	0.1047	0.1943	0.008	0.012	0.0241	1.1481	1.0938	0.2055
2	PID-IMC controller	Viola,J	UC Merced	US	0.071	0.20	0.003	0.016	0.1161	0.003	3.12	1.1	0.2065
3	MIMO robust PID	Tari,M	Bordeaux U	France	0.2819	0.2918	0.21776	0.09181	0.13766	0.065	1.0567	1.11143	0.2751
4	Data-driven error	Huff,D	UFRS	Brazil	0.1406	0.251	0.5444	0.0708	0.1901	0.03556	1.13	1.243	0.3225
5	Inverted decoupling	Garrido,J	Seville U	Spain	0.158	0.258	0.536	0.095	0.186	0.113	1.338	1.431	0.353
6	Feedforward control	Zhao,Y	UC Merced	US	0.4403	0.3297	0.3577	0.0812	0.111	0.1001	1.0738	1.5796	0.3744
7	Multiobjective Optimization	Reynoso-Mesa,G	UPC	Spain	0.2892	0.3569	0.6148	0.1705	0.2291	0.0967	1.147	1.1531	0.4028
8	MIMO virtual feedback	Bordignon,V	UFRS	Brazil	0.1569	0.3338	0.7193	0.3466	0.3704	0.0867	1.0499	0.9301	0.4134
9	Decentralized ADRC	Zhang,B	NCEPU	China	0.3951	0.5881	0.241	0.4858	0.5805	0.122	0.9751	0.7346	0.4182
10	Robust PID Autotuning	Zhao,S	NCEPU	China	0.3951	0.5881	0.241	0.4858	0.5805	0.122	0.9751	0.7346	0.4182
11	Model free adaptive control	Yu,X	BJU	China	0.2289	0.3516	1.0495	0.3267	0.6223	0.1593	1.0029	0.903	0.5393
12	Conditional Integrator	Yuan,J	UC Merced	US	0.406	0.4043	0.3682	0.5573	0.3682	0.2244	1.7494	1.7494	0.5662
13	IMC control	Cajo,R	Ghent U	Belgium	0.8716	0.8399	0.2954	0.6283	0.7651	0.3961	0.9024	0.7064	0.6332
14	Stochastic optimization	Ates,A	UC Merced	US	0.8728	0.8170	0.6820	0.3044	0.5517	0.1572	0.9755	0.9646	0.6532
15	Robust Decoupling	Rodriguez,D	Seville U	Spain	0.6046	0.6563	0.8932	0.5561	0.5043	0.4297	1.0012	0.7661	0.66
16	PID18 Benchmark	Bejarano,G	Seville U	Spain	0.3511	0.4458	1.6104	0.183	0.3196	0.128	1.1283	1.3739	0.68209
17	Robust FO controller	Muresan,C	Ghent U	Belgium	0.3609	0.5935	0.027	2.8161	1.6339	0.4625	0.927	0.6868	0.7837
18	nonlinear PID controller	Lei,Z	CEST	China	0.98	1.18	0.81	3.25	1.01	1.43	0.99	0.62	1.26
19	Evolutionary PID	Amador,G	UNAL	Colombia	0.752	0.6558	5.4358	0.4548	0.497	1.43	1.0658	1.1159	1.7472

Figure 3.6: Comparison between the PID18 benchmark challenge control strategies [47].

The improvement in results can be explained by looking at comparison of system outputs for RMPC versus multi-variable controller (Controller 2) in Fig. 3.5 where we observe better transient response. RMPC also shows significantly lower steady-state error.

It is worth mentioning that around time equal to 9 minutes, when a reference change happens for  $TSH$ , the compressor speed becomes saturated for both controllers (Fig. 3.4) and as a result some error on the outputs at this moment is unavoidable regardless of the controller utilized (Fig. 3.5). However, as it is implied from the quantitative comparison, the overall error on the two outputs are optimized for RMPC compared to multivariable PID controller.

## 3.5.2 RMPC for General Fractional Order Systems

### 3.5.2.1 Introduction

Since its development in 19th century by Riemann and Liouville, fractional calculus has grown its applications through different fields of science and engineering [48, 49]. Fractional order calculus, or fractional calculus in short, which can be considered as generalization of integer order calculus, allows for more accurate modeling of some complex physical systems, such as thermal systems, viscoelastic systems, electromagnetic waves, and many more [50–54]. Fractional calculus has also been widely used in control theory [55] since 1960 and one of the early contributions was the idea by Oustaloup [56]. CRONE [57],  $PI^\lambda D^\mu$  [58], and fractional order MPC [59] are some examples of fractional order controllers. As mentioned, some physical systems are fractional order in nature and it is important to advance the controller design arsenal of tools for such systems.

RIOTS toolbox is based on state-space model of the system and cannot handle OCP for fractional order systems directly. In order to apply RMPC for fractional order systems, the fractional order system first needs to be approximated with an integer order model. Tricaud and Chen [60] have shown the application of RIOTS toolbox to solve OCPs for fractional order systems by using Oustaloup approximation method. Zhao et al. [36] also used Oustaloup approximation to apply RMPC for fractional order systems. There are other examples of using an integer order MPC to control fractional order systems such as the paper [61] where Oustaloup’s method used for approximation and also the paper [62] where a direct method based on Grünwald–Letnikov’s definition is utilized for approximation.

The main contribution of this project is the implementation of integer order MPC based on RIOTS to control a general form of fractional order systems by approximating the fractional order system using time-response-invariant approximation method. The use of time-response-invariant approximation is motivated based on the observation that models approximated using Oustaloup’s method do not show accurate time-response unless for very high orders of approximation. Impulse-response-invariant discretization of fractional order filters are discussed in literature [63–65] and a Matlab tool is developed by Chen [66].

In the following two examples of implementation of RMPC for fractional order system are given. In the first example, comparison between Oustaloup approximation method and time-response-invariant approximation method in implementation of RMPC is mainly discussed. In the second example, the results of implementation of the proposed approach to control a MIMO commensurate fractional order system is presented and discussed.

### 3.5.2.2 Example 1: Comparison of RMPC for a Multi Fractional Order System Using Oustaloup Approximation and Time-Response-Invariant Approximation

Consider the fractional order system described by the following equation [62]:

$$G(s) = \frac{1}{0.8S^{2.2} + 0.5S^{0.9} + 1}. \quad (3.13)$$

First, approximation of the fractional order system using Oustaloup method [67, 68] and the step-response-invariant method are compared. The Oustaloup method is based on replacing the term  $s^\alpha$  by an integer order transfer function that has approximately the same frequency response in a specific frequency range. The Matlab function, “`ousta_fod`” [55] is used to obtain approximations for the terms  $s^{0.9}$  and  $s^{2.2}$ :

$$G_{\alpha}(s) = \text{ousta\_fod}(\alpha, N, w_b, w_h), \quad (3.14)$$

where  $\alpha$  is the order of fractional term,  $w_b$  and  $w_h$  are low and high transitional frequencies respectively, and  $N$  is the order of approximation. Figures 3.7, 3.8, and 3.9 show step response of fractional order system and the approximated models, error between step response of the fractional order system and the approximated models, and frequency response of fractional order system and the approximated models using Oustaloup method respectively. In this case, the same order,  $N$ , of approximation is used for approximating both  $s^{0.9}$  and  $s^{2.2}$ . Therefore, the order of the whole system will be  $2N$ . The order reported in figures mentioned are the total order of the system, meaning that to obtain the 4th order approximation, each  $s^{0.9}$  and  $s^{2.2}$  are approximated with order of  $N = 2$  using Oustaloup method. The high and low frequency limits are chosen as,  $w_b = 2 \times 10^{-2}$  rad/s and  $w_h = 2 \times 10^2$  rad/s. As it can be seen from Fig. 3.9 and Fig. 3.12, fractional system bandwidth is around 2 rad/s, so, the frequency range of 2 orders of magnitude higher and lower seems reasonable.

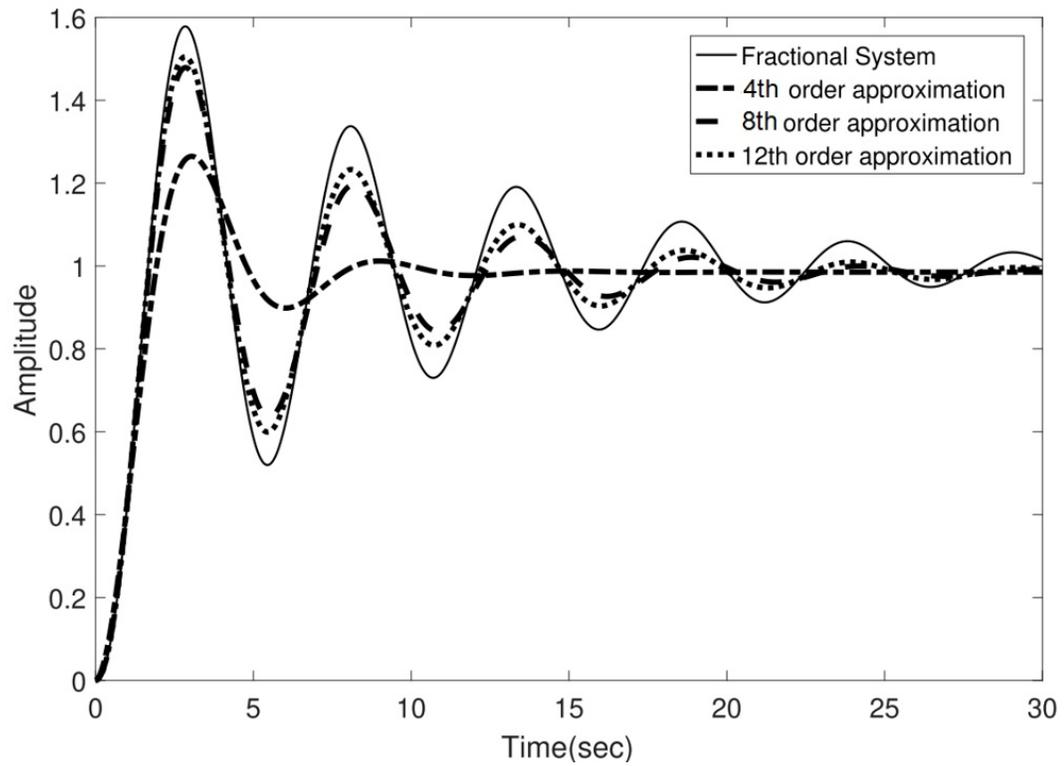


Figure 3.7: Step response comparison of models obtained using Oustaloup method with different orders.

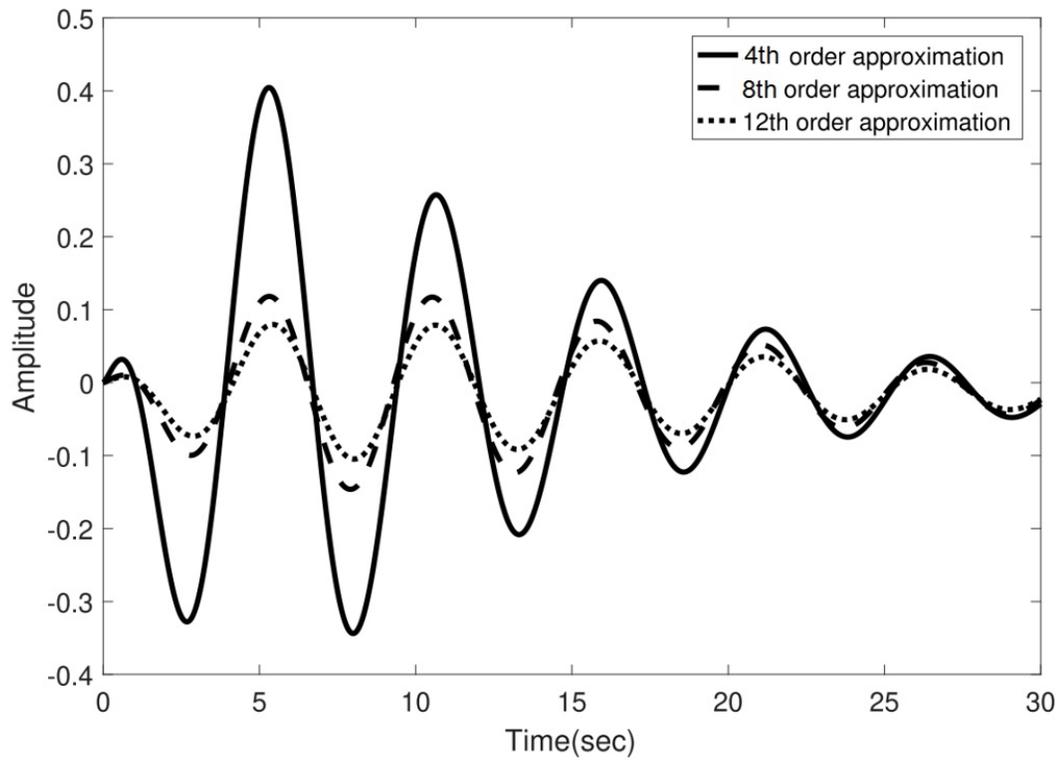


Figure 3.8: Error between step response of the fractional order system and approximated models using Oustaloup method.

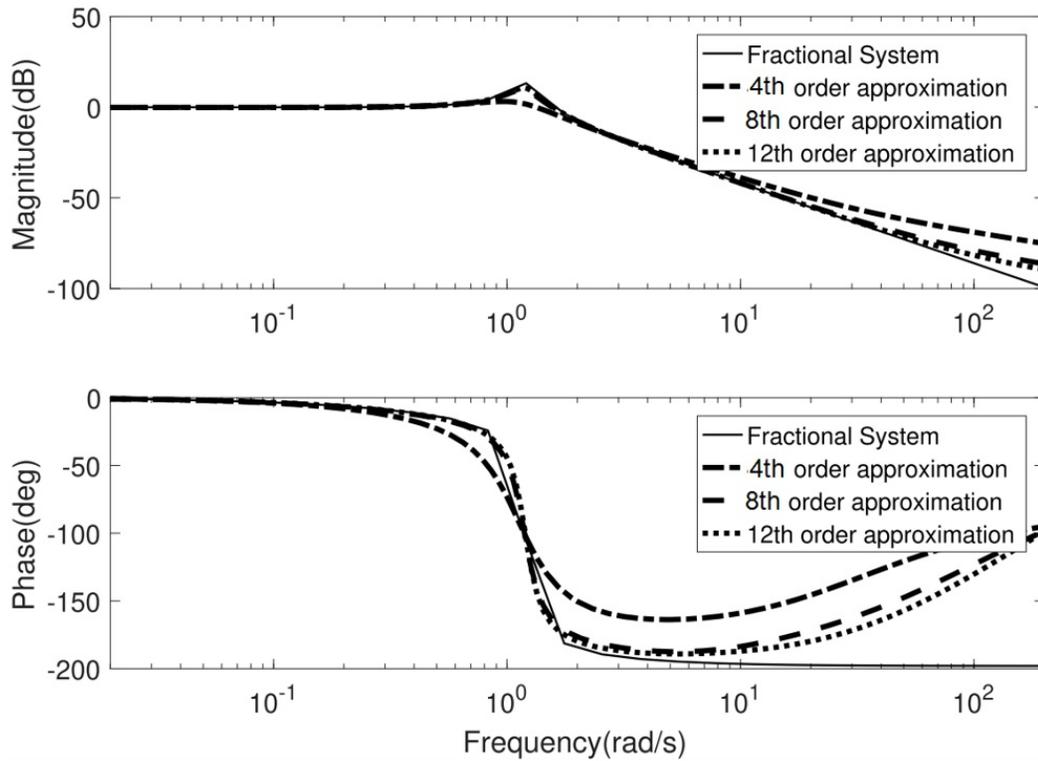


Figure 3.9: Frequency response comparison of models obtained using Oustaloup method with different orders.

Matlab `stmcb` function is used to obtain time-response-invariant approximation of the fractional order system. Equation (3.15) shows example application of this function:

$$[b, a] = \text{stmcb}(y, x, nb, na), \quad (3.15)$$

where  $y$  is the original system output to the input  $x$ ,  $nb$  and  $na$  are chosen orders of the approximated model numerator and denominator polynomials respectively and  $b$  and  $a$  are the approximated transfer function coefficients.

Figures 3.10, 3.11, and 3.12 show step response of fractional order system and the approximated models, error between step response of the fractional order system and the approximated models, and frequency response of fractional order system and the approximated models using the introduced time-response-invariant approximation method respectively. To obtain the approximations, unit step is used as the input signal  $x$  in (3.15). The output signal  $y$  is the fractional order system step response obtained using FOTF toolbox [69, 70]. The orders of numerator,  $nb$ , and denominator,  $na$ , are set to be equal since simulations show better estimation with the same order of numerator and denominator for this example.

Comparing Figs. 3.7, and 3.8 to Figs. 3.10 and 3.11 respectively, we can observe that to obtain a good time domain approximation, the Oustaloup method will lead to a very high order model, while, with the time-response-invariant method, even a second order model results in a good time domain approximation.

Oustaloup method is based on matching the frequency responses, however, even the 12th order approximation does not show a perfect frequency response fitting. Although the time-response-invariant method is based on matching the time domain response, Fig. 3.12 shows that the models obtained using this method have accurate frequency response although for a smaller frequency range. It is important to note that the fractional order system bandwidth is less than 2 rad/s and a good fitting up to 10 rad/s is practically acceptable.

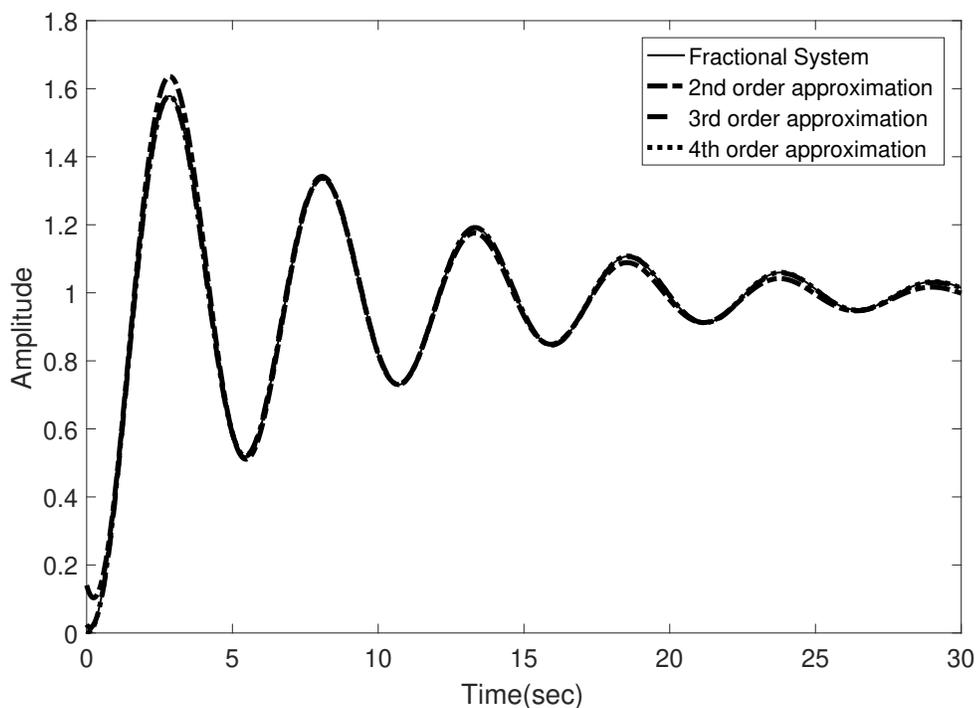


Figure 3.10: Step response comparison of models obtained using time-response-invariant approximation method with different orders.

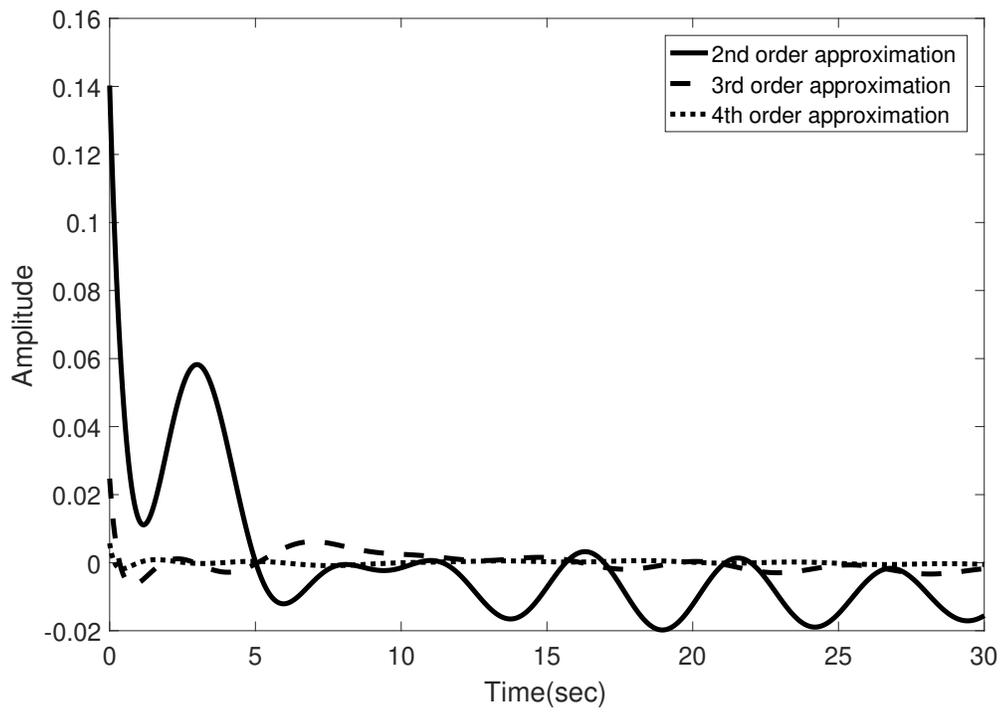


Figure 3.11: Error between step response of the fractional order system and approximated models using time-response-invariant approximation method.

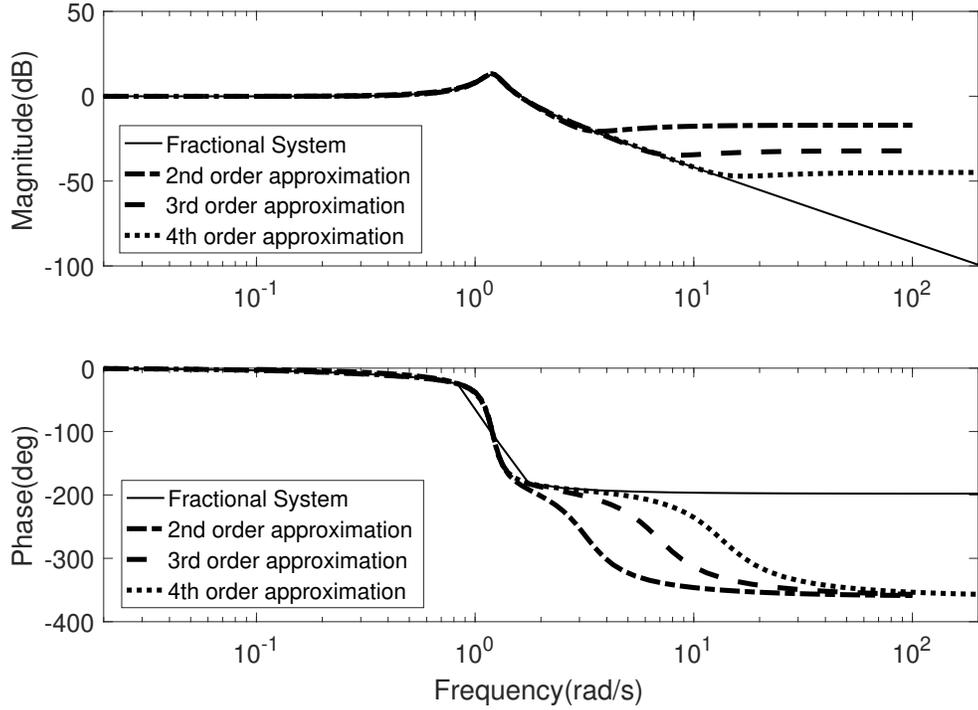


Figure 3.12: Frequency response comparison of models obtained using time-response-invariant approximation method with different orders.

