2018

Development of Biomimetic-Based Controller Design Methods for Advanced Energy Systems

Gaurav Mirlekar

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Mirlekar, Gaurav, "Development of Biomimetic-Based Controller Design Methods for Advanced Energy Systems" (2018). *Graduate Theses, Dissertations, and Problem Reports*. 6237.
<https://researchrepository.wvu.edu/etd/6237>

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Development of Biomimetic-based Controller Design Methods for Advanced Energy Systems

Gaurav Mirlekar

**Dissertation submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources
at West Virginia University**

In partial fulfillment of the requirements for the degree of

**Doctor of Philosophy
in
Chemical Engineering**

**Fernando V. Lima, Ph.D., Committee Chairperson
Debangsu Bhattacharyya, Ph.D.
David Klinke, Ph.D.
Mario Perhinschi, Ph.D.
Urmila Diwekar, Ph.D.**

Department of Chemical and Biomedical Engineering

**Morgantown, West Virginia
2018**

**Keywords: biomimetic control, adaptive control, artificial neural network,
advanced energy systems, agent-based control and optimization**

Copyright 2018 Gaurav Mirlekar

Abstract

Development of Biomimetic-based Controller Design Methods for Advanced Energy Systems

Gaurav Mirlekar

A biologically inspired optimal control strategy, denoted as BIO-CS, is proposed for advanced energy systems applications. This strategy combines the ant's rule of pursuit idea with multi-agent and optimal control concepts. The BIO-CS algorithm employs gradient-based optimal control solvers for the intermediate problems associated with the leader-follower agents' local interactions. The developed BIO-CS is integrated with an Artificial Neural Network (ANN)-based adaptive component for further improvement of the overall framework. In particular, the ANN component captures the mismatch between the controller and the plant models by using a single-hidden-layer technique with online learning capabilities to augment the baseline BIO-CS control laws. The resulting approach is a unique combination of biomimetic control and data-driven methods that provides optimal solutions for dynamic systems.

The applicability of the proposed framework is illustrated via an Integrated Gasification Combined Cycle (IGCC) process with carbon capture as an advanced energy system example. Specifically, a multivariable control structure associated with a subsystem of the IGCC plant simulation in DYNsIM[®] software platform is addressed. The proposed control laws are derived in MATLAB[®] environment, while the plant models are built in DYNsIM[®], and a previously developed MATLAB[®]-DYNsIM[®] link is employed for implementation purposes. The proposed integrated approach improves the overall performance of the process up to 85% in terms of reducing the output tracking error when compared to stand-alone BIO-CS and Proportional-Integral (PI) controller implementations, resulting in faster setpoint tracking.

Other applications of BIO-CS addressed include: i) a nonlinear fermentation process to produce ethanol; and ii) a transfer function model derived from the cyber-physical fuel cell-gas turbine hybrid power system that is part of the Hybrid Performance (HYPER) project at the National Energy Technology Laboratory (NETL). Other theoretical developments in this work correspond to the integration of the BIO-CS approach with Multi-Agent Optimization (MAO) techniques and casting BIO-CS as a Model Predictive Controller (MPC). These developments are demonstrated by revisiting the fermentation process example. The proposed biologically-inspired approaches provide a promising alternative for advanced control of energy systems of the future.

Dedication

||श्री||

*Dedicated to
Almighty God,
my parents Leena and Vikas,
my sister Devika,
my paternal grand-parents Chandrakant and Mangal,
and maternal grand-parents Vishwanath and Rajani*

Acknowledgments

I would like to thank many individuals who impacted and influenced my life throughout my stay in the USA or back in my homeland India. There are too many to mention here, but I do sincerely thank you all.

First of all, I would like to gratefully acknowledge, my Guru, Dr. Fernando V. Lima who gave me this great opportunity to work with such enthusiastic personality and the Control, Optimization and Design of Energy Systems (CODES) research group. I consider myself fortunate to have a research advisor who is very well organized as well as expert in the field that helped me to shape my PhD thesis in a timely manner. I would like to thank Dr. Lima once again for his mentorship, support, patience and for assigning me to a challenging and innovative biomimetic controller design project that gave me deeper understanding of this area. Secondly, I would like to express my sincere gratitude and appreciation to the Department of Energy (DoE) for their funding support (Grant No.: DE-FE0012451) via project entitled “Development of Integrated Biomimetic Framework with Intelligent Monitoring, Cognition and Decision Capabilities for Control of Advanced Energy Plants” for my PhD research work. I would like to also acknowledge the biomimetic control team of West Virginia University, especially Drs. Bhattacharyya, Perhinschi, Turton, Ghassan Al-Sinbol, Berhane Gebreslassie and Diwekar (at Vishwamitra Research Institute) for the useful discussions in monthly project meetings, insightful comments and guidance, that enriched my overall research experience. I would like to extend this gratitude to Dr. Klinke for his comments on my thesis proposal and also the mathematical methods in ChE course which was helpful for the PhD qualifier. Drs. David Tucker and Paolo Pezzini are also acknowledged for providing the transfer function models derived from the Hybrid Performance system as part of the HYPER project at the National Energy Technology Laboratory (NETL).