### RMPC Comparison

Here, the results of applying RMPC to control the plant in Eqn. (3.13) using the two approximation methods are presented, compared and discussed.

One important issue to consider for MPC is the computational time. So, this is important to use the lowest possible order of internal model for MPC that satisfies the performance. In this case, an 8th order model for RMPC based on Oustaloup method is considered. The reason is that lower order models have more mismatch than the controller can stabilize the system and for higher orders, the approximated model becomes unobservable. For RMPC based on time-response-invariant method, a 4th order approximation is used since it was the lowest order approximation that shows satisfactory performance for this method.

Table 3.2: Comparison of computational time.

Method	Time(s)
Oustaloup method based RMPC	365.6
Time-response-invariant method based RMPC	76.8

Figures 3.13 and 3.14 show comparison of the system output and control effort for RMPC based on the two methods respectively. Although both methods demonstrate almost the same performance, RMPC based on Oustaloup approximation method requires more computational time because of the higher order model used. Table 3.2 shows the simulation time for each of the methods to obtain the results shown in Fig. 3.13 and 3.14. Another factor that contributes to the big difference in simulation time is that for RMPC based on time-response-invariant method, a low gain observer is used, i.e.  $k = 1$ . However, for RMPC based on Oustaloup approximation method, it was necessary to choose a higher gain for the observer, i.e.  $k = 4$ , to guarantee its performance.

For this example a simple form of the objective function in Eqn. (3.9) is used to reduce the number of control parameters:

$$J = \int_0^{N_p} ((y(t) - r(t))^T (y(t) - r(t))) dt. \quad (3.16)$$

The sampling time of simulations is 0.5 second and the prediction horizon is  $N_p = 10$ .

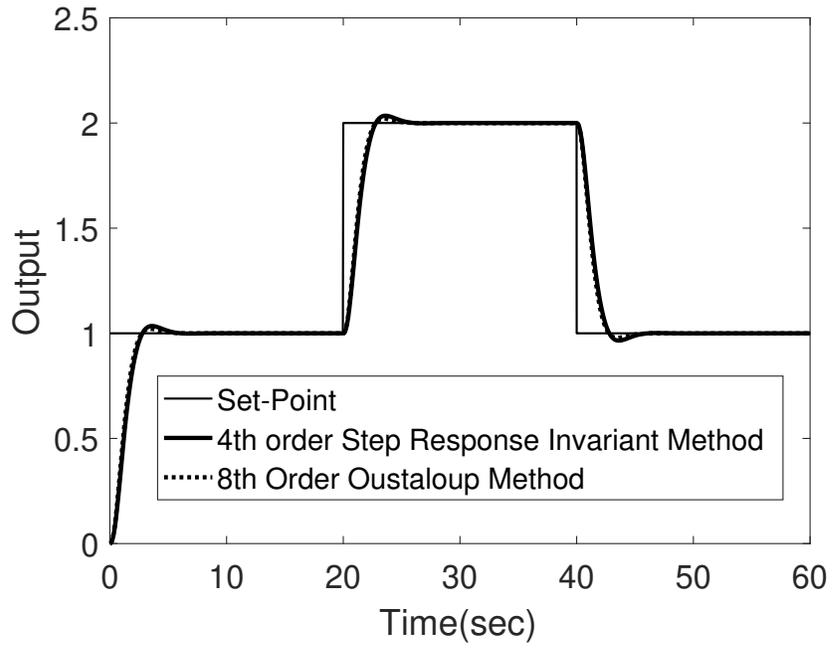


Figure 3.13: Closed loop response of the fractional order system in example 1 under RMPC.

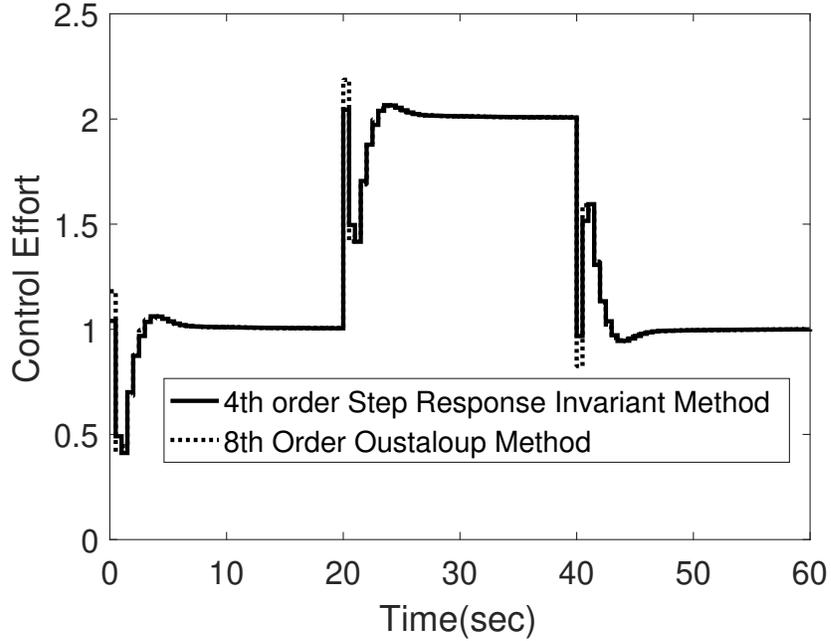


Figure 3.14: Control effort of RMPC controller for example 1.

### 3.5.2.3 Example 2: Application of RMPC for a MIMO Commensurate Fractional Order System Based on Time-Response-Invariant Approximation Method.

Consider the following commensurate fractional order system with three fractional order terms:

$$G(s) = \frac{K(-bs^\alpha + 1)}{a_2s^{2\alpha} + a_1s^\alpha + 1}, \quad (3.17)$$

where  $a_2 = 87.27$ ,  $a_1 = 25.33$ ,  $b = -21.41$ ,  $K = 3.45$  and  $\alpha = 0.823$ . This model is identified for a two-by-two thermal system in [71].

Figures 3.15, 3.16, and 3.17, show the time-domain and frequency-domain responses of approximated models of this fractional order system using the two methods. Having three fractional order terms leads to very high order overall approximation using Oustaloup method and a proper fitting is still not achieved. In this case, the frequency range of  $[10^{-2} \ 10^2]$  rad/s is considered since the system bandwidth is around 1 rad/s. For this example, even a 6th order approximated model (2nd order approximation for each fractional order) using Oustaloup method is not observable and it is not possible to perform RMPC based on Oustaloup approximation.

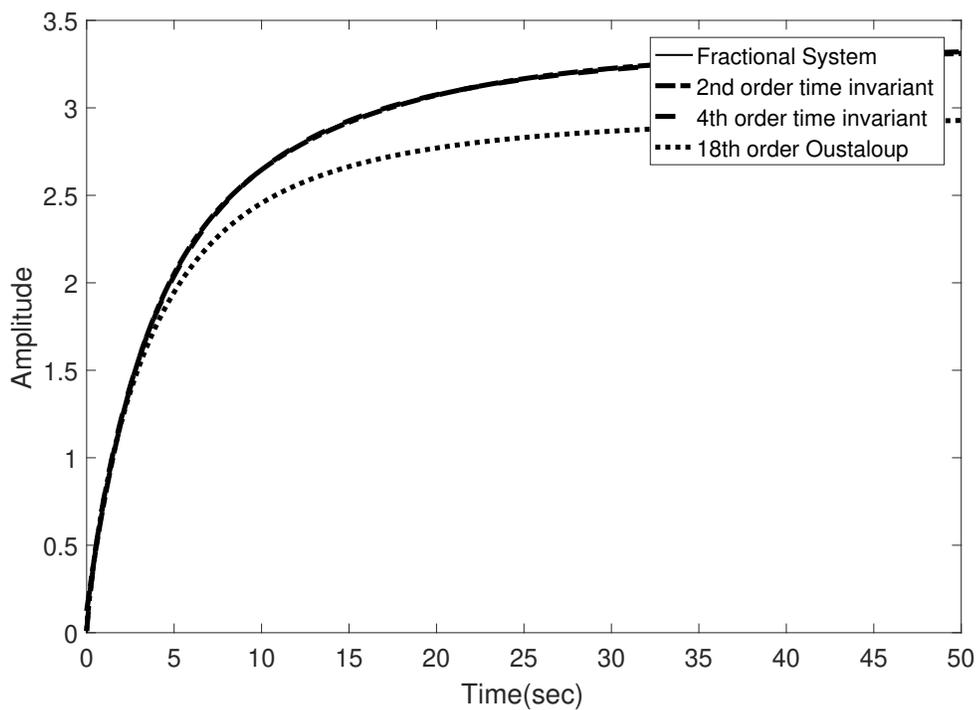


Figure 3.15: Step response comparison of models obtained using time-response-invariant approximation and Oustaloup method with different orders.

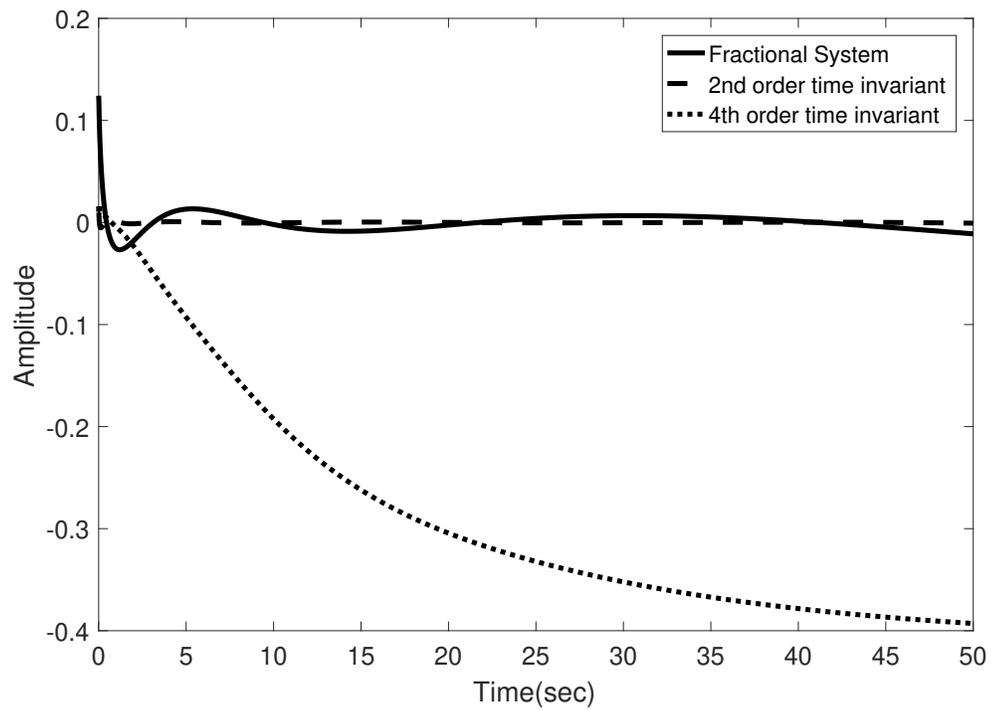


Figure 3.16: Error between step response of the fractional order system and approximated models using time-response-invariant approximation method and Oustaloup method.

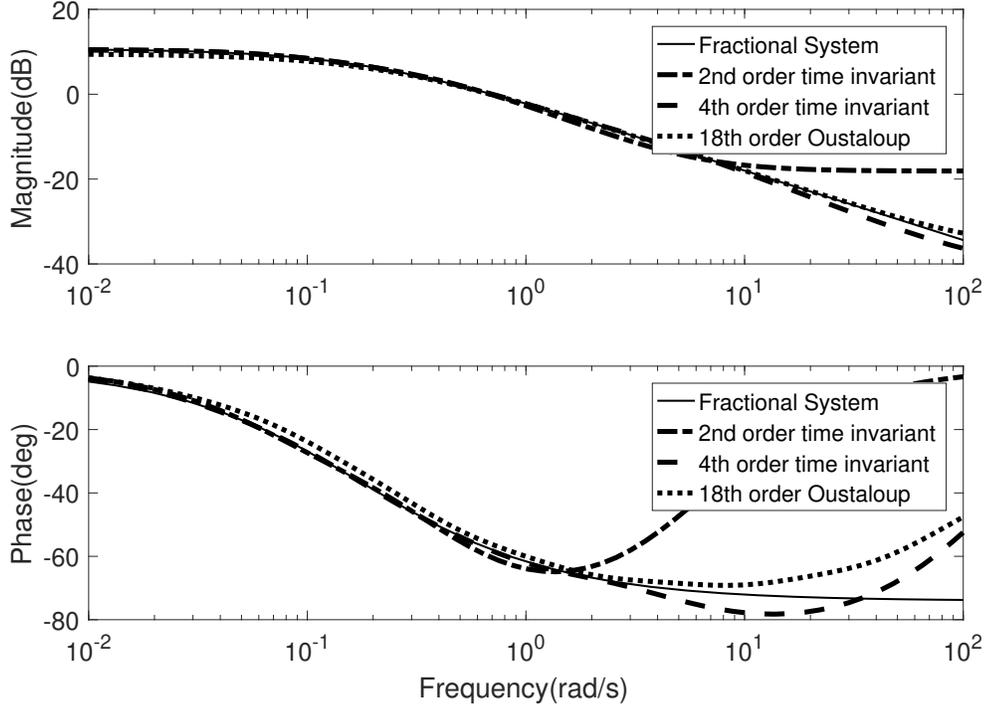


Figure 3.17: Frequency comparison of models obtained using time-response-invariant approximation method and Oustaloup method with different orders.

To demonstrate the capabilities of the introduced approach, the following coupled MIMO system is considered:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} G(s) & \frac{2}{s+3} \\ \frac{1}{s+2} & G(s) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (3.18)$$

where  $G(s)$  is the commensurate fractional order system introduced in Eqn. (3.17).  $G(s)$  is approximated using step-response-invariant method with the order of  $nb = na = 3$  as explained in previous example. RMPC is performed with the cost function in Eqn. (3.16) subjected to constraint on control input,  $u_1 \in [-11 \ 11]$  and  $u_2 \in [-3 \ 3]$ . The set-point for the outputs  $y_1$  and  $y_2$  are set to  $r_1 = 10$  and  $r_2 = 2$  respectively and a step disturbance with the value of 0.5 is added to output  $y_1$  at  $t = 10$  seconds. Sampling time and prediction horizon are  $T_s = 0.5$  seconds and  $N_p = 10$ .

The results are shown in Fig. 3.18 and Fig. 3.19. As shown in Fig. 3.18, RMPC using 3rd order step-response-invariant approximation can successfully

deal with control of MIMO commensurate fractional order system presented in Eqn. (3.18). Moreover, the system returns to the set-point after the added disturbance on the first output (considering the 0.5 seconds sampling time) and the bump on the second output shows the effect of coupling on the system. The control inputs remain in the limits imposed by the constraint as shown in Fig. 3.19.

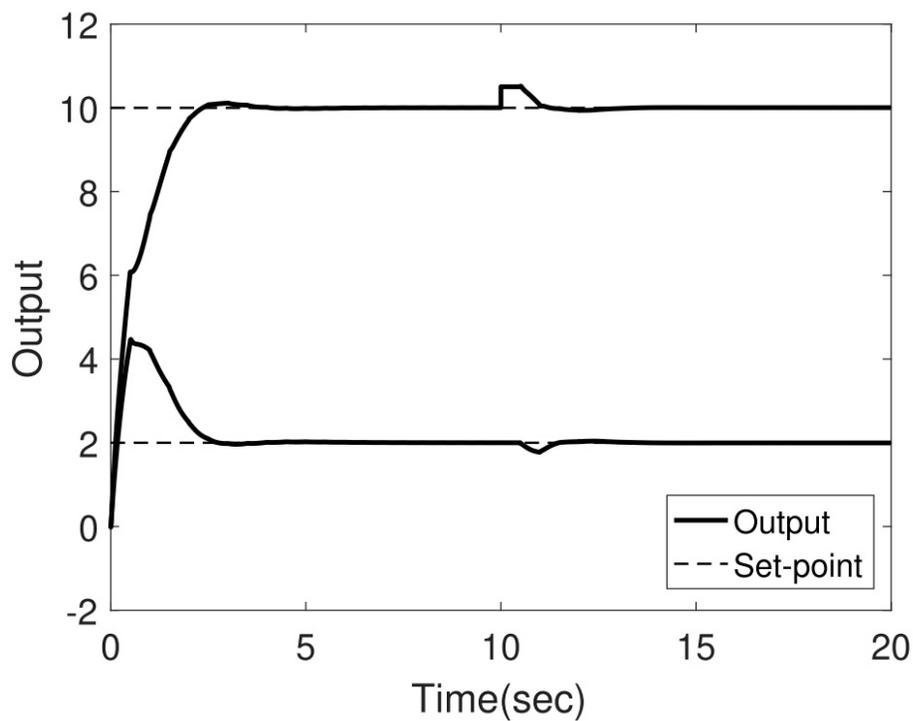


Figure 3.18: Closed loop response of the fractional order system in example 2 under RMPC.

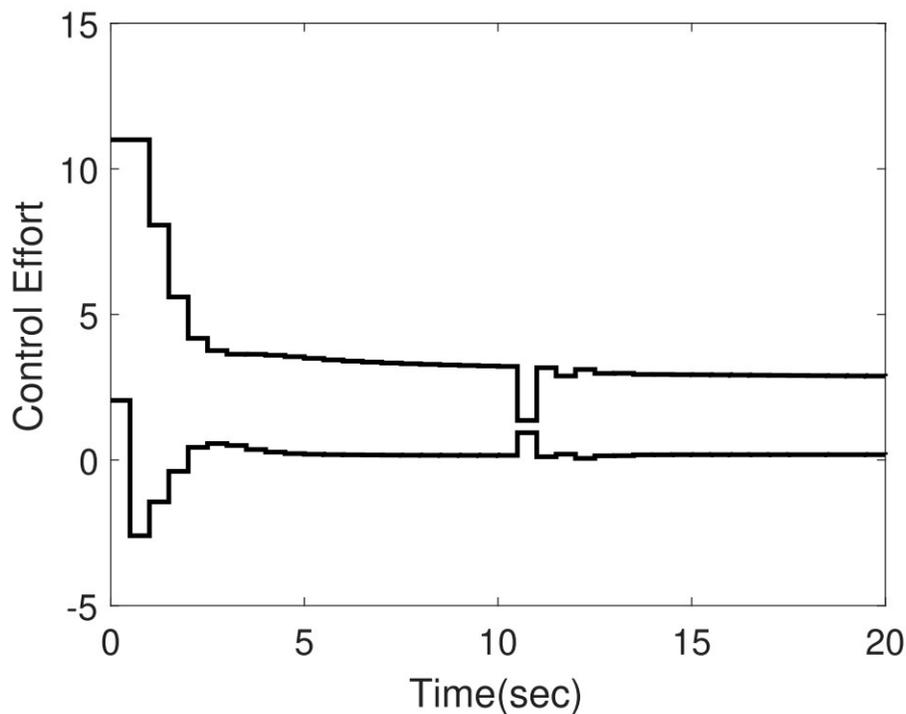


Figure 3.19: Control effort of RMPC controller for example 2. the top curve shows the input for first output,  $y_1$ , following the set-point  $r_1 = 10$ , and the bottom curves shows the input for the second output,  $y_2$ , following the set-point  $r_2 = 2$ .

### 3.6 RIOTS MPC with Preview for a Thermal Hardware-in-the-Loop System

#### 3.6.1 Introduction

In tracking and regulation tasks, the future reference data can be used as feedforward data for the control system to improve the system performance. This idea is known as preview control and was initially employed for thermal process control [72] and has been extended to process control and autonomous vehicles [73–78]. In these works, the preview control problem is formulated as an extended state-space model formulation. The  $h$  preview values available from the reference are considered as extended states of the system to design a Linear-Quadratic-Regulator (LQR) or a Linear-Quadratic-Integral (LQI).

On the other hand, the preview control idea has also been combined with Model Predictive Control (MPC), injecting the preview information into the control system optimal criterion to find the optimal control action. For example, [79] presents preview control implementation with stochastic MPC control for energy management of electric vehicles. In [80], the design of an MPC with the preview

for the setpoint optimization for active suspension actuators is proposed based on a finite preview horizon.

This work presents the design and practical implementation of model predictive control with preview using Recursive Integration Optimal Trajectory Solver (RIOTS) toolbox for solving the optimal control problem (OCP) to control a thermal hardware-in-the-loop system. The preview controller is designed for setpoint trajectory tracking of a stepped profile signal. The thermal system corresponds to a Peltier thermoelectric module employed as heating and cooling element. The control algorithms are implemented in Matlab using Hardware In the Loop (HIL) configuration. The thermal process is identified using a stepped signal to obtain a discrete fourth order state-space model. A trajectory planner is employed to provide all the preview information available during the system operation for each sampling time. Setpoint tracking performance of the RIOTS MPC (RMPC) controller with preview is investigated and impact of some RMPC design parameters in performance are studied.

The main contribution of this work is the combination of preview control with RIOTS MPC for thermal process control, its practical implementation using edge computing devices, and its performance assessment under different preview horizon lengths.

### 3.6.2 RMPC with Preview

In order to improve RMPC using the knowledge of future system references (disturbances), this information should be incorporated in MPC objective function. The main benefit of MPC in this case is that at each sampling time, the whole OCP is solved again. This allows for direct application of preview information in the objective function without the need for state augmentation. Whereas, in  $H_\infty$  based control with preview, the optimal control problem is translated into a feedback control law where the preview information need to be considered in the optimal control problem through the augmented states.

Consider a MIMO system with  $m$  number of inputs and  $p$  number of outputs and assume that  $h$  (preview horizon) samples of future references for each output is known. Equation (3.19) presents the modified objective function considered in this work to incorporate the preview information in RMPC.

$$\begin{aligned}
 J = & \|W_g(y(N_p) - R)W_r\|^2 + \\
 & \int_0^{N_p} (\|W_l(y(t) - R)W_r\|^2 + \\
 & \|W_u(u(t) - mag(G^{-1}(s))r(t))\|^2)dt,
 \end{aligned} \tag{3.19}$$

where  $R$ , defined below, contains the current and all future reference information up to the preview horizon.

$$R = \begin{bmatrix} r_1(t) & r_1(t+1) & \dots & r_1(t+h) \\ r_2(t) & r_2(t+1) & \dots & r_2(t+h) \\ \vdots & \vdots & \ddots & \vdots \\ r_p(t) & r_p(t+1) & \dots & r_p(t+h) \end{bmatrix}.$$

All the weighting matrices in (3.19) are defined the same as in Section 3.4, except for  $W_r$  which is a  $h$  by  $h$  matrix to weight the error associated to different elements of preview information separately. For the systems that some amount of control input to system is needed to maintain the desired output (reference) in steady-state, it is necessary to regulate the control-input ( $u$ ) penalization term using the open-loop system model to avoid steady state error due to control-input penalization.

### 3.6.3 Case Study: A Peltier System

#### 3.6.3.1 MESA Lab Edge Computing Based Peltier Box

The MESA Lab Edge Computing Based Peltier temperature control platform shown in Fig. 3.20 is employed as case study in this work. The system is composed of a Peltier cell (M1) that works as a heating element and a thermal infrared camera (M2) as a temperature feedback sensor to perform temperature distribution measurement and control. A LattePanda board (M3), running Windows 10 64 bits and Matlab, is used for Hardware in the Loop implementation. This board includes an Arduino Leonardo board to manage the power applied to the Peltier cell via Pulse Width Modulation (PWM) using the power driver (M4), and a battery (M5) with 4 hours of autonomy.

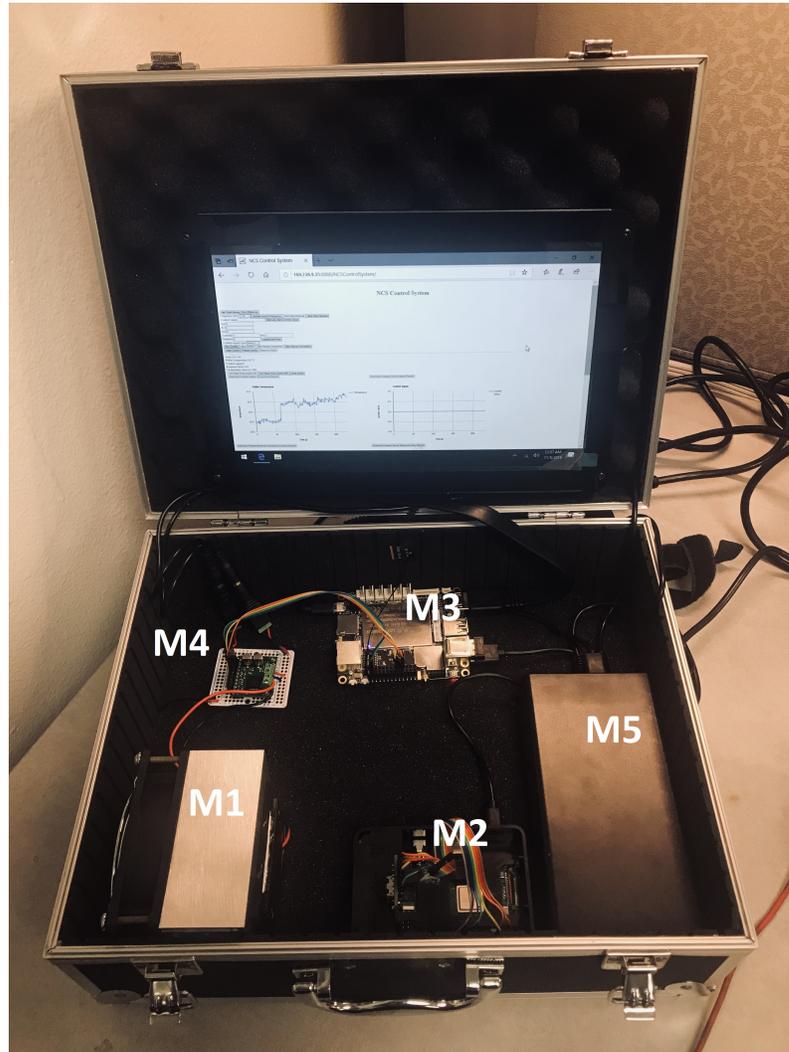


Figure 3.20: MESA Lab Peltier temperature platform

### 3.6.3.2 System Identification

The dynamic behavior of the Peltier module is identified applying a stepped signal to analyze the natural heating and cooling open-loop response for different operating points. With the system data, a four order discrete state-space model is obtained using the Matlab Identification toolbox with a sample time of 1 second given by (3.20). Figure 3.21 shows that the obtained model has a 91% of fitness, indicating a good representation of the system dynamics.

$$\begin{aligned}
A &= \begin{bmatrix} 0.990 & -0.1169 & -0.0102 & -0.1129 \\ -0.002 & 0.9718 & -0.1709 & -0.0178 \\ 0 & -0.002 & 0.9828 & -0.2836 \\ -0.0001 & -0.0013 & 0.1169 & 0.9611 \end{bmatrix} \\
B &= \begin{bmatrix} -0.0421 \\ -0.0129 \\ -0.0007 \\ -0.0020 \end{bmatrix} \quad C^T = \begin{bmatrix} -0.2102 \\ -0.3466 \\ -0.2559 \\ -0.2809 \end{bmatrix} \\
D &= [-0.0421 \quad -0.0129 \quad -0.0007 \quad -0.0020]. \quad (3.20)
\end{aligned}$$

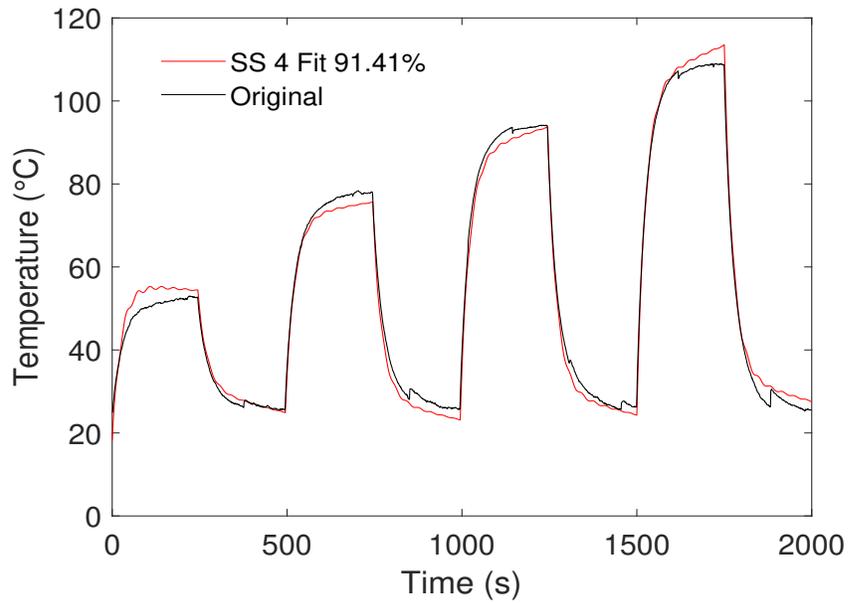


Figure 3.21: Identified model for Peltier system and fitting

### 3.6.4 Controller Implementation

A discrete state observer is designed using the classic discrete Luenberger [45] observer approach. The design procedure for the observer gains consists of initially calculating a closed-loop stabilizable controller for the system, which is made using the `dlqr` command in Matlab, returning the position of the close-loop system. From this closed-loop behavior, the observer poles are chosen to be 10 times faster than the system closed-loop poles. In this case, due to the fact that the observer is discrete time, the system poles should be located closer to the center of the unit radius circle. After that, based on the observer poles locations, the observer gains

are calculated using `place` command. For this system, the observer gains are given by  $L = [284, 198, 10.007, 23.74]$ .

Moreover, the preview controllers requires that the reference information inside the preview horizon is available for all the system sampling times to compute the control action. So, a trajectory planner algorithm was created to shift all the system trajectory in the preview horizon.

Figure 3.22 shows the Simulink model developed to implement RMPC with preview for the study case. As it is shown in this figure, three inputs are provided to the RMPC controller, the estimated state from the discrete state observer, the current reference for the system output, and the preview information generated using Preview Trajectory Planner. In this case the latter two inputs together make up the matrix  $R$  defined in Section 3.6.2.

Table 3.3 presents all the parameters that influence the performance of RMPC with preview as well as the base values used to obtain results for the study case in hand. The results shown in the results section are obtained using these base values, unless effect of change in one of these parameters is studied, which in that case the associated parameter values are showcased.

The choice of RMPC parameters is important in achieving stable and optimal performance. For example, increasing the prediction horizon might result in expensive OCP computation and destabilize the system due to not-in-time calculation by the controller for the considered sampling time.

Table 3.3: RMPC with preview, parameters

Parameter	Definition	Base Value
$k_{obs}$	Discrete Observer Gain	0.8
$W_g$	Terminal Cost Weight	1000
$W_l$	Trajectory Cost Weight	1000
$W_u$	Control penalization weight	1.8
$N_p$	Prediction/Control horizon	10
$dt$	Time-grid size for RIOTS	1 second
<i>solver</i>	RIOTS solver	RK4
$h$	Preview horizon (samples)	6
$W_r$	Preview error weight	$I$

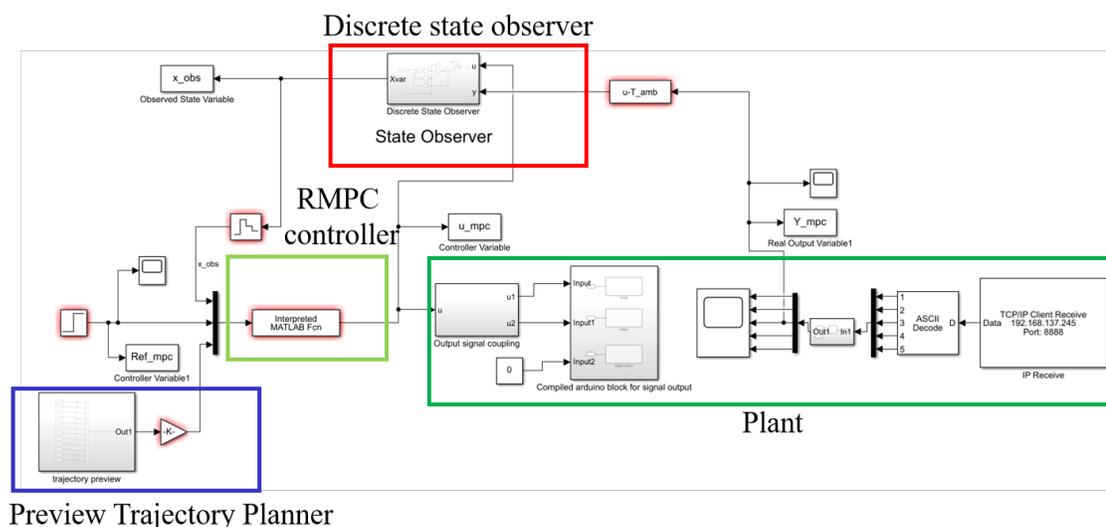


Figure 3.22: RMPC with preview controller

### 3.6.5 Results and Discussion

Figure 3.23 shows the systems behavior under RMPC with preview for different preview horizons. The objective is to track the step change from 20°C to 60°C. It can be observed from this figure that providing the preview information allows the system to react to the step change in advance and reach the set-point sooner. However, it seems that all three responses show the same rising slope and the system only leaves the previous set-point sooner when preview information is used. Looking at Fig. 3.24 which shows the associated control action, helps with better understanding the behaviour. The RMPC controller successfully moves the output towards the set-point with highest possible speed (saturated control input) and without any delay for this system. This means that there is no room for improving the rising slope or reducing any delay. So, for this system, preview information could only help the system to initiate the change sooner in the cost of leaving the current set-point earlier. It is also important to note that the behaviour of controller due to preview information could be adjusted based on a particular specification by designing the preview weight,  $W_r$ .

Figure 3.24 also shows that as the preview horizon is increased, the control action becomes harsher. This observation is explained by the fact that the terms in the current version of RMPC objective function (Eqn. (3.19)) are not normalized and adding the preview, changes the balance of the objective function towards less penalty for control action.

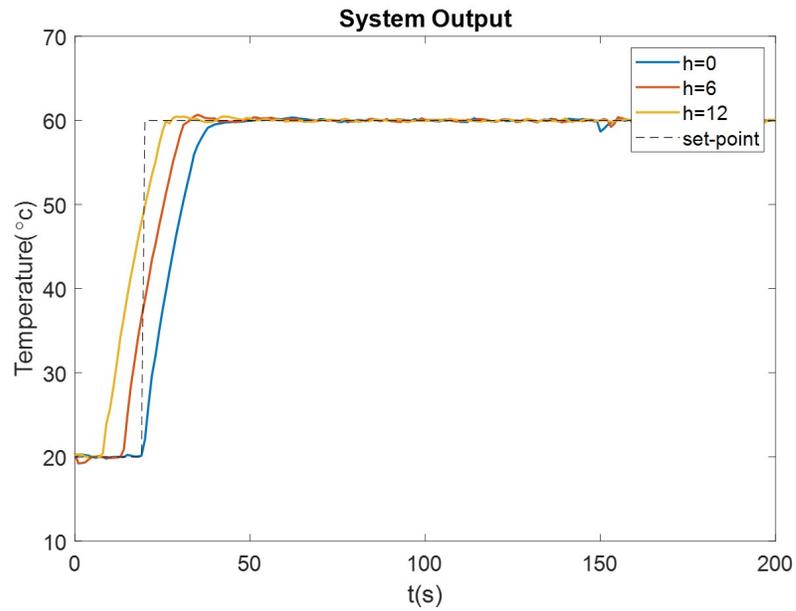


Figure 3.23: Plant output under RMPC with Preview for different preview horizons for set-point  $T = 60^{\circ}\text{C}$ .

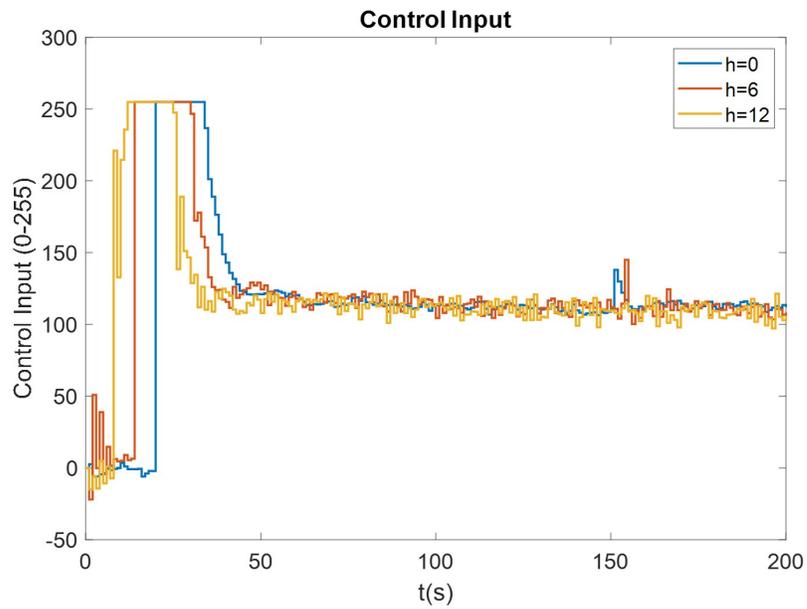


Figure 3.24: Control action for plant under RMPC with Preview for different preview horizons for set-point  $T = 60^{\circ}\text{C}$ .

Figure 3.25 and Fig. 3.26 show the controlled system behaviour for set-points  $T = 45^{\circ}\text{C}$  and  $T = 10^{\circ}\text{C}$  respectively. Considering that the system is nonlinear with respect to temperature and features asymmetric heating-cooling behaviour, these figures give some insight regarding controller robustness with respect to model uncertainties. Although some overshoot (less than 4%) is observed for the set-point of  $45^{\circ}\text{C}$ , the steady state behaviour shows no error and overall behaviour is acceptable. The model used to design RMPC is based on a heating process and as explained before, this system shows highly asymmetric behaviour with respect to heating-cooling. This explains the harsher steady state behaviour for the cooling process (Fig. 3.26). It is important to note that the observer model is also designed based on the heating process and inaccurate state estimation is the main factor in less-than-optimal controller behaviour in this case.

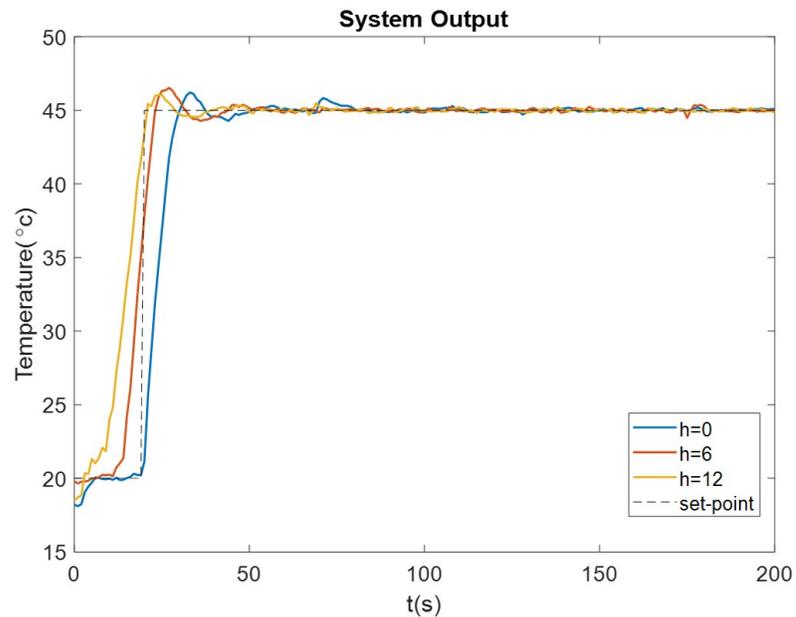


Figure 3.25: Plant output under RMPC with Preview for different preview horizons for set-point  $T = 45^{\circ}\text{C}$ .

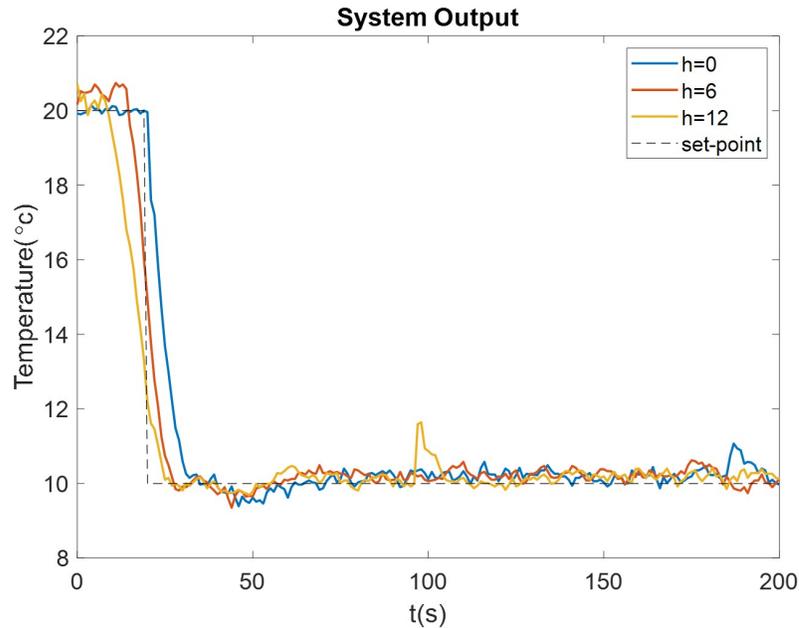


Figure 3.26: Plant output under RMPC with Preview for different preview horizons for set-point  $T = 10^{\circ}\text{C}$ .

In the following we look at the influence of variation in parameters such as prediction horizon, objective function weight balance, and observer gain, on the performance of the system. Figure 3.27 illustrates the system behaviour for different choices of prediction horizon. This figure shows that for prediction horizon equal to 20, the system starts oscillating and finally becomes unstable while for smaller prediction horizons this is not the case. This means that for this prediction horizon, the OCP has become computationally expensive and the controller becomes slow enough not to be in time to generate the control input for the next sample. However, considering the fact that the closed-loop system has a time constant of around 10 seconds, smaller prediction horizons should work and this is confirmed by the results. Although, reducing the prediction horizon too far can make the control action harsher and induce high frequency jitters in the behaviour of the system.

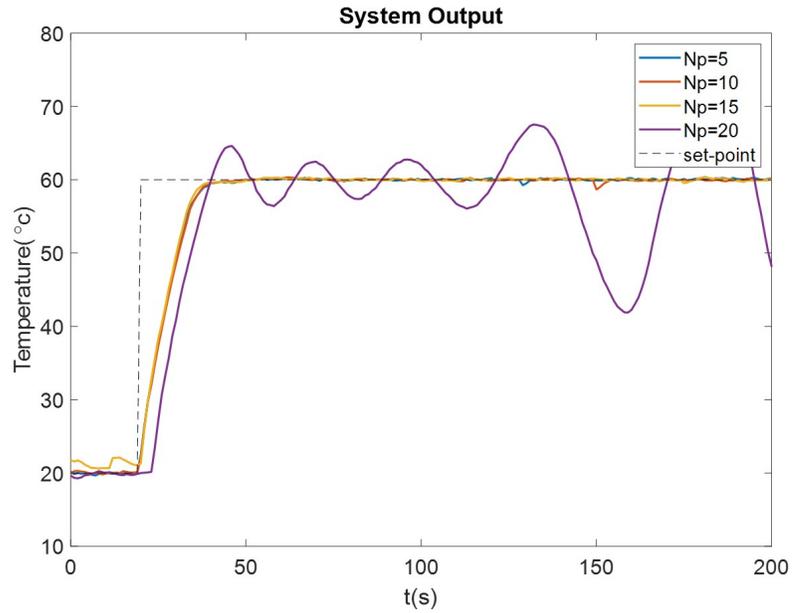


Figure 3.27: Plant output under RMPC for different prediction horizons.

Figure 3.28 and Fig. 3.29 show the system output and control action for different values of control input penalization weight,  $W_u$ , respectively. As it can be observed from these two figures, increasing the weight too high results in looser reference tracking while decreasing it makes the control action harsher to the point that can result in high-frequency oscillations on the output.

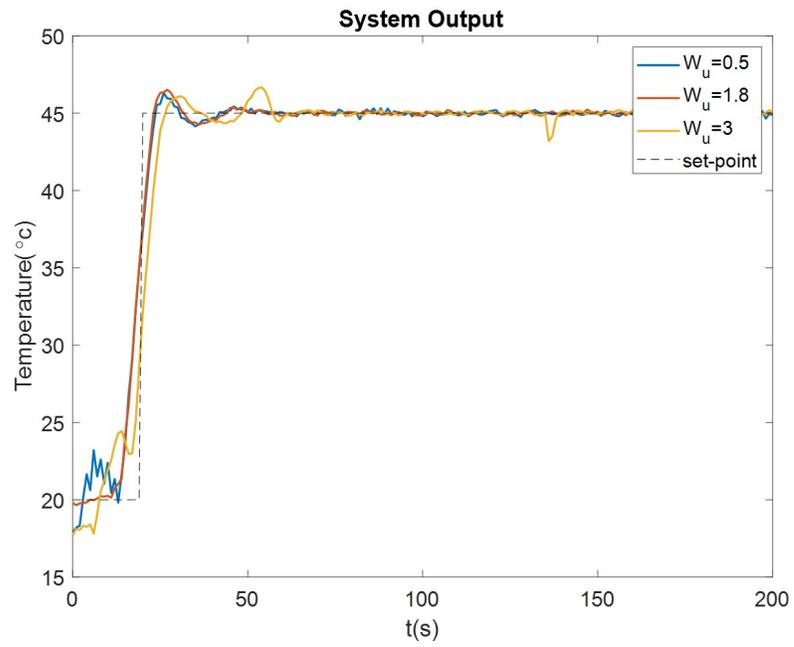


Figure 3.28: Plant output under RMPC for different control penalization weight,  $W_u$ .

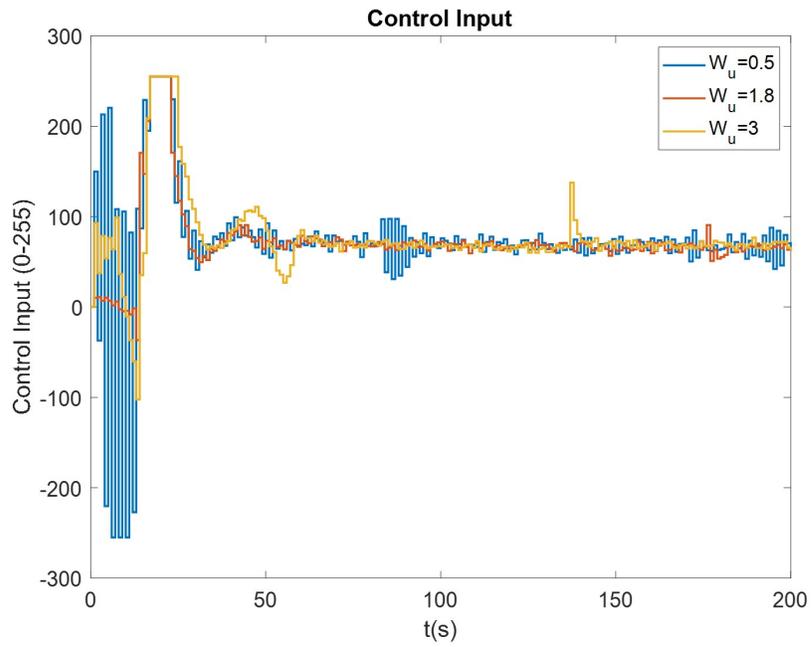


Figure 3.29: Control action of RMPC for different control penalization weight,  $W_u$ .

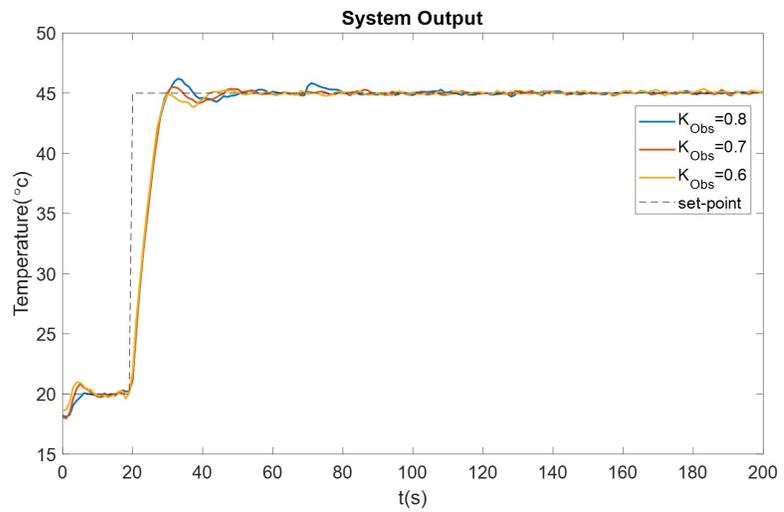


Figure 3.30: Plant output under RMPC for different observer gains.

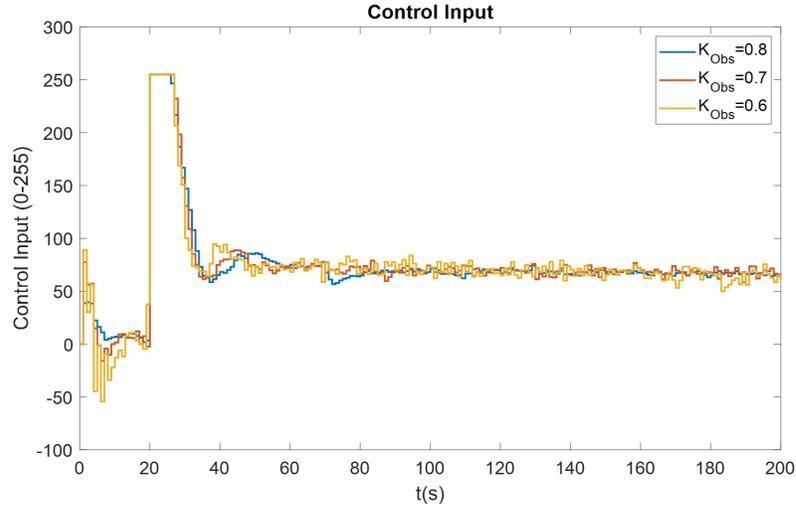


Figure 3.31: Control action of RMPC for different observer gains.

Figures 3.30 and 3.31 show the system output and control action for different values of observer gain respectively. According to the design of discrete observer considered for this work, higher gain  $K_{obs}$  is associated with slower observer dynamics. So, these figures show that faster state observer dynamics results in harsh control action (high-frequency jitters) which could be justified by the fact that the system output measured by the thermal camera is very noisy and following the output too closely results in amplification of that noise. While slower state-observer dynamics helps with partially filtering the feedback noise, making the dynamics too slow results in inaccurate state estimation and decline in performance.

### 3.7 Conclusion

This chapter presented the design and development of RMPC, which is a model predictive control toolbox based on RIOTS optimal control problem solver (RMPC), and its implementation to solve several control problems. The objective was to enable a straight forward approach for application of MPC to control a large class of SISO and MIMO systems. RMPC only requires an identified state-space model of the system and then tuning of the MPC parameters such as prediction horizon and objective function weights.

First, RMPC was used to control the refrigeration cycle provided by the IFAC PID2018 benchmark challenge and resulted in winning the challenge by achieving the best performance index among the contestants. Then, step-response-invariant approximation method was used to allow application of RMPC to control fractional order plants and the implementation to control a MIMO fractional order model was showcased in simulation. Finally, the capability to use a known future reference was

incorporated in RMPC to develop RMPC with preview and its implementation to control a thermal hardware-in-the-loop system was presented. A state-space model was identified using a set of I/O data and RMPC was implemented using embedded systems to control the experimental thermal system. The results showed optimal temperature reference tracking before preview information was added. The preview information allowed RMPC to react to the change in temperature reference ahead of time to achieve shorter settling time.

All in all, RMPC toolbox enabled straightforward implementation of model predictive control for process control applications and showcased the optimal performance expected from model predictive control for several experimental and simulation-based examples. We believe that development of a plug and play embedded RMPC with relay auto-tuning has a great commercial potential.

## Chapter 4

# SELF OPTIMIZING CONTROL OF AN RF MATCHING NETWORK FOR SEMICONDUCTOR MANUFACTURING APPLICATIONS

### 4.1 Introduction

Impedance matching is an essential part of plasma-enhanced processing in semiconductor manufacturing. In plasma-enhanced etching and deposition, semiconductor wafers undergo a chemical process in the presence of an RF generated plasma in the processing chamber. The process conditions such as mass flow rate of different gas species, the substrate temperature, plasma power, etc., which are determined by the so-called process recipe, continuously manipulate the electrical impedance of the chamber. Therefore, active impedance matching networks, with closed-loop control systems are necessary to keep stabilizing the plasma and ensure an efficient power transfer to the chamber.

As the semiconductor manufacturing is following the Moore's law [7] and moving toward  $10nm$  and below, the original equipment manufacturers (OEMs) are facing the customers ever-changing demands. The next level of circuit patterning technology, such as 3D-NAND and stacked finEFT, requires shorter and shorter plasma processing times [3,8]. Shorter process time means that plasma instabilities can affect a more significant portion of the process and as a result the need for a more precise RF matching control with shorter time-to-match becomes inevitable. On the other hand, availability of computational power and sensing technologies at lower cost allow for pursuing more complex control systems for impedance matching to meet those new requirements; which serves as a motivation for the self optimizing control approach proposed in this work.

There are different types of tunable matching networks (TMNs) utilized to match dynamic loads in variety of applications, such as, wireless power transfer [81], adaptive RF receivers and transmitters [82,83], plasma drivers [84], and power converters [85]. Conventional TMNs can be classified into digital (adjustable among a set of discrete values) and analog (continuously adjustable). The analog TMNs utilize variable reactance elements whose value could be tuned in a continuous manner, while the digital TMNs use digitally switched arrays to implement variable reactance [86]. High power plasma drivers in semiconductor industry, conventionally employ analog TMNs based on stepper-motor-adjusted variable vacuum-capacitors due

to the need for accurate impedance matching and operation over a wide impedance range [86, 87].

Several control approaches for automatic tuning of analog TMNs based on stepper-motor-adjusted variable-capacitor are discussed in the literature, such as a decoupled PID based on magnitude and phase of output impedance [88], a seek (iterative error-based) and follow (decoupled PID) [87], a variable structure controller based on the circuit model [89], a seek (based on circuit model) and follow (decoupled PID) [90]. All of the above mentioned control approaches are designed based on impedance matching configuration with a single-sensor at the upstream of the matching network, with the sensor ultimately measuring the total output impedance in terms of magnitude and phase. Considering the non-linear nature of the problem, the non model-based approaches (such as decoupled PID) lack optimal performance while the model-based approaches have limitation in stability and robustness due to lack of observability over potential model mismatches.

In this work, we present two control approaches based on self optimizing control framework utilizing a double-sensor configuration where in addition to the sensor measuring the output impedance at the upstream of the match, another sensor measures the load impedance at the downstream of the match:

- The Model Reference Algebraic Controller consisting of a match solver, a model mismatch compensator, and a trajectory generator.
- The Gradient-Based Control approach, pursuing minimum reflection trajectory for the matching.

A conventional decoupled proportional controller is also presented to use as a baseline to compare to the proposed control approaches.

## 4.2 Problem Description

In this section, first, impedance matching problem is defined in general and in the context of semiconductor applications, then an L-type matching network is introduced and formulation for calculation of match impedance and reflection coefficient is derived. Finally, the sensor configuration and actuator (motor-driven variable-capacitor) properties are presented to completely define the control problem in hand.

### 4.2.1 Impedance Matching

Impedance matching comes into play whenever power delivery in the context of electronics is discussed. It is the practice of designing input impedance and/or output impedance in order to achieve maximum power delivery and avoid reflection from the source. According to the maximum power theorem, the maximum power

is transferred from a source to a load, when the load impedance is equal to complex conjugate of the source impedance [91].

As it was mentioned in the introduction section, in the semiconductor industries, tunable matching networks (TMNs) are used to manipulate the overall load impedance for the objective of impedance matching. Moreover, the RF source input impedance is standardized to have a typical real impedance of  $50 \Omega$ . [87]. Considering that the source impedance does not have an imaginary part, the impedance matching problem can be described as tuning the matching network so that the overall output impedance becomes equals to source resistance at  $50 \Omega$ :

$$Z_o = Z_s = 50 \Omega.$$

#### 4.2.2 The Matching Network

While the ideas behind the proposed control methods can be applied to control any matching network that has a unique solution for any given load; we need a specific matching network to derive the formulation. Considering that L-type matching networks are the most common in semiconductor manufacturing applications, the L-type matching network shown in Fig. 4.1, is chosen for this project [87].

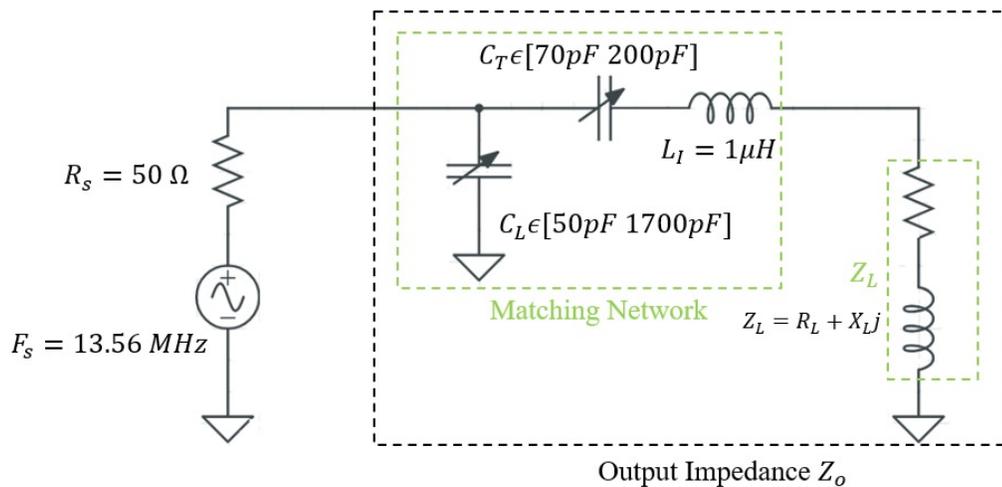


Figure 4.1: Circuit diagram of high-power RF plasma drive with an L-type matching network considered in this work.

This matching network has two motorized variable-capacitors called “Tune” capacitor,  $C_T$ , and “Load” capacitor,  $C_L$ , to manipulate the output impedance. The capacitance range for the Load and Tune capacitors, which is also shown in Fig. 4.1, are considered to be as follows:

$$C_T \in [ 70 \quad 200 ] pF, \quad C_L \in [ 50 \quad 1700 ] pF. \quad (4.1)$$

Considering the load impedance,  $Z_L$ , with resistance of  $R_L$ , and reactance of  $X_L$  ( $Z_L = R_L + X_Lj$ ); the capacitance range of the variable-capacitors, shown in Eqn. (4.1), allows matching load impedances within the following ranges:

$$R_L \in [0 \sim 50]\Omega, \quad X_L \in [0 \sim 100]\Omega. \quad (4.2)$$

While a variety of RF source frequencies are used in semiconductor processing for different purposes, we chose  $F_s = 13.56 \text{ MHz}$  as the RF source frequency which is a very common frequency used in plasma enhanced etching applications.

#### 4.2.2.1 Reflection Coefficient Definition and Formulation

Reflection coefficient is a parameter that characterizes the amount of reflected wave due to impedance discontinuity in power transmission line. For a power transmission system with source impedance of  $Z_s$  and overall output impedance of  $Z_o$ , the reflection coefficient,  $\Gamma$ , is obtained by:

$$\Gamma = \frac{(Z_o - Z_s)}{(Z_o + Z_s)}, \quad (4.3)$$

which then allows for calculation of reflected power using:

$$\frac{P_{reflected}}{P_{forward}} = |\Gamma|^2. \quad (4.4)$$

Here,  $P_{reflected}$  is reflected power and  $P_{forward}$  is the incident power provided by the source.

For the circuit shown in Fig. 4.1, the output impedance can be obtained by finding the equivalent impedance of the load and the match as follows:

$$Z_o = \frac{(Z_L + Z_{C_T} + Z_{L_I}) Z_{C_L}}{(Z_L + Z_{C_T} + Z_{L_I} + Z_{C_L})} \quad (4.5)$$

where  $Z_{C_T} = \frac{1}{j\omega C_T}$  is Tune capacitor impedance,  $Z_{C_L} = \frac{1}{j\omega C_L}$  is Load capacitor impedance,  $Z_{L_I} = j\omega L_I$  is impedance of the inductor,  $L_I$ , and  $\omega$  is the angular frequency of the RF power source. To match inductive plasma, the Tune branch of the match network has to be inductive and because of that the inductor  $L_I$  is used which is just a constant.

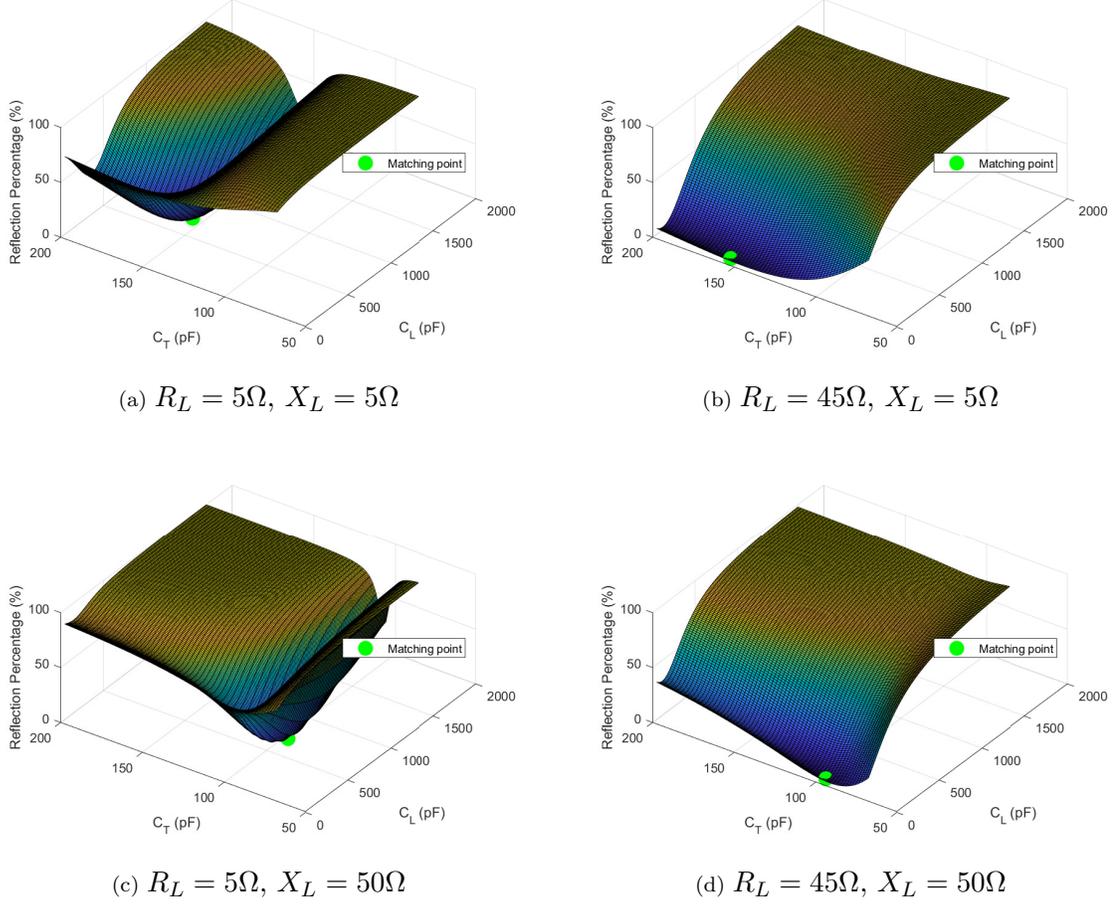


Figure 4.2: Reflection percentage ( $|\Gamma|^2$ ) over Load and Tune capacitance space for different values of load impedance specified in Eqn. (4.2).

#### 4.2.2.2 Motorized Capacitors

Commercial motorized vacuum-capacitors typically feature linear transformation between the rotational position of the motor shaft and the capacitance. Let us consider a variable-capacitor with capacitance  $C_s$  at one end and  $C_e$  at the other end and  $C_s < C_e$ . We define the capacitor rotational position associated with  $C_s$  as  $P_s$  and we assign it the position value of 0, and the capacitor rotational position associated with  $C_e$  as  $P_e$  and we assign it the position value of 100. The position unit is percentage of the whole positions range. So, the capacitor position associated to a given capacitance  $C$ , in percentage of motion range, can be obtained by:

$$P = \frac{100}{C_e - C_s} (C - C_s). \quad (4.6)$$

Similarly, at a given capacitor position percentage  $P$ , the capacitance of the capacitor can be presented by:

$$C = \frac{C_e - C_s}{100}P + C_s \quad (4.7)$$

so, the variable-capacitor reactance at position  $P$  can be derived as:

$$X = -\frac{1}{\omega \left( \frac{C_e - C_s}{100}P + C_s \right)} \quad (4.8)$$

where  $\omega = 2\pi F_s$  is the angular frequency of the RF power source.

This work is focused on analysis and design of high-level impedance matching control and not the low-level motion control of the motorized capacitors. Therefore, it is assumed that the stepper-motors that drive the variable-capacitors, come with an ideal motion controller that allows them to track a velocity reference with a maximum velocity limit,  $Vel_{max} = 50$  (%/s), for both Load and Tune capacitors. It is also assumed that the bandwidth of the velocity tracking motion control system of the motors are significantly higher than the rate at which high-level controller updates the velocity commands for the capacitors. As a result, motors immediately follow the velocity command and the dynamic of low-level motion control is ignored. The input to the motion controller is a voltage in the range of  $-24V$  to  $24V$  which is linearly transformed to the velocity input range of  $-Vel_{max}$  to  $Vel_{max}$  by the motion controller.

### 4.2.3 State-Space Formulation

The first step in formulating this problem in state-space is to determine the states. The impedance matching problem in the context of this work is defined as manipulating the output impedance of the system using the two variable-capacitors in the matching network, in order to match it to the input impedance of  $50\Omega$ . This means moving the magnitude of the output impedance towards  $50\Omega$  while moving its phase towards  $0rad$ . Therefore, it can be seen that this system can be expressed using two independent states, i.e. magnitude and phase of output impedance. Some alternatives for the states of the system are real and imaginary parts of output impedance, the magnitude and phase of reflection coefficient  $\Gamma$  defined in Eqn. (4.3), and real and imaginary part of reflection coefficient. Considering that reflection coefficient is more directly representative of amount of reflected power, real and imaginary parts of reflection coefficient are chosen as the states of the system:

$$\begin{cases} x_1 = \Gamma_r \\ x_2 = \Gamma_i \end{cases}, \quad (4.9)$$

where  $\Gamma_r = Re(\Gamma)$  and  $\Gamma_i = Im(\Gamma)$ .

Then, considering Eqn. (4.5) and (4.3), we can write the system differential equations as follows:

$$\begin{cases} \dot{x}_1 = h_1 \left( Z_L, \dot{Z}_L, C_L, C_T, u \right) \\ \dot{x}_2 = h_2 \left( Z_L, \dot{Z}_L, C_L, C_T, u \right) \end{cases}, \quad (4.10)$$

where the system input  $u$  is the input voltage to the two motorized capacitors which is associated with rate of change in the two capacitors rotational position:

$$u = \begin{bmatrix} \dot{P}_{C_L} \\ \dot{P}_{C_T} \end{bmatrix}. \quad (4.11)$$

Here,  $P_{C_L}$  and  $P_{C_T}$  are Load and Tune capacitor rotational positions in percentage of motion range respectively. The rate of rotational position for the two capacitors is associated to the rate of change in capacitance through Eqns. (4.7) and (4.1).

To write Eqn. (4.10) in this form, some assumptions are made:

- The RF frequency is constant and not supposed to change.
- Electrodynamic effects are ignored since they are orders of magnitude faster than the variable-capacitor control loop (electrodynamics are settled in tens of nanoseconds and will not impact the capacitor tuning).
- The characteristic impedance ( $R_s$ ) of the RF source is also constant. It is a common practice in RF systems to have  $R_s = 50 \Omega$ .

#### 4.2.3.1 Phase-Portrait Analysis

In this part we explain the methodology used to obtain phase-portrait plots to investigate stability of different control approaches in the later sections. Analytical derivation of differential equations of the system is nearly impossible due to complexity, so Matlab symbolic toolbox is used to derive the equations, and finally Matlab functions  $h_1$  and  $h_2$  in Eqn. (4.10) are generated such that given the values of  $Z_L, \dot{Z}_L, C_L, C_T$ , and  $u$ , the associated value of  $\dot{x}_1$  and  $\dot{x}_2$  are returned. Similarly, functions  $f_1$  and  $f_2$  in Eqn. (4.12) are also generated to allow for calculation of considered states.

$$\begin{cases} x_1 = f_1 (Z_L, C_L, C_T, L_I, \omega, R_0) \\ x_2 = f_2 (Z_L, C_L, C_T, L_I, \omega, R_0) \end{cases}. \quad (4.12)$$

Assuming a given plasma load ( $Z_L$ ), any chosen point in capacitor position-space will translate to a point in state-space and the differentials can also be calculated in that point which allows for drawing phase-portrait plots.

The plasma impedance and its derivative should be given for this analysis. If plasma impedance is not constant and as a result its derivative is non-zero, this means the system is time-varying and the vector field needs to be updated in each sample time.

#### 4.2.4 Sensor Configuration

Conventionally, a single sensor at the upstream of the matching network (as shown in Fig. 4.3) has been utilized for the purpose of impedance matching control. The upstream sensor is enough for a successful impedance matching control system design using non-model-based approaches, although the performance is typically sub-optimal and dependent on proper tuning. Any model-based approach based on a single-sensor configuration is susceptible to stability and robustness issues because of model uncertainties due to stray components in the RF system.

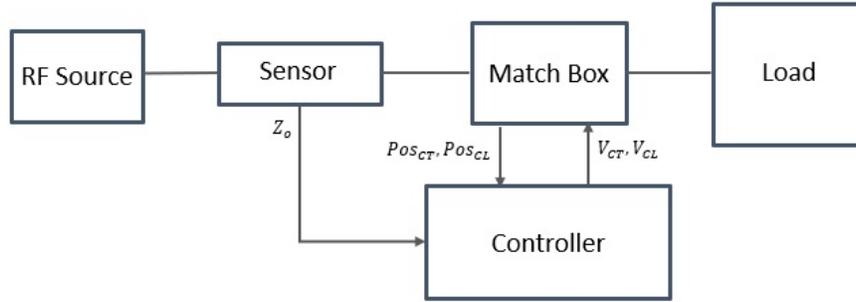


Figure 4.3: Single-sensor impedance matching control configuration for conventional control approaches.

In this work we propose two model-based control approaches that provide optimal performance; however, they rely on a double-sensor configuration shown in Fig. 4.4 to achieve robustness in stability and performance with respect to model uncertainties. The use of a second sensor measuring the load impedance is justified based on the following two statements:

- With the next level of circuit patterning technologies and the new requirements on shorter process-time and faster time-to-match, the performance improvement outweighs the extra sensor cost.
- There are many tool configurations such as multi-station tools where the load measuring sensors already exist for other purposes such as power distribution control, so those extra sensors could be leveraged for enhancement of impedance matching control.

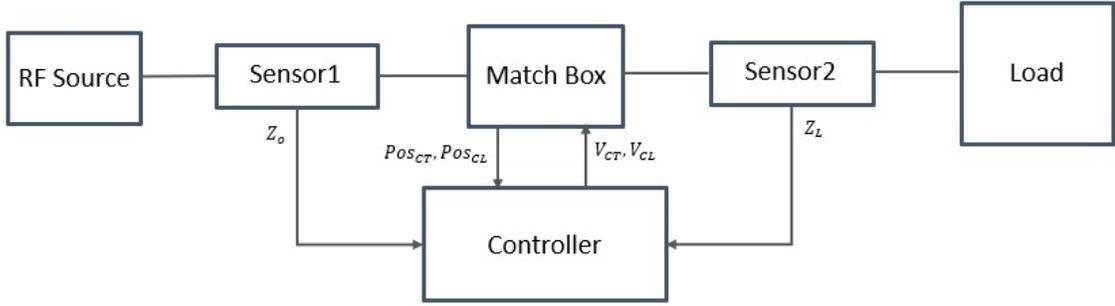


Figure 4.4: Double-sensor impedance matching control configuration for the proposed SOC approaches.

### 4.3 Decoupled Proportional Control

A conventional control method considered for L-type networks is to ignore the coupling in the system and use two separate single-input-single-output controllers. Figure 4.5 shows the schematic of the decoupled proportional control system which is derived from an article by Bacelli et al [87]. As it is shown in this figure, the reflection coefficient is calculated from the output impedance measured by the sensor feedback. Then, the real part of reflection coefficient is used to drive the Load capacitor, while the Tune capacitor is driven by the imaginary part of the reflection coefficient.

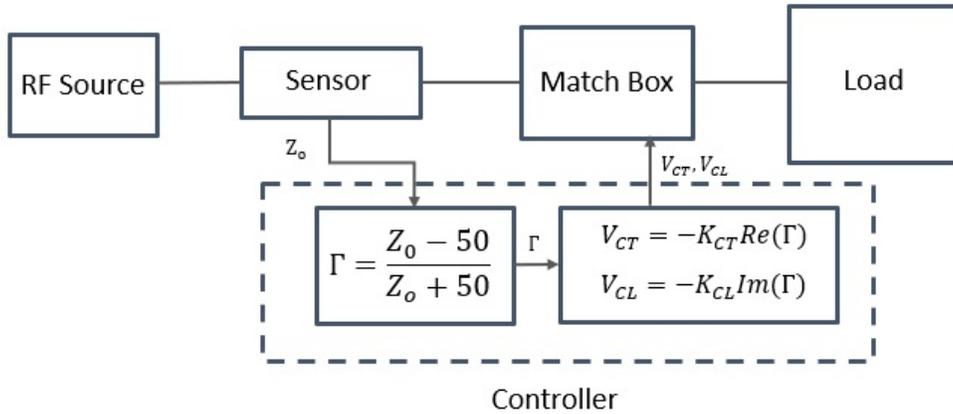


Figure 4.5: Schematic of conventional decoupled proportional control system.

Considering that the sensor feedback provides direct measurement of the reflection coefficient (as oppose to measuring the rate of change in reflection coefficient), an integrator is embedded in the feedback loop and the proportional control is enough to achieve successful control without any steady state error. Although adding integrator and derivative might improve the performance, considering the

non-linearity of the control problem with respect to the load impedance (refer to Fig. 4.2), complexity of PID tuning that shows globally robust performance for the whole load impedance range is a big downside.

This work is not focused on achieving the most optimal design for conventional decoupled proportional (or PID) control approach, but to use a version of it as a baseline to showcase the general difference in concept and performance. Therefore, for the result and analysis shown for this approach in the rest of this work, the proportional gains are chosen to be  $K_{CL} = -100$  and  $K_{CT} = 100$  unless mentioned otherwise. These values are obtained by trial and error through visual assessment by seeking reasonable performance over the whole range of load impedance.

### 4.3.1 Stability and Robustness

Phase-portrait plots are used in this work to evaluate the stability of different control approaches. Methodology used to obtain phase-portrait plots is explained in Section 4.2.3. So, by setting the control input function equal to the control law of the decoupled proportional control approach as:

$$u = K \begin{bmatrix} -Im(\Gamma) \\ -Re(\Gamma) \end{bmatrix} = K \begin{bmatrix} -x_2 \\ -x_1 \end{bmatrix} \quad (4.13)$$

where,

$$K = \begin{bmatrix} K_{CL} & 0 \\ 0 & K_{CT} \end{bmatrix} = \begin{bmatrix} -100 & 0 \\ 0 & 100 \end{bmatrix}$$

and assuming a constant load ( $\dot{Z}_L = 0$ ), the phase-portrait plot for a given load can be plotted.

Figure 4.6 shows phase-portrait plot when the system is controlled using the described decoupled proportional control for a certain load. It can be observed that for this load impedance, the system is stable and trajectories starting from the whole initial condition domain converge to the matching point.

In order to get a better understanding of how the Load and Tune capacitor position trajectories look like, it is really helpful to look at the phase-portrait plot when capacitor positions are used as states of the system.

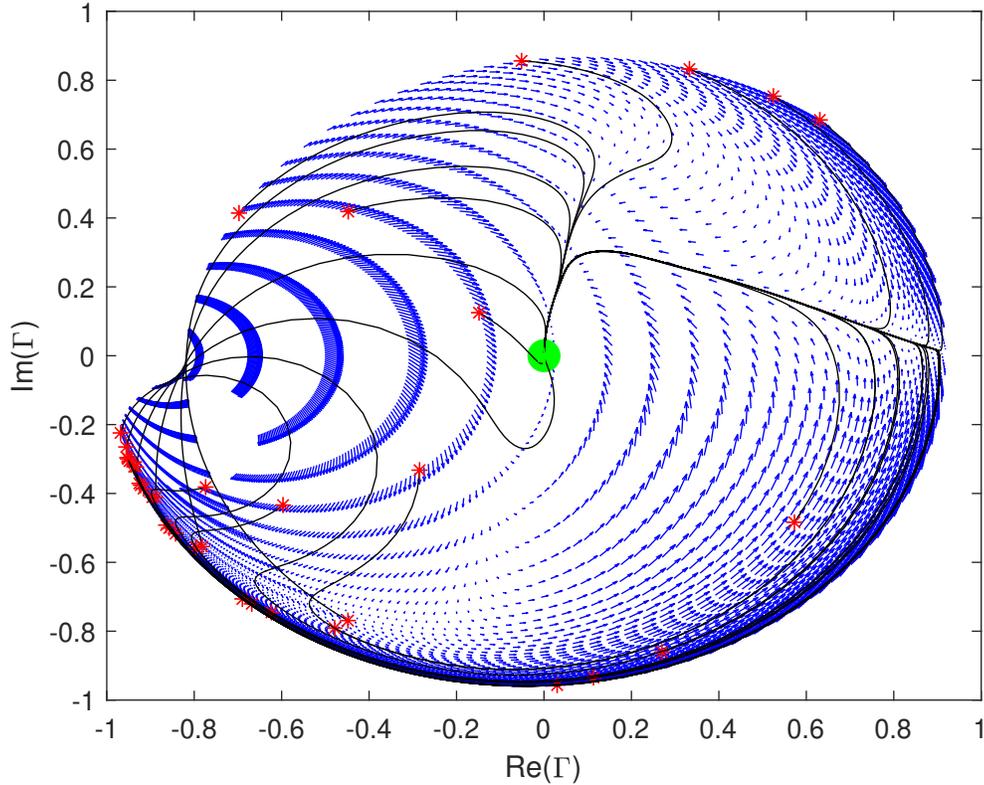


Figure 4.6: Reflection coefficient based phase-portrait plot for decoupled proportional control for the constant load  $R_L = 5\ \Omega$  and  $X_L = 50\ \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.

Figure 4.7 shows the phase-portrait plot for  $R_L = 5\ \Omega$ , and  $X_L = 50\ \Omega$  (similar to Fig. 4.6) when the capacitor positions are used as states of the system. Comparing the two plots provides better understanding of the relationship between reflection coefficient and the capacitor positions. For example, it can be observed that trajectories starting from the top right of Fig. 4.7 are the same trajectories starting from bottom left of Fig. 4.6. Although those trajectories eventually converge to the match-point, the Tune capacitor is initially forced to the 100 % saturation limit until the Load capacitor brings the system to the state from which it starts to converge to the match-point.

Overall, the decoupled proportional control shows globally stable behavior for a static load. However, for a realistic power dependent load, the trajectories

that start from the points with high reflection percentage or move to high reflection percentage state along their path could become unstable (plasma extinction) in practice.

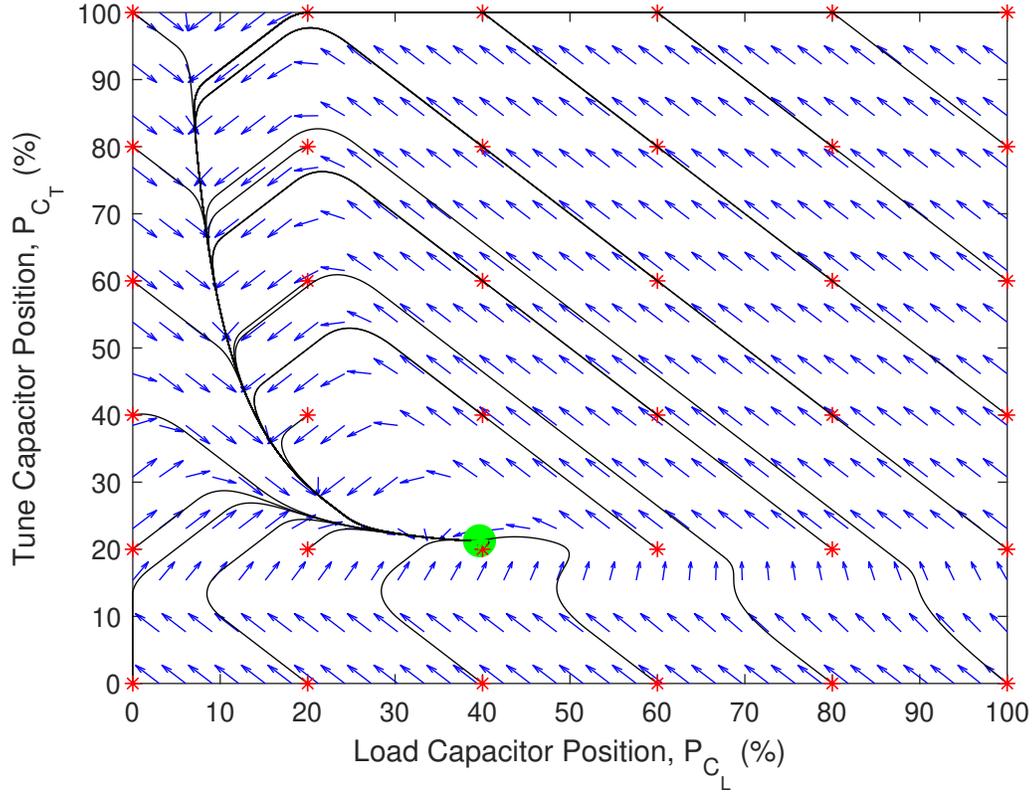


Figure 4.7: Capacitor position based phase-portrait plot for decoupled proportional control for the constant load  $R_L = 5\ \Omega$  and  $X_L = 50\ \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.

In this work we discuss robustness only with respect to model uncertainties. Considering the existing sensor technology (VI probes), the sensor measurement accuracy and sensor noise do not pose issues in practical applications. The sensor noise is easily handled with low-pass filtering due the fact that required update rate for the control is much slower than measurement rate; and existing sensor measurement accuracy is typically accurate enough to result in only ignorable levels

of steady-state reflection ( $\Gamma^2 \ll 0.01$ ) after matching to the impedance measured by the sensor.

Since the conventional approach does not use the model of the system, discussing the robustness with respect to model uncertainty is irrelevant. So, while in the later sections we evaluate the robustness of proposed control approaches with respect to model uncertainty, the conventional approach could be considered totally robust.

Considering that model uncertainties for matching networks are typically due to stray reactive components, we study the robustness of proposed controllers by assuming that the internal model for the controllers see up to 50 % more reactance at the downstream of the matching network. So, while the actual load impedance seen by the matching network is  $Z_L = R_L + X_Lj$ , we assume that the internal controller model sees  $R_L + 1.5X_Lj$  at the downstream of the matching network. Although this is not a comprehensive robustness study, it showcases the impact of an exaggerated practical model mismatch and how it is handled by the proposed controllers.

#### 4.4 Model Reference Algebraic Controller

In this section we present the Model Reference Algebraic Controller; however, we start by examining the Algebraic Controller without model reference adjustments to show why it is important to have a double-sensor configuration to enable inclusion of model reference adjustments to the control system.

##### 4.4.1 Algebraic Controller without Model Reference

The idea behind the Algebraic Controller is to simply solve for the exact match position and then drive the Load and Tune capacitors there. Given the value of the load impedance, it is always possible to solve for the exact match-point for any matching network (either using simple algebra, or numerical methods). The double-sensor configuration presented in Fig. 4.4 in Section 4.2.4 features direct measurement of the load impedance. So, for the L-type matching network presented in Section 4.2.2, and a measured load impedance  $Z_L = R_L + X_Lj$ , the desired Load and Tune capacitor reactances can be obtained by assuming that the output impedance in Eqn. (4.5) is matched to source impedance  $R_s$ , then:

$$\begin{cases} X_{CT_d} = \sqrt{R_s R_L - R_L^2} - X_L - X_I \\ X_{CL_d} = \frac{1}{\frac{X_L + X_I + X_{CT_d}}{R_L^2 + (X_L + X_I + X_{CT_d})^2}} \end{cases}, \quad (4.14)$$

where  $X_I$  is reactance of the series inductor.

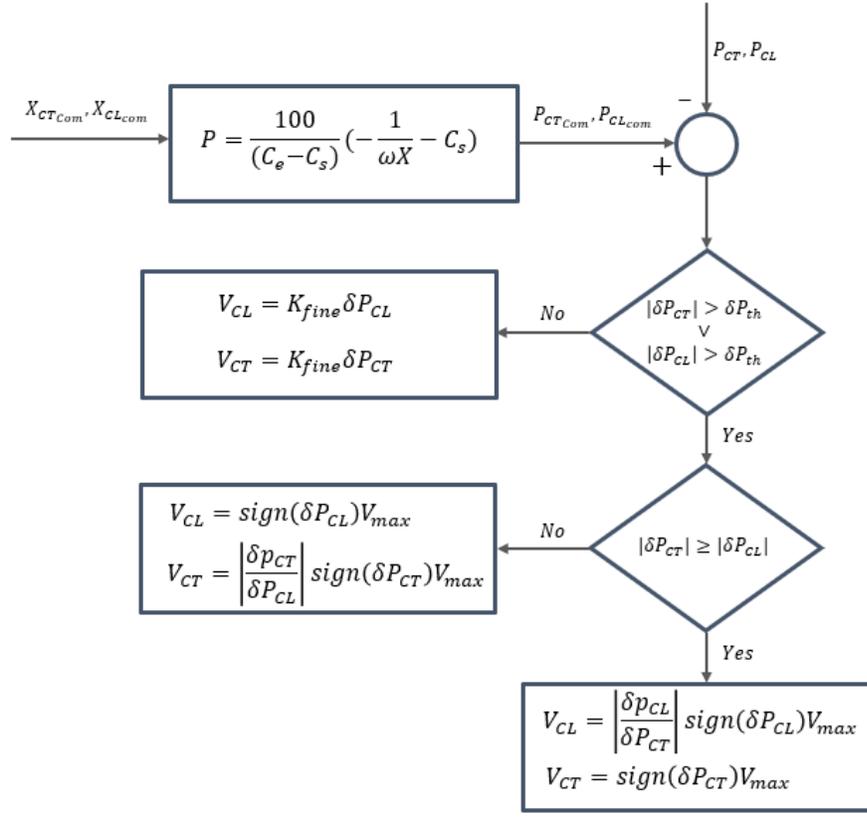


Figure 4.8: Flowchart of voltage command generation for the lower level motion controller by the high level matching controller, after desired/commanded capacitor positions are calculated.  $\delta P_{th}$  is the maximum position change possible in a controller loop time. As long as at least one of the capacitors is further from the commanded position than this threshold ( $\delta P_{th}$ ), then capacitors are moved with maximum speed while maintaining the direction in the capacitor position space; after that system switches into fine tuning mode with the gain  $K_{fine}$ , to avoid overshoot or potential oscillation.

After the desired Load and Tune capacitor reactances are known, the desired positions,  $P_{CL_d}$  and  $P_{CT_d}$ , can be obtained based on Eqn. (4.6) by substituting the capacitance associated to the desired reactance. Next step is to generate the velocity profile that the motors need to track, to reach the obtained desired positions (refer to Section 4.2.2.2). Since the desired positions are known, it is possible to drive the motors towards the desired positions with maximum speed, while maintaining the correct orientation towards the match-point in the capacitor position space, until the desired positions are within the reach in a single sample time. The flowchart in Fig. 4.8 demonstrates how the desired capacitor positions and current capacitor

position measurements are used to generate the voltage output required to optimally drive the motors towards the match-point.

#### 4.4.1.1 Stability and Robustness

Figures 4.9 and 4.10 show the reflection coefficient based and capacitor position based phase-portrait plots for the Algebraic Controller in the absence of any model mismatch, respectively. It is observed that all system trajectories are converging to the match-point, and the capacitor position based phase-portrait clearly shows that all trajectories follow straight lines towards the match-point as expected.

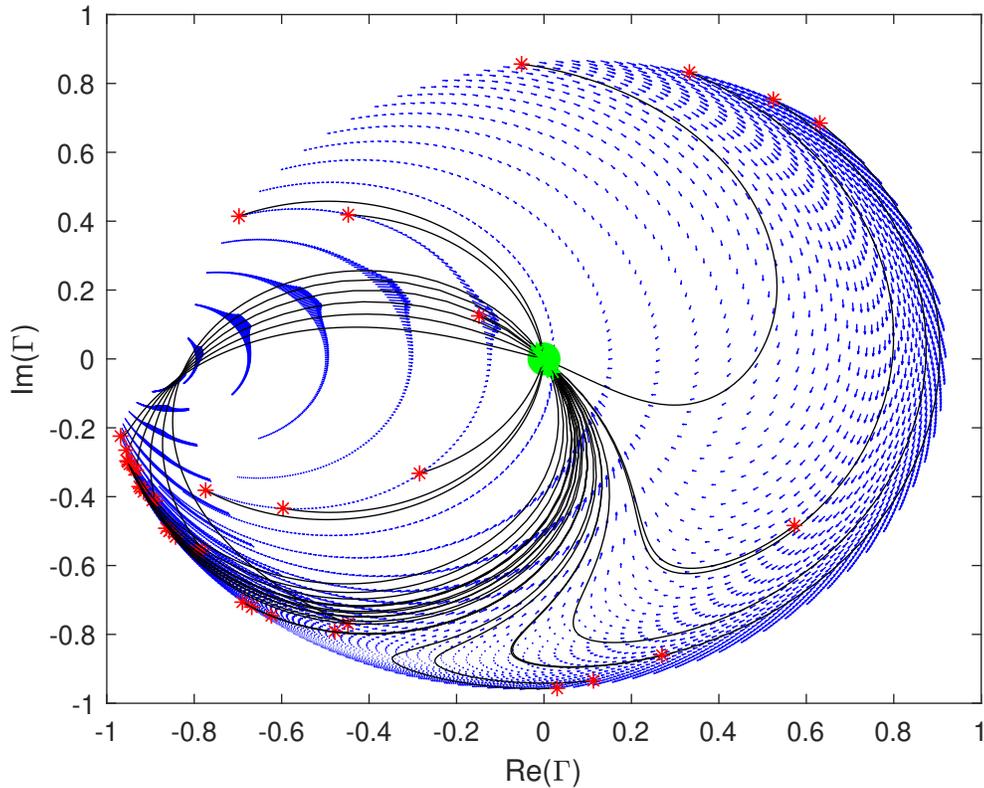


Figure 4.9: Reflection coefficient based phase-portrait plot for the Algebraic Control for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.

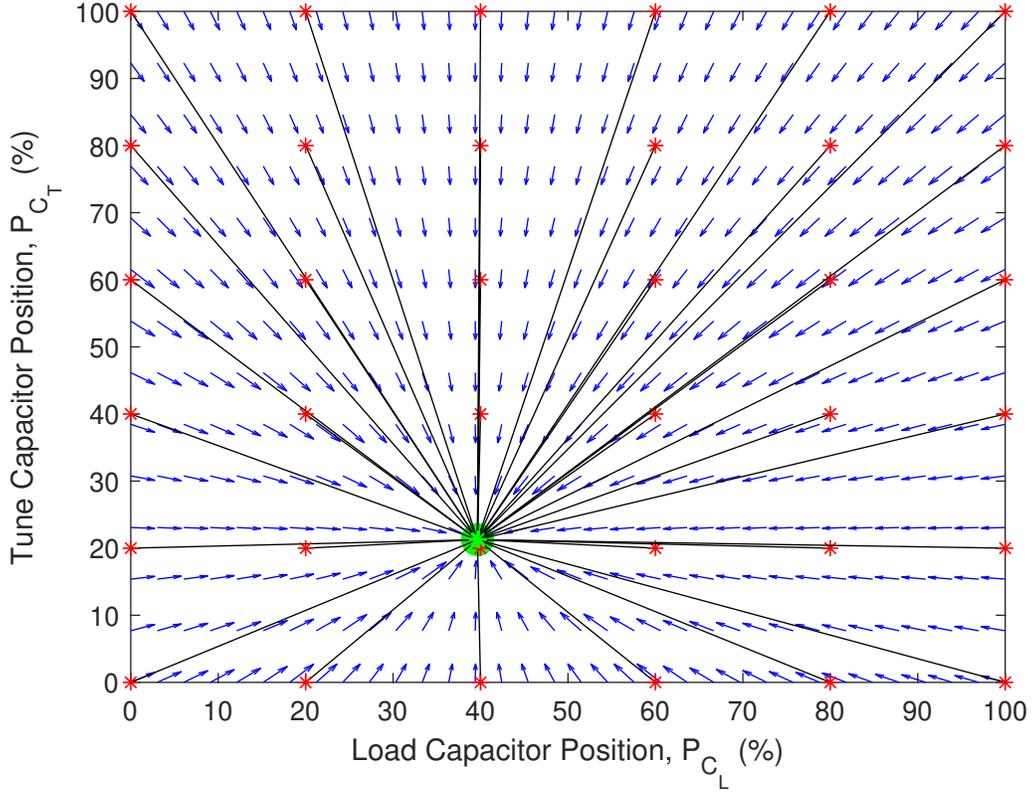


Figure 4.10: Capacitor position based phase-portrait plot for the Algebraic Control for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.

While this behavior seems favorable, Figs. 4.9 and 4.12 show the reflection coefficient based and capacitor position based phase-portrait plots in the presence of model mismatch of the form described in Section 4.3.1. This plots show that the described model mismatch would result in an offset in math-point calculation of the controller which will show up as steady-state error. While the capacitor position based phase-portrait shows the steady-state position distance from the match-point to be approximately 15%, the reflection coefficient based phase-portrait reveals that this steady-state error results in approximately 85% reflected power.

Overall, the Algebraic Controller without model reference shows global and robust stability for a static load with and without model mismatch. Note that even with the model mismatch, system output is bounded and technically considered

stable; however, with a realistic power dependent load, system would show unstable behavior in practice due to lack of power delivery to the load.

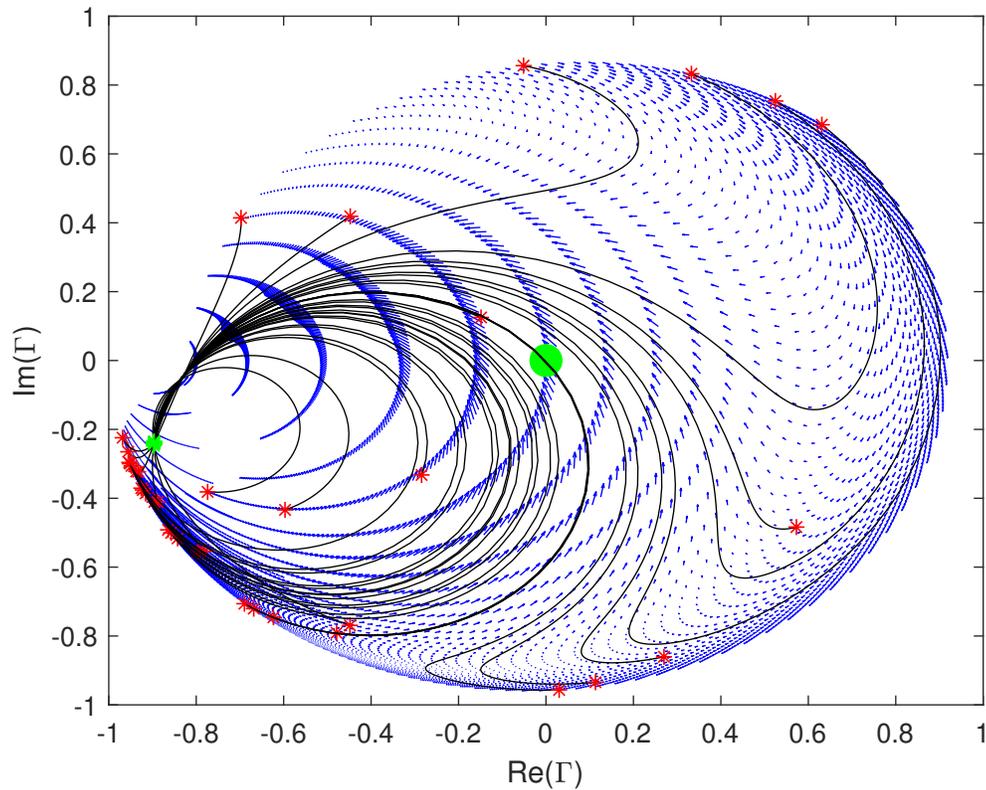


Figure 4.11: Reflection coefficient based phase-portrait plot for Algebraic Control for the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$  with added 50% reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.

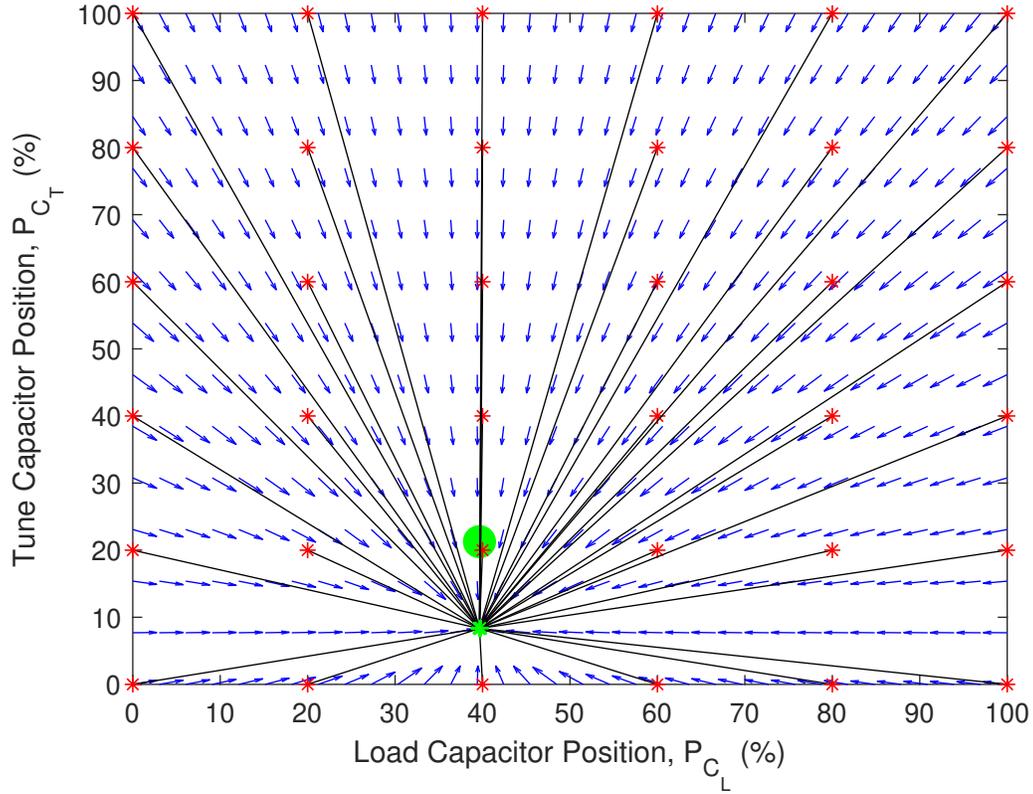


Figure 4.12: Capacitor position based phase-portrait plot for the Algebraic Control for the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$  with added 50% reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.

#### 4.4.2 Algebraic Controller with Model Reference

The redundant information provided by the extra sensor can be used to obtain a measure of the model mismatch and compensate for the steady-state error through an adjustment mechanism introduced with the Model Reference Algebraic Controller.

Figure 4.13 shows a high-level block diagram for the Model Reference Algebraic Controller; while, Fig. 4.14 provides the details of the adjustment mechanism used in this controller.

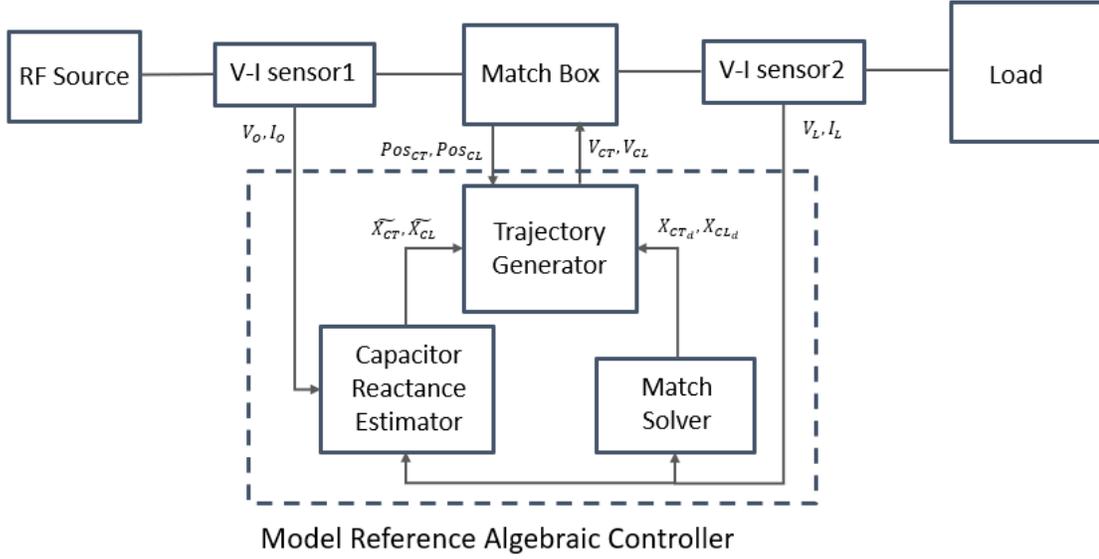


Figure 4.13: Block diagram of RF impedance matching with the Model Reference Algebraic Controller.

The availability of the two sensors allows for estimation of the Load and Tune capacitor reactances, using the measurements and the knowledge of matching network model. After substituting load impedance measurement ( $Z_L$ ) from VI Sensor 2 in Eqn. (4.5), the equivalent output impedance,  $Z_o$ , can be set equal to output impedance measured using VI Sensor 1 ( $Z_M = R_M + X_M i$ ), to obtain an estimation of the Load and Tune capacitor reactances:

$$\begin{cases} \tilde{X}_{CT} = \sqrt{\frac{|Z_M|^2}{R_M} R_L - R_L^2 - X_L - X_I} \\ \tilde{X}_{CL} = \frac{X_M}{|Z_M|^2} - \frac{X_L + X_I + \tilde{X}_{CT}}{R_L^2 + (X_L + X_I + \tilde{X}_{CL})^2} \end{cases} \quad (4.15)$$

Let us assume that having model mismatch will result in erroneous calculation of desired capacitor reactances. The adjustment mechanism is based on the idea that for each capacitor, the error between the actual matching reactance and the calculated desired reactance ( $|X^* - X_d|$ ), is the same as the error between the estimated reactance and the measured reactance ( $|\tilde{X} - X|$ ). So, the commanded capacitor reactances (the reactances that will be used to generate the command to motion controller), are calculated by adjusting the calculated desired reactance with the amount of drift due to model mismatch:

$$\begin{cases} X_{CTCom} = X_{CTd} + X_{CT} - \tilde{X}_{CT} \\ X_{CLCom} = X_{CLd} + X_{CL} - \tilde{X}_{CL} \end{cases} \quad (4.16)$$

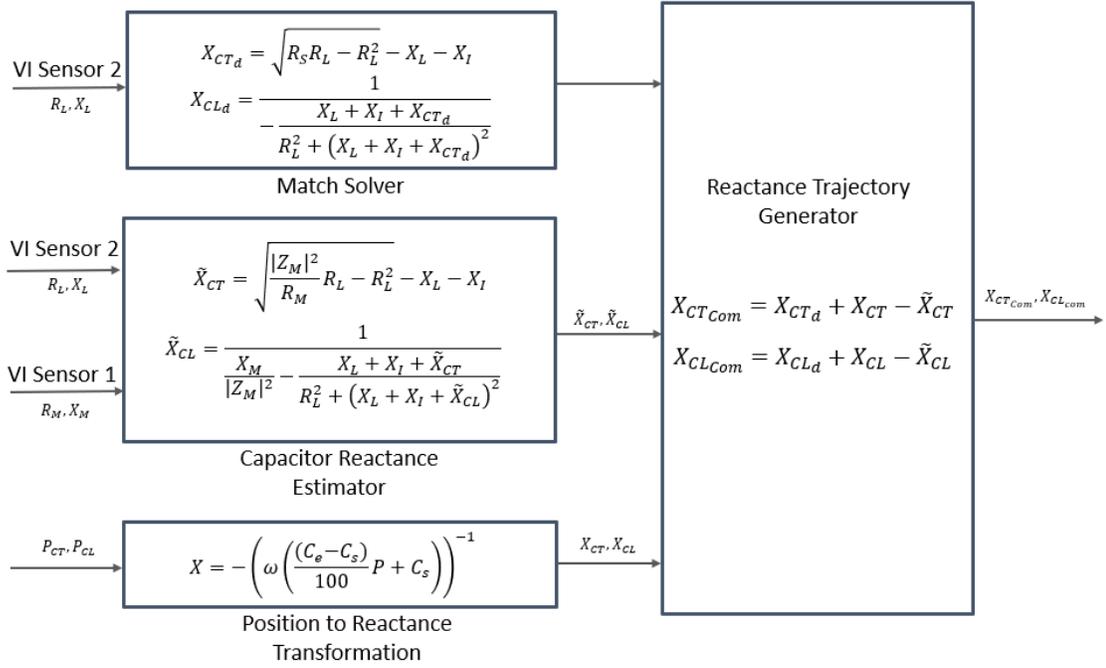


Figure 4.14: In-detail schematic of reactance trajectory generation flow in the Model Reference Algebraic Controller.

After commanded reactances are calculated, the calculation process shown in the flowchart in Fig. 4.8 is followed to generate the voltage command profile to the motion controller to move the system to the matching position.

#### 4.4.2.1 Stability and Robustness

Figures 4.15 and 4.16 show the reflection coefficient based and capacitor position based phase-portrait plots for the Model Reference Algebraic Controller in presence of the model mismatch of the form described in Section 4.3.1, respectively. These figures show that the steady-state error is eliminated after addition of the model reference to the Algebraic Controller; however, the trajectories starting from Tune capacitor positions less than approximately 10% will not converge to the match-point. It can also be seen that all the non-convergent trajectories are starting from points with high percentage of reflected power ( $> 80\%$ ).

Overall, the Model Reference Algebraic Controller shows local robustness in stability and performance for trajectories starting from a large region of capacitor position space; however in presence of model mismatch, certain trajectories starting from low Tune capacitor position, which are also associated with high reflection percentage, will practically become unstable and result in plasma extinction. This

means that although the model reference has improved the robustness substantially, the system is still not globally robust with respect to model uncertainties.

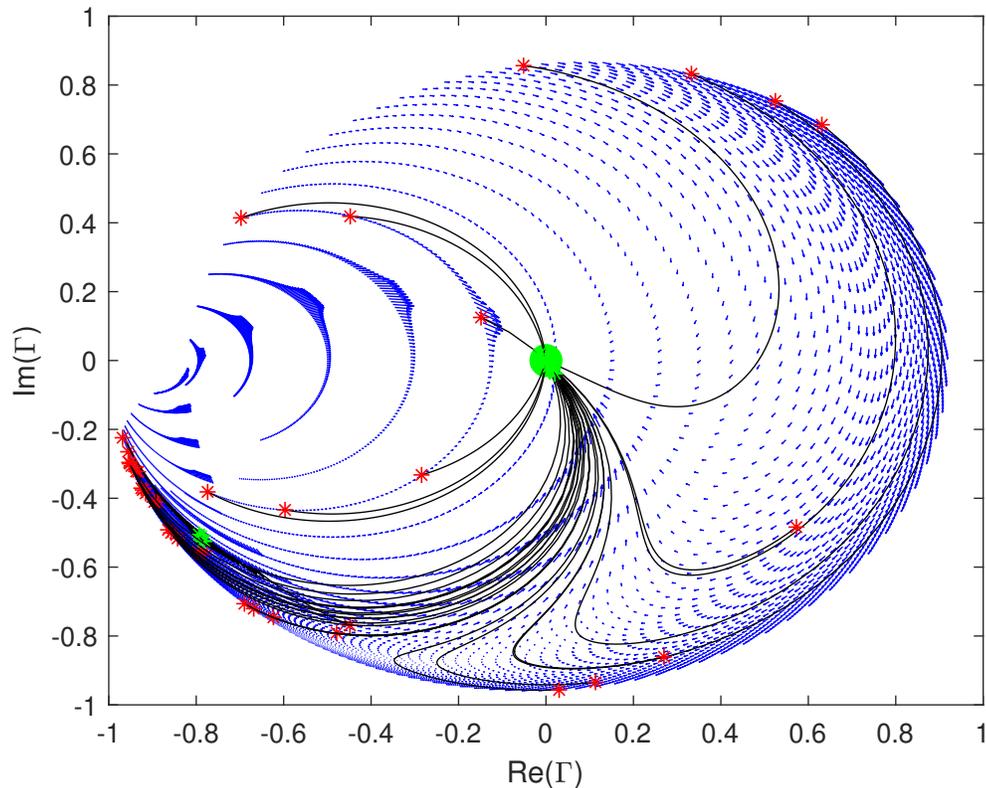


Figure 4.15: Reflection coefficient based phase-portrait plot for the Algebraic Control with model reference for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$  with added 50% reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.

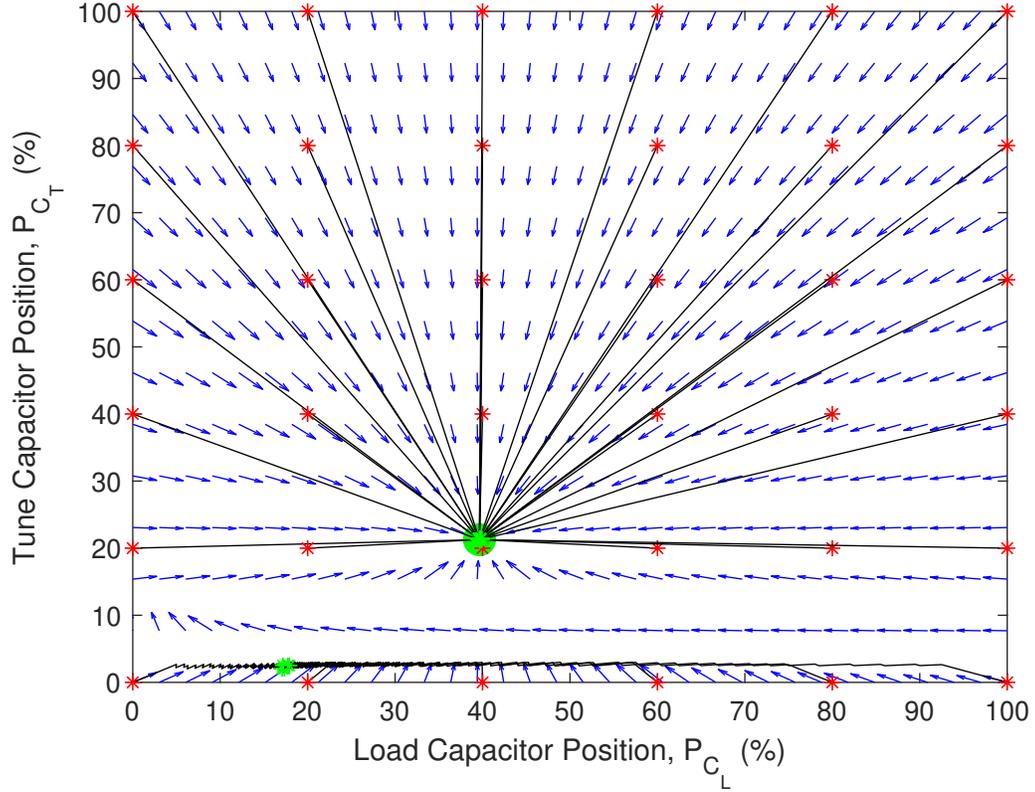


Figure 4.16: Capacitor position based phase-portrait plot for the Algebraic Control with model reference for the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$  with added 50 % reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20 % from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.

#### 4.5 Gradient-Based Controller

The proposed Gradient-Based Controller is based on the idea of following the minimum reflection trajectory as oppose to following the shortest trajectory to the match-point. So, let us consider the following cost function:

$$J = \Gamma^2, \quad (4.17)$$

where the reflection coefficient,  $\Gamma$ , is a function of the equivalent output impedance through Eqn. (4.3). The output impedance at each sample time is a function of the current load impedance, and the current matching network state described by the Load and Tune capacitor positions; so:

$$J = \Gamma^2(R_L, X_L, P_{CL}, P_{CT}). \quad (4.18)$$

Therefore, at each instant, the reflection ratio ( $\Gamma^2$ ), and the direction of descent in the reflection ratio with respect to capacitor positions,

$$\begin{bmatrix} -\frac{\partial J}{\partial P_{CL}} \\ -\frac{\partial J}{\partial P_{CT}} \end{bmatrix},$$

can be calculated numerically; given the load impedance measurement from VI Sensor 2, and Load and Tune position measurement from motion controller. After the direction of descent is known, capacitors can be moved with maximum speed as long as reflection percentage is large enough. After reflection percentage becomes smaller than reflection percentage threshold ( $RP_{th}$ ), then correct gradient step size ( $K_{CL}, K_{CT}$ ) need to be chosen to ensure stable convergence. Figure 4.17 shows the detailed schematic of how the Gradient-Based Controller generates the voltage output profiles for the motion controller.

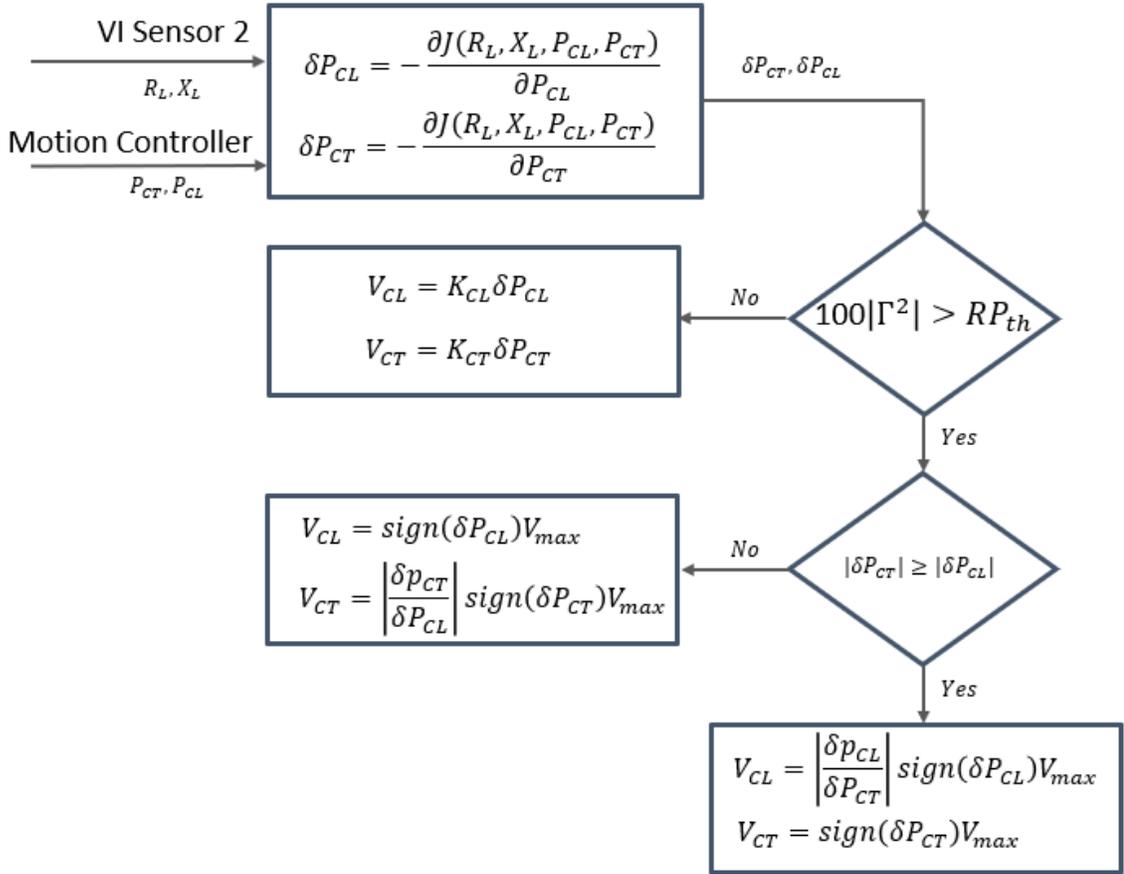


Figure 4.17: Detailed schematic of the Gradient-Based Controller.  $RP_{th}$  is the reflection percentage threshold, and  $K_{CL}$  and  $K_{CT}$  are the gains deciding the gradient descent step size when reflection percentage drops below the threshold.

It is important to note that the cost function (reflection ratio) is always convex and differentiable and its gradient is Lipschitz continuous. This can be numerically shown; some examples of the reflection percentage function over Load and Tune capacitor capacitance space for 4 different load impedances are shown in Fig. 4.2.

#### 4.5.0.1 Stability and Robustness

Considering that the proposed Gradient-Based method depends on the model of the system, and the fact that no model mismatch adjustment mechanism is included in its design, we expect the gradient method not to be robust with respect to model uncertainty. This work only presents the idea of pursuing optimal reflection minimization trajectory towards the match-point, and leaves the further development towards achieving robustness for future projects.

Figures 4.18 and 4.19 show the convergent behavior of the system towards the match-point in absence of model uncertainty, and Figs. 4.20 and 4.21 show similar steady-state error observed with the Algebraic Controller in presence of model uncertainty. Comparing Figs. 4.18 and 4.19 to Figs. 4.9 and 4.10 clearly shows different trajectories towards the match-point between the Gradient-Based approach and the Algebraic Control approach; we will illustrate this difference in the next section using reflection percentage surface over capacitor position space plot.

It is important to note that stability of the Gradient-Based approach is dependent on the choice of step size when system gets closer to the match-point. Considering the non-linear nature of the problem, it might be impossible to find a universal design for the step size. The Gradient-Based approach can be used in a mode switching structure to make it independent of the gain design.

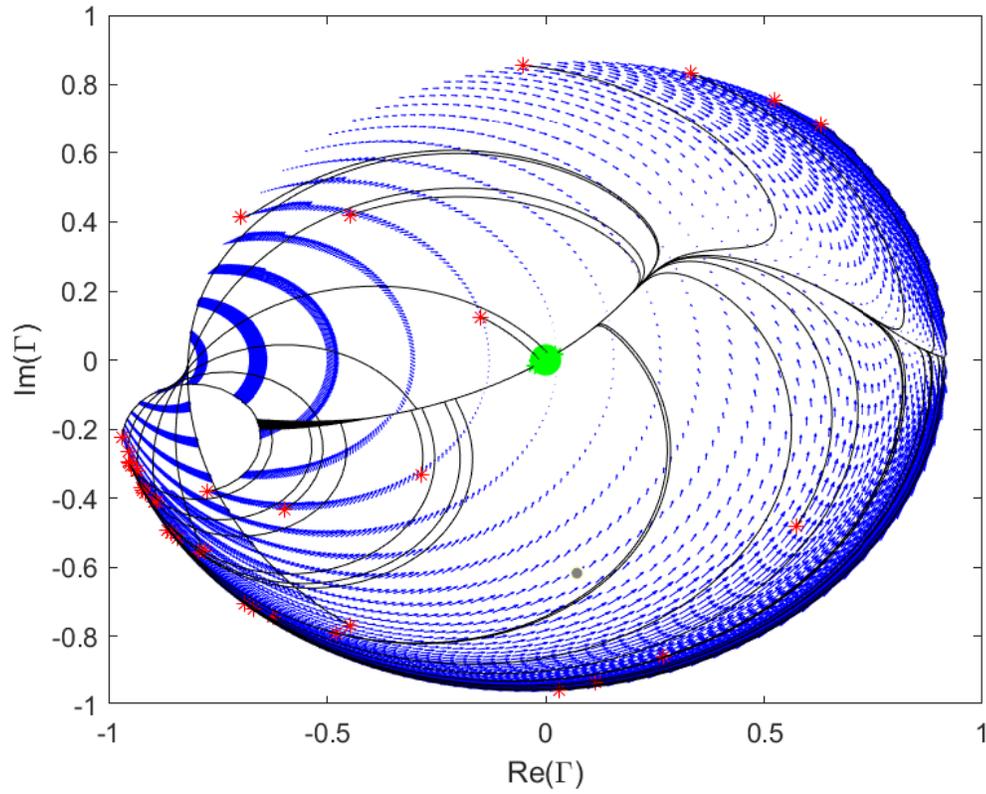


Figure 4.18: Reflection coefficient based phase-portrait plot for Gradient-Based Controller for the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.

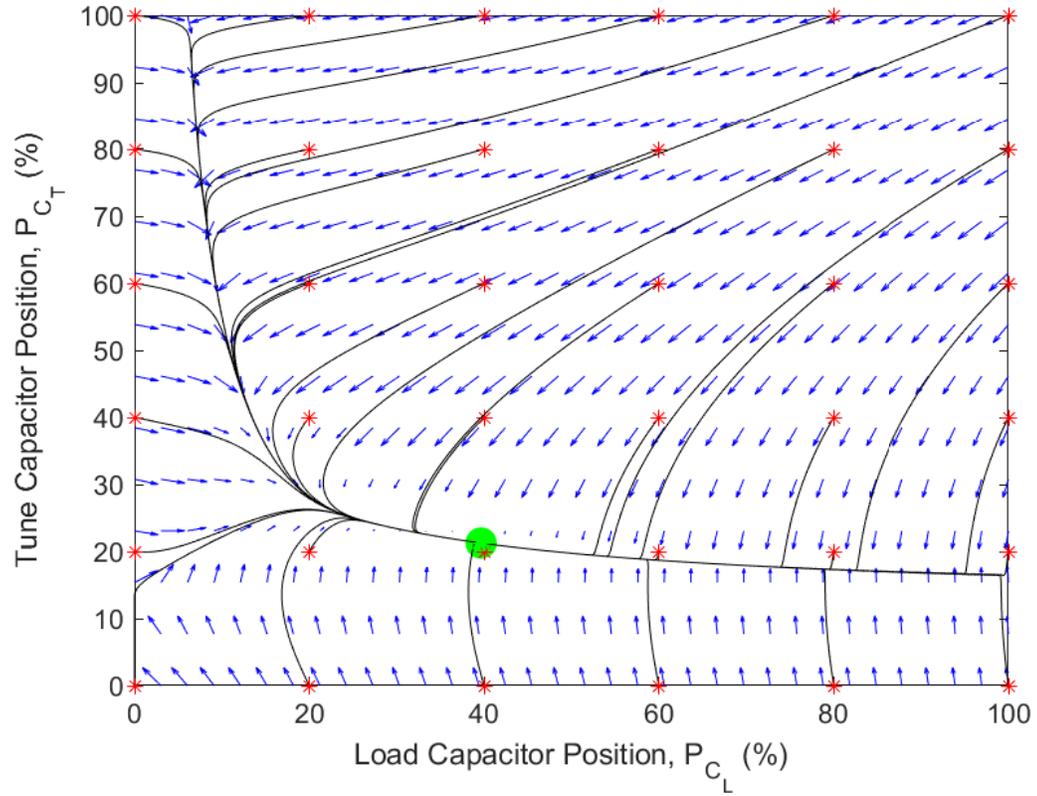


Figure 4.19: Capacitor position based phase-portrait plot for Gradient-Based Controller for the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$ . The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks (overlapped by the green dot) show the end-point of the trajectories.

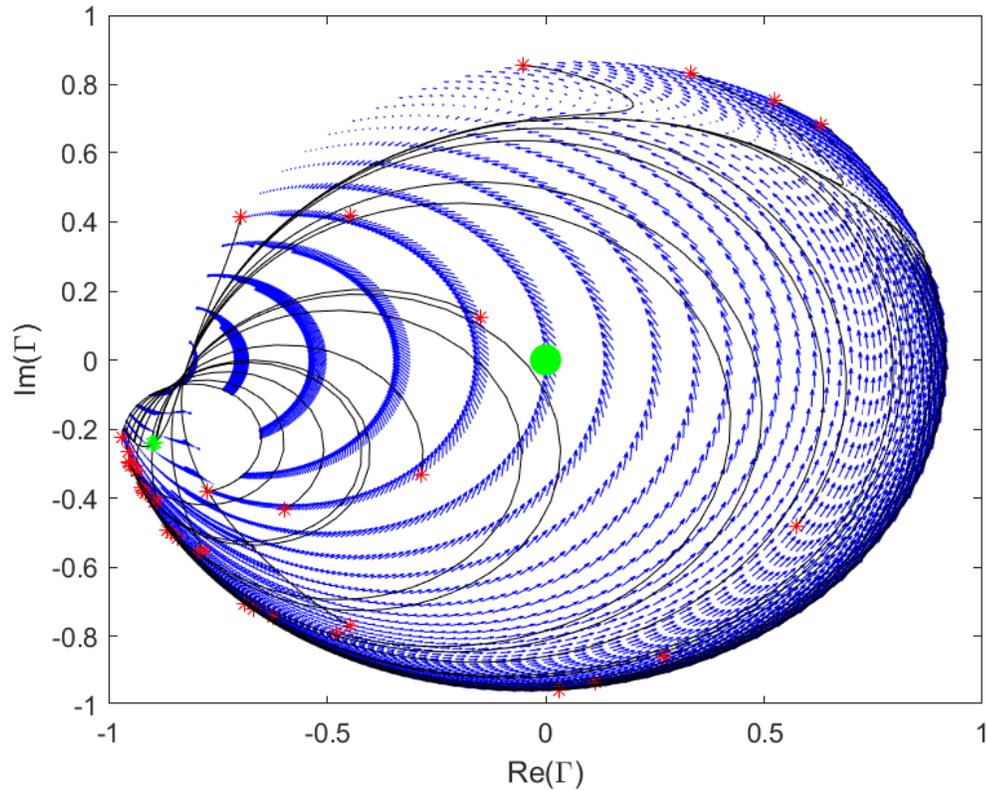


Figure 4.20: Reflection coefficient based phase-portrait plot for Gradient-Based Controller for the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$  with added 50% reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.

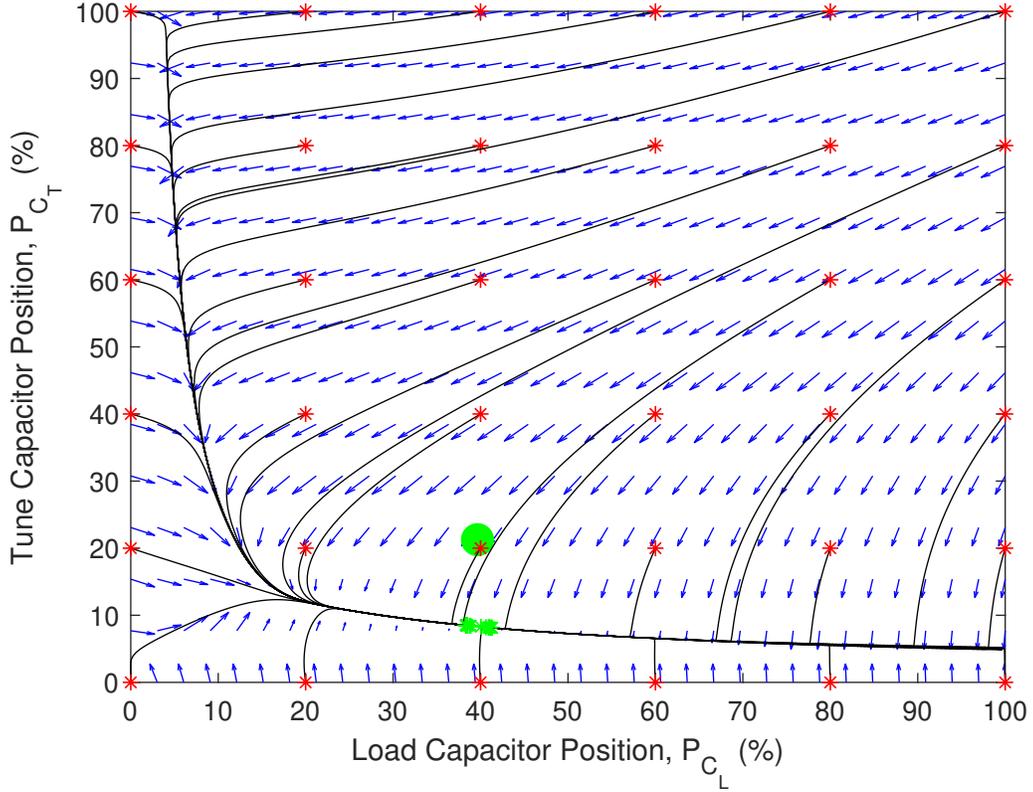


Figure 4.21: Capacitor position based phase-portrait plot for Gradient-Based Controller for the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$  with added 50% reactance model mismatch. The lines show system trajectories starting from a group of points, uniformly distributed in the capacitor position space with distance of 20% from each other. The red asterisks mark the starting point, the big green dot shows the match-point, and the green asterisks show the end-point of the trajectories.

#### 4.6 Further Discussions

As briefly mentioned in previous discussions in stability and robustness sections, from practical perspective, if a trajectory starts from a state with very high reflection percentage, regardless of the control approach, plasma will not stabilize and will extinguish. This is a known fact in semiconductor manufacturing industry that in order to have a successful plasma ignition, system has to be initiated from a state that enough power is instantaneously delivered to the plasma. If all the power is reflected and no power is delivered to the load, the measured load impedance data will be irrelevant for any closed-loop control approach to perform impedance matching. A similar scenario still holds when the power delivered to the load is very small (high reflection). The load impedance has very high variation with respect to

power delivered to it at low powers; considering the very slow dynamic of impedance matching, no closed-loop control approach is able to recover the system from very high reflection percentage state ( $> 70\%$ ). On a side note, typically process is designed in a way that system will not undergo such fast impedance changes that closed-loop controller falls behind in a way that the system ends up in a very high reflection state. So, from practical perspective, convergence of trajectories starting from very high reflection state is not relevant; it is rather important to achieve more optimal performance for trajectories not too far from the match-point. This ensures that closed-loop system will be more capable of following faster impedance changes and will never fall into very high reflection state. These explanations justifies the relevance of the proposed Model Reference Algebraic Control in spite of having non-convergent trajectories from high reflection state in presence of model uncertainty. In the following, we compare the performance of the three discussed controllers, i.e., the decoupled proportional controller, the Model Reference Algebraic Controller, and the Gradient-Based Controller, for a specific trajectory that starts at approximately 50% reflection percentage state.

Figures 4.22 to 4.24, show the forward and reflected power (top plots), Load and Tune capacitor profiles (middle plot), and controller voltage output (bottom plot), for the three controllers at indicated load impedance, and for the specified initial capacitors position. The decoupled proportional control shows the longest time to match. This approach does not consider the coupling in the system, and as a result, in certain states, the behavior of the two capacitors could be conflicting (also refer to Fig. 4.7). We see that system goes through a higher reflection state along the trajectory compared to the state that it started from .

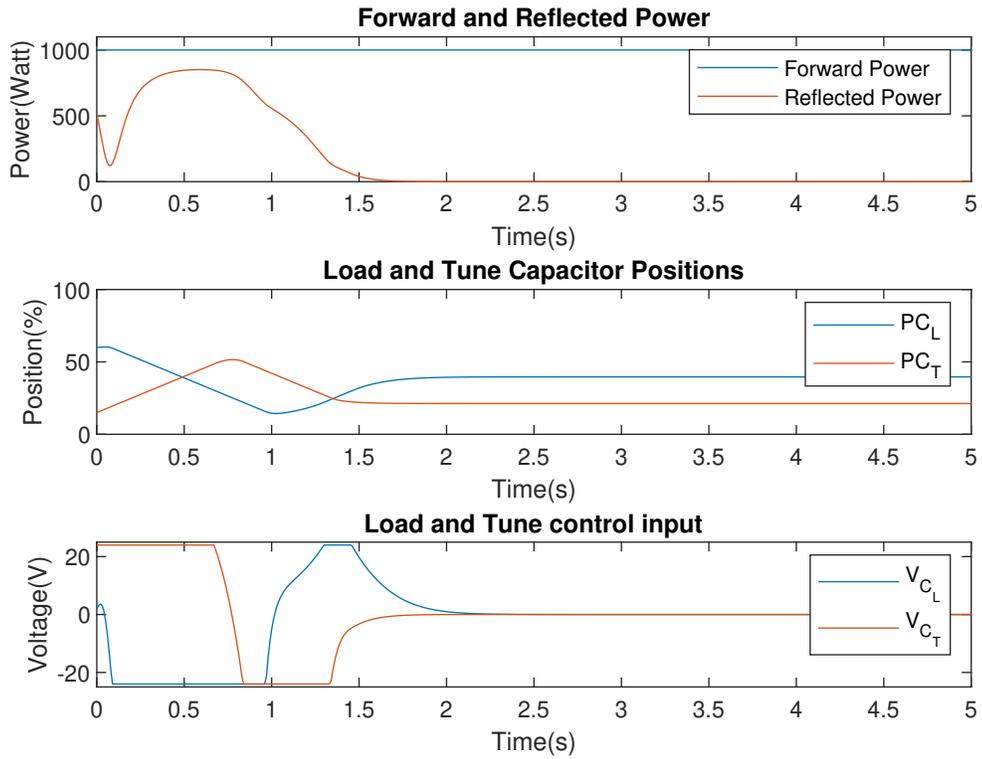


Figure 4.22: Time trace plot for the decoupled proportional controller at the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$ , with capacitor position initial condition of  $PC_L = 60\%$  and  $PC_T = 15\%$ . The top plot shows the simulated reflected and forward power. The middle plot shows the Load and Tune capacitor positions. The bottom plot shows the control action which is the voltage input to the low-level motion controllers for Load and Tune capacitors.

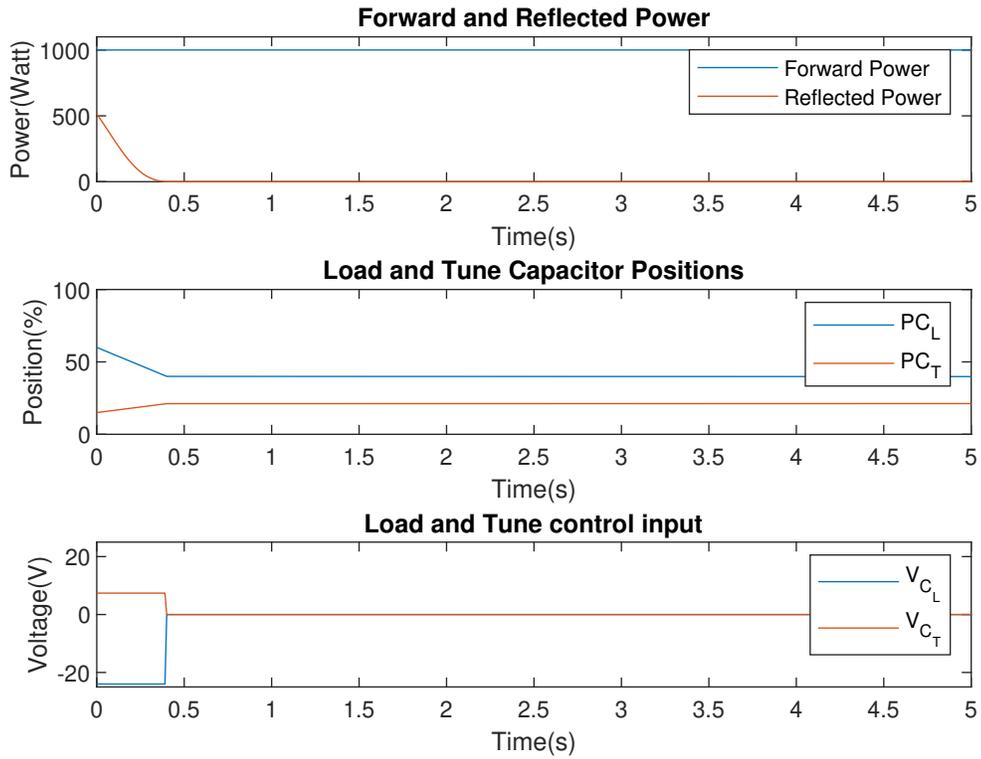


Figure 4.23: Time trace plot for the Model Reference Algebraic Controller at the constant load  $R_L = 5 \Omega$  and  $X_L = 50 \Omega$ , with capacitor position initial condition of  $P_{C_L} = 60\%$  and  $P_{C_T} = 15\%$ . The top plot shows the simulated reflected and forward power. The middle plot shows the Load and Tune capacitor positions. The bottom plot shows the control action which is the voltage input to the low-level motion controllers for Load and Tune capacitors.

The Model Reference Algebraic Controller is the fastest to reach the match-point due to direct motion of capacitors towards the match-point; however, the shortest path does not guarantee the most optimal reflection minimization path. The Gradient-Based controller shows fastest initial rate of reduction in reflected power; however, it takes longer to reach the steady-state match-point due to taking the longer route. Figure 4.25 shows the three controllers trajectories for this example over the reflection percentage surface at two different view angles. It can be clearly observed that the algebraic approach trajectory moves towards the match-point in a straight line in the capacitor position space, while the Gradient-Based approach follows the steepest descent trajectory which is longer, but results in faster initial rate of reflection reduction. The decoupled proportional approach trajectory ignores

the model and controls the capacitors based on the sign of real and imaginary part of reflection coefficient.

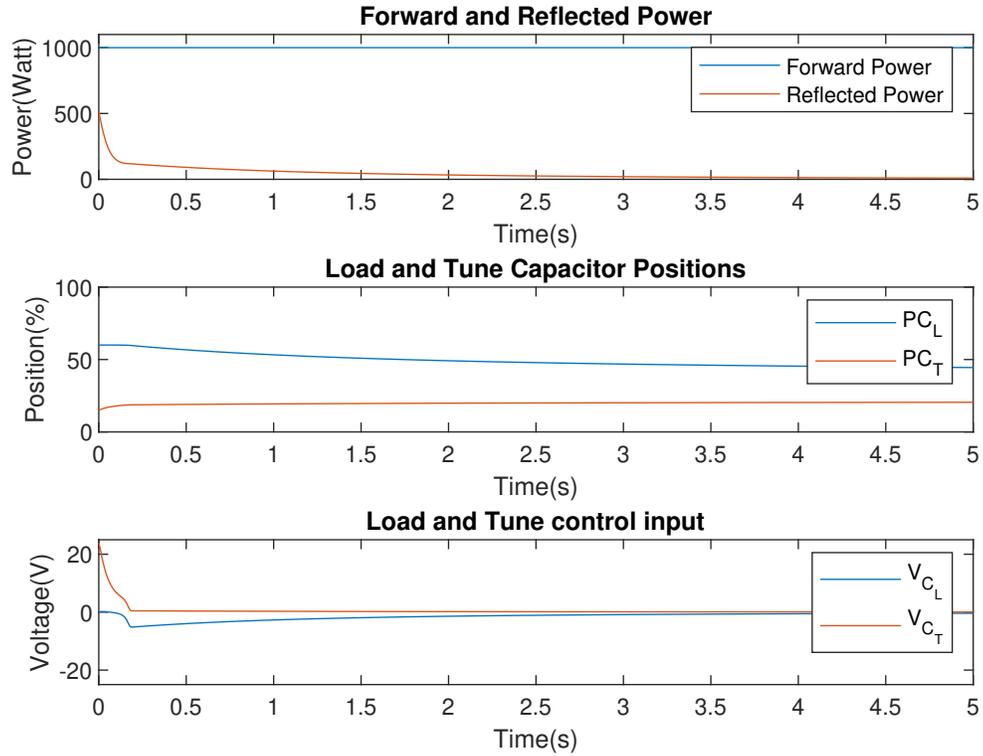


Figure 4.24: Time trace plot for the Gradient-Based controller at the constant load  $R_L = 5\Omega$  and  $X_L = 50\Omega$ , with capacitor position initial condition of  $PC_L = 60\%$  and  $PC_T = 15\%$ . The top plot shows the simulated reflected and forward power. The middle plot shows the Load and Tune capacitor positions. The bottom plot shows the control action which is the voltage input to the low-level motion controllers for Load and Tune capacitors.

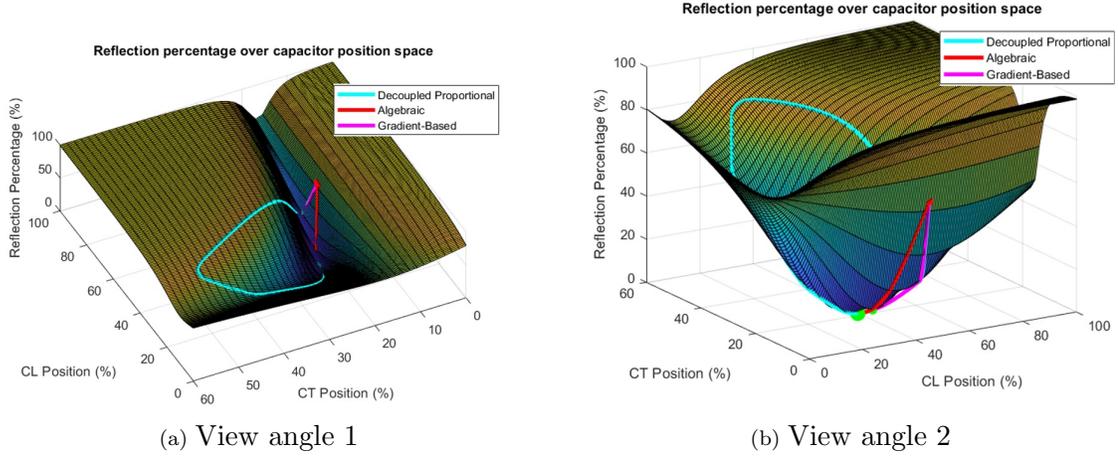


Figure 4.25: Reflection percentage ( $|\Gamma|^2$ ) surface over Load and Tune capacitor position space, for the constant load of  $R_L = 5\ \Omega$  and  $X_L = 50\ \Omega$ , including system trajectory starting from capacitor position initial condition of  $P_{C_L} = 60\%$  and  $P_{C_T} = 15\%$  for the three control approaches. (a) and (b) are two different view angles of the same plot.

It is important to point out that in the existing design of the Gradient-Based Controller, the performance of the system is highly dependent on step size as we get closer to the match-point (refer to Fig. 4.17 for control law and refer to Fig. 4.24 to notice the slow convergence towards the matching-point). One idea to attempt at taking advantage of the Gradient-Based Control approach while avoiding the complications and potential robustness issues that comes with step size design, is to use gradient-based method while the reflection is larger than a certain threshold and then switch to algebraic approach when system reaches below that threshold. As it was mentioned in Section 4.5.0.1, the Gradient-Based Control idea is presented in this work, but its improvement towards a reliable and robust solution is part of a future work.

## 4.7 Conclusion

In this chapter, motivated by the need of semiconductor manufacturing industry for reducing the plasma processing time, the problem of impedance matching was investigated and two advanced control approaches based on a double-sensor impedance matching configuration were proposed. An L-type matching network with a typical target range of load impedance in plasma enhanced processing applications was considered to implement the proposed control approaches. A decoupled proportional control approach based on the conventional single-sensor configuration was used as the baseline for comparison. Stability and robustness of different control techniques were studied through numerical examples with practical applications in mind, using phase-portrait analysis. A model-based control approach based on

obtaining algebraic solution for matching problem is proposed. First, the implementation based on a single-sensor configuration was presented and it was shown that model uncertainties would result in significant steady-state high reflection behavior for all system trajectories. Then, a model reference adjustment strategy using the double-sensor configuration was presented to introduce the Model Reference Algebraic Control method. Model Reference Algebraic Controller showed robust performance with respect to model uncertainties for the range of operation that are relevant in real life applications. Moreover, it showed significant performance improvement over the decoupled proportional control approach in reducing the time to match and reflection minimization. A Gradient-Based Control approach was also presented that is based on achieving minimal reflection trajectory towards the matching-point. The Gradient-Based Controller showed fastest initial rate in reducing reflected power, but longer overall time to match compared to the Model Reference Algebraic Control approach. Gradient-Based Control is not robust with respect to model uncertainties. Incorporating a model reference adjustment mechanism into the Gradient-based approach to achieve robustness is part of future projects.

## Chapter 5

### CONCLUDING REMARKS

#### 5.1 Conclusion

In this dissertation, a self optimizing control (SOC) framework was proposed based on applications in thermal processing and RF impedance matching with the aim to leverage today's availability of measured big data for processes and also the increased computational power to achieve precision and optimal process control. The history of self optimizing control was discussed and some important control strategies that can serve the attainment of the proposed framework were described. Then, some of those strategies were focused on to develop practical control tools, and were used to control several simulation-based and experimental systems.

First, iterative learning control (ILC) was used to control the reactor temperature for a physical carbon activation plant; where the reactor lid was opened up at high temperature for the carbon to be placed in. The designed control approach resulted in achieving uniform temperature for the whole duration of the treatment in spite of the sudden temperature drop due to carbon placement operation.

Then, a model predictive control (MPC) toolbox called RMPC was developed enabling straightforward implementation of MPC for a large class of SISO and MIMO problems. Recursive Integration Optimal Trajectory Solver (RIOTS) which is a Matlab toolbox, was utilized as the optimal control problem (OCP) solver for RMPC. RMPC was employed to solve the control problem introduced in IFAC PID Benchmark Challenge 2018 which led to results that won the challenge. Furthermore, step-response-invariant approximation method was used to enable RMPC to control general MIMO fractional order systems and the implementation for a simulation-based example was demonstrated. Considering that for many process control applications, the future reference (preview) is known, the RMPC objective function was modified to enable it to leverage preview information. RMPC with preview was then employed to control a thermal hardware-in-the-loop system.

Finally, motivated by problems faced in RF impedance matching in semiconductor industry, a conventional proportional control approach was studied to investigate the issues and look for improvements. Then, a double-sensor configuration was proposed to enable implementation of model-based approaches. The Model Reference Algebraic Controller was developed that showed optimal capacitor trajectory towards the match-point. The model reference adjustments completely

eliminated steady-state error in presence of model uncertainties; however, shows lack of robustness for certain trajectories of the system that were typically out of practical operation range. Moreover, a gradient-based controller was developed that allowed for impedance matching with optimal reflection trajectory.

## 5.2 Summary of Dissertation Contributions

The main contribution of this dissertation can be summarized as follows:

- Presenting a self optimizing control framework in the context of process control.
- Experimental implementation of ILC for a physical carbon activation plant to achieve uniform temperature during carbon processing.
- Developing of a plug and play MPC toolbox based on RIOTS OCP solver (RMPC).
- Winning IFAC PID Benchmark Challenge 2018 using RMPC.
- Using step-response-invariant approximation to equip RMPC with capability to control general MIMO fractional order systems.
- Modifying the RMPC toolbox objective function to equip it with the capability of leveraging reference preview information in control.
- Using RMPC with preview to control a thermal hardware-in-the-loop system.
- Proposing a double-sensor configuration for RF impedance matching in semiconductor industries.
- Developing the Model Reference Algebraic Controller to control an L-type matching network in the context of RF impedance matching in semiconductor manufacturing.
- Developing the Gradient-Based Controller to control an L-type matching network in the context of RF impedance matching in semiconductor manufacturing.

## 5.3 Future Work

There are considerable room left for further defining and developing the idea of SOC framework for process control. Specific process control applications where SOC implementation is essential can be identified, specific SOC strategies for those applications can be picked out, and procedures for transformation from conventional approaches towards SOC can be laid out. While this work showed applications of

some control strategies that serve SOC framework, it is possible to showcase a complete implementation of SOC framework including all three characteristics of optimal feedback/feedforward control law, online adjustment of parameters, and high-level data-based learning from operation history.

The developed RMPC toolbox is already a very strong tool for realization of the optimal feedback control in SOC framework. There is a great intellectual and financial opportunity in developing a plug and play embedded RMPC with relay auto-tuning capabilities to be used in the industry. Moreover, RMPC should be applied to more number of experimental or simulation-based examples, such as higher order MIMO systems to further explore the potential and limitations.

As the requirements for impedance matching in semiconductor manufacturing becomes more and more demanding in faster plasma stabilization, shorter time-to-match, and multi-station processing for higher throughput, it becomes more important to squeeze as much performance improvement as possible from impedance matching control systems. So, we believe that the proposed double-sensor configuration should get more attention to achieve robust optimal performance. New adjustment mechanisms could be considered for the proposed Model Reference Algebraic Controllers to improve the robustness. Moreover, the extra sensor could be used to come up with an adaptive scheme for the proposed Gradient-Based Controller.

## BIBLIOGRAPHY

- [1] W. R. Ahrendt, J. F. Taplin, Automatic Feedback Control, McGraw-Hill, 1951.
- [2] P. Daoutidis, M. Zachar, S. S. Jogwar, Sustainability and process control: A survey and perspective, *Journal of Process Control* 44 (2016) 184–206.
- [3] J. Voltz, How does RF matching improve process control for plasma-based processes? (Jul 2020).  
URL <https://renosubsystems.com/blog-how-does-rf-matching-improve-process-control-for-plasma-based-processes/>
- [4] W. K. Roots, J. Woods, On-off control of thermal processes, *IEEE Transactions on Industrial Electronics and Control Instrumentation* (2) (1969) 136–146.
- [5] A. B. Corripio, Tuning of Industrial Control Systems, ISA, 2000.
- [6] P. M. Bellan, Fundamentals of Plasma Physics, Cambridge University Press, 2008.
- [7] G. E. Moore, Cramming more components onto integrated circuits, *Proceedings of the IEEE* 86 (1) (1998) 82–85.
- [8] J. H. Lee, H. K. Park, J. Lee, S. J. Hong, Understanding of RF impedance matching system using VI-Probe, *Journal of the Semiconductor & Display Technology* 19 (3) (2020) 43–48.
- [9] G. Strawn, C. Strawn, Moore’s law at fifty, *IT Professional* 17 (6) (2015) 69–72.
- [10] P. R. Stomer, Adaptive or self-optimizing control systems—a bibliography, *IRE Transactions on Automatic Control* (1) (1959) 65–68.
- [11] S. Skogestad, Self-optimizing control: The missing link between steady-state optimization and control, *Computers & Chemical Engineering* 24 (2-7) (2000) 569–575.
- [12] L. Dong, Y. Li, B. Mu, Y. Xiao, Self-optimizing control of air-source heat pump with multivariable extremum seeking, *Applied Thermal Engineering* 84 (2015) 180–195.

- [13] B. Mu, Self-optimizing control for building ventilation and air conditioning systems, The University of Texas at Dallas, 2015.
- [14] P. Y. H. Li, Self-optimizing control and passive velocity field control of intelligent machines, Ph.D. thesis, University of California, Berkeley (1995).
- [15] S. J. Qin, T. A. Badgwell, An overview of industrial model predictive control technology, in: *AIChE Symposium Series*, Vol. 93, New York, NY: American Institute of Chemical Engineers, 1971-c2002., 1997, pp. 232–256.
- [16] M. Morari, J. H. Lee, Model predictive control: past, present and future, *Computers & Chemical Engineering* 23 (4-5) (1999) 667–682.
- [17] C. R. Cutler, B. L. Ramaker, Dynamic matrix control: A computer control algorithm, in: *Joint Automatic Control Conference*, no. 17, 1980, p. 72.
- [18] W. L. Luyben, *Practical Distillation Control*, Springer Science & Business Media, 2012.
- [19] B. Czakó, J. Sápi, L. Kovács, Model-based optimal control method for cancer treatment using model predictive control and robust fixed point method, in: *2017 IEEE 21st International Conference on Intelligent Engineering Systems (INES)*, IEEE, 2017, pp. 000271–000276.
- [20] A. T. Hafez, A. J. Marasco, S. N. Givigi, M. Iskandarani, S. Yousefi, C. A. Rabbath, Solving multi-UAV dynamic encirclement via model predictive control, *IEEE Transactions on Control Systems Technology* 23 (6) (2015) 2251–2265.
- [21] T. Alvarez, M. Sanzo, C. De Prada, Identification and constrained multivariable predictive control of chemical reactors, in: *Proceedings of International Conference on Control Applications*, IEEE, 1995, pp. 663–664.
- [22] E. F. Camacho, M. Berenguel, Application of generalized predictive control to a solar power plant, in: *The 1994 IEEE Conference on Control Applications. Part 3(of 3)*, Glasgow, UK, 08/24-26/94, 1994, pp. 1657–1662.
- [23] J. Richalet, Industrial applications of model based predictive control, *Automatica* 29 (5) (1993) 1251–1274.
- [24] J. M. Maciejowski, *Predictive Control: With Constraints*, Pearson Education, 2002.
- [25] D. A. Bristow, M. Tharayil, A. G. Alleyne, A survey of iterative learning control, *IEEE Control Systems Magazine* 26 (3) (2006) 96–114.

- [26] D. R. Yang, K. S. Lee, H. J. Ahn, J. H. Lee, Experimental application of a quadratic optimal iterative learning control method for control of wafer temperature uniformity in rapid thermal processing, *IEEE Transactions on Semiconductor Manufacturing* 16 (1) (2003) 36–44.
- [27] D. Gorinevsky, Loop shaping for iterative control of batch processes, *IEEE Control Systems Magazine* 22 (6) (2002) 55–65.
- [28] S. Kawamura, N. Sakagami, Analysis on dynamics of underwater robot manipulators based on iterative learning control and time-scale transformation, in: *Proceedings of the IEEE International Conference on Robotics and Automation* (Cat. No. 02CH37292), Vol. 2, IEEE, 2002, pp. 1088–1094.
- [29] G. Kreisselmeier, B. Anderson, Robust model reference adaptive control, *IEEE Transactions on Automatic Control* 31 (2) (1986) 127–133. doi:10.1109/TAC.1986.1104217.
- [30] P. Jain, M. Nigam, Design of a model reference adaptive controller using modified MIT rule for a second order system, *Advance in Electronic and Electric Engineering* 3 (4) (2013) 477–484.
- [31] A. Shekhar, A. Sharma, Review of model reference adaptive control, in: *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*, 2018, pp. 1–5. doi:10.1109/ICICET.2018.8533713.
- [32] A. Muñoz-Hernández, S. Dehghan, G. Diaz, Physical (steam) activation of post-gasification biochar derived from peach pits, in: *ASME 2018 International Mechanical Engineering Congress and Exposition*, American Society of Mechanical Engineers, 2018, pp. V06AT08A003–V06AT08A003.
- [33] A. Schwartz, E. Polak, Y. Q. Chen, RIOTS Manual: A Matlab toolbox for solving optimal control problems <http://mechatronics.ucmerced.edu/riots> (May 1997).
- [34] A. Schwartz, E. Polak, Y. Q. Chen, Homepage of RIOTS —the most powerful optimal control problem solver <http://www.schwartz-home.com/riots/> (1997).
- [35] C. Tricaud, Y. Chen, Linear and nonlinear model predictive control using a general purpose optimal control problem solver RIOTS\_95, in: *Control and Decision Conference, 2008. CCDC 2008. Chinese, IEEE, 2008*, pp. 1552–1557.
- [36] T. Zhao, Z. Li, Y. Chen, Fractional order nonlinear model predictive control using RIOTS\_95, in: *Fractional Differentiation and Its Applications (ICFDA), 2014 International Conference on, IEEE, 2014*, pp. 1–6.

- [37] Z. Li, Fractional order modeling and control of multi-input-multi-output processes, Ph.D. thesis, UC Merced (2015).
- [38] S. Dehghan, T. Zhao, Y. Zhao, J. Yuan, A. Ates, Y. Chen, PID2018 benchmark challenge: Model predictive control with conditional integral control using a general purpose optimal control problem solver—RIOTS., *IFAC-PapersOnLine* 51 (4) (2018) 882–887.
- [39] S. Dehghan, T. Zhao, Y. Chen, T. Homayouni, Model predictive control of general fractional order systems using a general purpose optimal control problem solver: RIOTS, in: *ASME 2019 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers Digital Collection.
- [40] J. Viola, S. Dehghan, Y. Chen, Embedded RIOTS: Model predictive control towards edge, in: *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 59292, American Society of Mechanical Engineers, 2019, p. V009T12A001.
- [41] D. W. Clarke, C. Mohtadi, P. Tuffs, Generalized predictive control—part I. the basic algorithm, *Automatica* 23 (2) (1987) 137–148.
- [42] P. Lundström, J. Lee, M. Morari, S. Skogestad, Limitations of dynamic matrix control, *Computers & Chemical Engineering* 19 (4) (1995) 409–421.
- [43] B. E. Ydstie, A. Kemna, L. Liu, Multivariable extended-horizon adaptive control, *Computers & Chemical Engineering* 12 (7) (1988) 733–743.
- [44] D. G. Byun, W. H. Kwon, Predictive control: A review and some new stability results, *IFAC Proceedings Volumes* 21 (4) (1988) 81–87.
- [45] D. Luenberger, An introduction to observers, *IEEE Transactions on Automatic Control* 16 (6) (1971) 596–602.
- [46] G. Bejarano, J. A. Alfaya, D. Rodríguez, M. G. Ortega, Benchmark for PID control of refrigeration systems based on vapour compression, <http://www.dia.uned.es/~fmorilla/benchmarkPID2018/>.
- [47] J. Viola, A. Radici, Y. Chen, Comparison of control strategies for the temperature control of a refrigeration system based on vapor compression, arXiv preprint arXiv:1810.06074.
- [48] J. T. Machado, V. Kiryakova, F. Mainardi, Recent history of fractional calculus, *Communications in Nonlinear Science and Numerical Simulation* 16 (3) (2011) 1140–1153.

- [49] J. Tenreiro Machado, M. F. Silva, R. S. Barbosa, I. S. Jesus, C. M. Reis, M. G. Marcos, A. F. Galhano, Some applications of fractional calculus in engineering, *Mathematical Problems in Engineering* 2010.
- [50] T. Djamah, R. Mansouri, S. Djnnoune, M. Bettayeb, Heat transfer modeling and identification using fractional order state space models, *Journal Européen des Systèmes Automatisés* 42 (6-8) (2008) 939–951.
- [51] N. Retiere, M. Ivanes, Modeling of electric machines by implicit derivative half order systems, *IEEE Power Engineering Review* 18 (9) (1998) 62–62.
- [52] J. Sabatier, A. Mohamed, A. Oustaloup, G. Grégoire, F. Ragot, R. Patrick, Estimation of lead acid battery state of charge with a novel fractional model, *IFAC Proceedings Volumes* 39 (11) (2006) 296–301.
- [53] S. Victor, P. Melchior, A. Oustaloup, Robust path tracking using flatness for fractional linear MIMO systems: a thermal application, *Computers & Mathematics with Applications* 59 (5) (2010) 1667–1678.
- [54] H. Sun, A. Charef, Y. Tsao, B. Onaral, Analysis of polarization dynamics by singularity decomposition method, *Annals of Biomedical Engineering* 20 (3) (1992) 321–335.
- [55] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, V. Feliu-Batlle, *Fractional-order systems and controls: fundamentals and applications*, Springer Science & Business Media, 2010.
- [56] A. Oustaloup, From fractality to non-integer derivation through recursivity, a property common to these two concepts: a fundamental idea from a new process control strategy, in: *Proceeding of the 12th IMACS World Congress*, Paris, July, Vol. 3, 1988, p. 203.
- [57] A. Oustaloup, B. Mathieu, P. Lanusse, The CRONE control of resonant plants: application to a flexible transmission, *European Journal of Control* 1 (2) (1995) 113–121.
- [58] I. Podlubny, Fractional-order systems and  $PI^\lambda D^\mu$ - controllers, *IEEE Transactions on Automatic Control* 44 (1) (1999) 208–214.
- [59] M. Romero, C. Mañoso, P. Ángel, B. M. Vinagre, Fractional-order generalized predictive control: Formulation and some properties, in: *2010 11th International Conference on Control Automation Robotics & Vision*, IEEE, 2010, pp. 1495–1500.

- [60] C. Tricaud, Y. Chen, Solving fractional order optimal control problems in RIOTS 95—a general purpose optimal control problem solver, in: Proceedings of the 3rd IFAC Workshop on Fractional Differentiation and its Applications, 2008.
- [61] M. M. Joshi, V. A. Vyawahare, M. D. Patil, Analysis of model predictive control for fractional-order system, in: Simulation and Modeling Methodologies, Technologies and Applications, Springer, 2015, pp. 173–194.
- [62] A. Rhouma, F. Bouani, B. Bouzouita, M. Ksouri, Model predictive control of fractional order systems, *Journal of Computational and Nonlinear Dynamics* 9 (3) (2014) 031011.
- [63] Y. Li, H. Sheng, Y. Chen, Analytical impulse response of a fractional second order filter and its impulse response invariant discretization, *Signal Processing* 91 (3) (2011) 498–507.
- [64] Y. Li, H. Sheng, Y. Q. Chen, On distributed order integrator/differentiator, *Signal Processing* 91 (5) (2011) 1079–1084.
- [65] Z. Jiao, Y. Chen, Impulse response of a generalized fractional second order filter, *Fractional Calculus and Applied Analysis* 15 (1) (2012) 97–116.
- [66] Y. Chen, Impulse response invariant discretization of fractional order low-pass filters (September 2009).  
URL <https://www.mathworks.com/matlabcentral/fileexchange/21365-impulse-response-invariant-discretization-of-fractional-order-low-pass-filters>
- [67] A. Oustaloup, F. Levron, B. Mathieu, F. M. Nanot, Frequency-band complex noninteger differentiator: Characterization and synthesis, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 47 (1) (2000) 25–39.
- [68] J. Trigeassou, T. Poinot, J. Lin, A. Oustaloup, F. Levron, Modeling and identification of a non integer order system, in: Proceedings of European Control Conference (ECC), IEEE, 1999, pp. 2453–2458.
- [69] D. Xue, Y. Chen, Modeling, analysis and design of control systems in MATLAB and Simulink, World Scientific Publishing, 2015.
- [70] Y. Chen, I. Petras, D. Xue, Fractional order control—a tutorial, in: Proceedings of American Control Conference, IEEE, 2009, pp. 1397–1411.

- [71] Z. Li, T. Zhao, Y. Chen, A low cost research platform for modeling and control of multi-input multi-output fractional order dynamic systems, in: Proceedings of International Conference on Fractional Differentiation and Its Applications, IEEE, 2014, pp. 1–6.
- [72] M. Tomizuka, D. E. Rosenthal, On the optimal digital state vector feedback controller with integral and preview actions, *Journal of Dynamic Systems, Measurement and Control* 101 (2) (1979) 172–178. doi:10.1115/1.3426416.
- [73] M. Tomizuka, D. H. Fung, Design of digital feedforward/preview controllers for processes with predetermined feedback controllers, *American Society of Mechanical Engineers (Paper)* 102 (80 -WA/DSC-2) (1980) 218–225.
- [74] M. Tomizuka, On the design of digital tracking controllers, *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME* 115 (2B) (1993) 412–418. doi:10.1115/1.2899081.
- [75] T. Katayamat, T. Itohf, M. Ogawafand, H. Yamamoto, Optimal tracking control of a heat exchanger with change in load condition, in: Proceedings of the 29th Conference on Decision and Control, Honolulu, Hawaii, December, 1990, pp. 1584–1589.
- [76] H. Peng, M. Tomizuka, Optimal preview control for vehicle lateral guidance, *California Partners for Advanced Transit and Highways (PATH)* (1991) 1–28. URL <http://escholarship.org/uc/item/3jj2q67v.pdf>
- [77] J. N. Strohm, F. Christ, Preview  $H_\infty$  control of a hybrid suspension system, *IFAC-PapersOnLine* 52 (5) (2019) 237–242. doi:10.1016/j.ifacol.2019.09.038.
- [78] H. Peng, M. Tomizuka, Preview control for vehicle lateral guidance in highway automation, in: Proceedings of American Control Conference, IEEE, 1991, pp. 3090–3095.
- [79] X. Zeng, J. Wang, A parallel hybrid electric vehicle energy management strategy using stochastic model predictive control with road grade preview, *IEEE Transactions on Control Systems Technology* 23 (6) (2015) 2416–2423. doi:10.1109/TCST.2015.2409235.
- [80] M. Ahmed, F. Svaricek, Preview control of semi-active suspension based on a half-car model using fast fourier transform, in: Proceedings of 10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13), IEEE, 2013, pp. 1–6.

- [81] Y. Lim, H. Tang, S. Lim, J. Park, An adaptive impedance-matching network based on a novel capacitor matrix for wireless power transfer, *IEEE Transactions on Power Electronics* 29 (8) (2013) 4403–4413.
- [82] R. Malmqvist, P. Rantakari, C. Samuelsson, M. Lahti, S. Cheng, J. Saijets, T. Vaha-Heikkila, A. Rydberg, J. Varis, RF MEMS based impedance matching networks for tunable multi-band microwave low noise amplifiers, in: *Proceedings of International Semiconductor Conference*, Vol. 1, Ieee, 2009, pp. 303–306.
- [83] H. M. Nemati, C. Fager, U. Gustavsson, R. Jos, H. Zirath, Design of varactor-based tunable matching networks for dynamic load modulation of high power amplifiers, *IEEE Transactions on Microwave Theory and Techniques* 57 (5) (2009) 1110–1118.
- [84] A. Al Bastami, A. Jurkov, P. Gould, M. Hsing, M. Schmidt, J.-I. Ha, D. J. Perreault, Dynamic matching system for radio-frequency plasma generation, *IEEE Transactions on Power Electronics* 33 (3) (2017) 1940–1951.
- [85] W.-J. Gu, K. Harada, A new method to regulate resonant converters, *IEEE Transactions on Power Electronics* 3 (4) (1988) 430–439.
- [86] A. S. Jurkov, A. Radomski, D. J. Perreault, Tunable impedance matching networks based on phase-switched impedance modulation, in: *Proceedings of Energy Conversion Congress and Exposition (ECCE)*, IEEE, 2017, pp. 947–954.
- [87] G. Bacelli, J. Ringwood, P. Iordanov, Impedance matching controller for an inductively coupled plasma chamber: L-type matching network automatic controller, *ICINCO 2007 - International Conference on Informatics in Control, Automation and Robotics*.
- [88] N. Mazza, Automatic impedance matching system for RF sputtering, *IBM Journal of Research and Development* 14 (2) (1970) 192–193.
- [89] C. J. Cottee, S. R. Duncan, Design of matching circuit controllers for radio-frequency heating, *IEEE Transactions on Control Systems Technology* 11 (1) (2003) 91–100.
- [90] S. Ishida, A. Kawamura, A. Takayanagi, H. Takada, Robust convergence of the impedance matching process based on seek control, in: *Proceedings of IECON 2011-37th Annual Conference of the IEEE Industrial Electronics Society*, IEEE, 2011, pp. 1554–1559.
- [91] M. H. Jones, *A Practical Introduction to Electronic Circuits*, Cambridge University Press, 1995.