I am immensely grateful to West Virginia University and the Department of Chemical and Biomedical Engineering for providing me this platform to pursue my doctoral studies here in Morgantown. It also gave me a chance to explore the "wild and

wonderful" and I take great pleasure in referring it as my second home and myself as a "PROUD MOUNTAINEER".

I sincerely thank all the past and present members of CODES research group for all their mutual trust and friendship through the years, especially Dr. Juan Carlos Carrasco and Shuyun Li. I would like to also thank visiting postdoctoral fellows Drs. David Rincon and Bruno F. Santoro for their help in my different stages of PhD research. I want to give special thanks to Manish, Prathyusha, Naman, Daksha, Samson, Rebecca, Vitor G., Xin, Taiwo, Lei, Brent, Vitor F., Gilson, Selorme, Allyson, Alanis, Tyler, Jacob, Nick, Matt, Joe, Eilish, Yacine, Zach, Nirjhar, Yuan, and many others for building a friendly, interesting and comfortable work environment.

Finally and most importantly, I am blessed to have family that have stayed the course and enriched my life. I must thank my family for always encouraging me and believing in me. My mother, Leena, who always trusted and supported me for achieving my dreams. My father, Vikas, who is always ready with an encouraging and inspiring word. My sister, Devika, who has toughened me up. My maternal and paternal grandparents, gone but never forgotten, who always make me feel like the decisions I make are the right decisions. Last but not the least, my relatives in India for helping me with important details in my life.

Thank you all, thank you very much!

Gaurav Mirlekar
April 26, 2018

Table of Contents

Abstract	ii
Dedication	iii
Table of Contents	vi
List of Figures	x
List of Tables	xiii
1 Introduction	2
1.1 Research Outputs	4
1.2 Thesis Organization	7
2 Literature Review	8
2.1 Process Systems Engineering	8
2.2 Process Control	9
2.3 Advanced Energy Systems Applications and Challenges	10
2.4 Literature Review Summary	12
3 Development of Deterministic Biomimetic Controller Design	13
3.1 Introduction	13
3.2 Approach	14
3.2.1 Optimal Control Background	14
3.2.2 Proposed Framework	14
3.3 Results & Discussions	20
3.3.1 Fermentation Processes Case Study	20
3.3.2 Open-loop Results	23
3.3.3 Closed-loop Results	24
3.4 Conclusions	39

4	Incorporation of Adaptive Component into Biomimetic Controller Design	41
4.1	Introduction	41
4.2	Approach.....	42
4.2.1	Optimal Control Background.....	42
4.2.2	Integrated BIO-CS Framework with Adaptive Component.....	43
4.3	Results & Discussions.....	47
4.3.1	Model Development for BIO-CS Design	47
4.3.2	BIO-CS Stand-alone Implementation	51
4.3.3	BIO-CS + ANN Implementation	52
4.4	Conclusions.....	54
5	Implementation of Biomimetic-based Method in DYNsIM[®] plant	55
5.1	Introduction	55
5.2	Approach.....	55
5.2.1	BIO-CS Implementation Framework	55
5.2.2	Integrated BIO-CS Framework with Adaptive Component.....	58
5.3	Results & Discussions.....	59
5.3.1	Input-Output State-space Analysis	59
5.3.2	BIO-CS Stand-alone Implementation Results.....	61
5.3.3	BIO-CS + ANN Implementation Results	64
5.4	Conclusions.....	67
6	Integration of Controller Design Method with Multi-agent Optimization Framework.....	69
6.1	Introduction	69
6.2	Approach.....	70

6.2.1	Proposed Integrated Framework	70
6.2.2	Multi-agent Optimizer (MAO)	72
6.2.3	BIO-CS Controller.....	74
6.3	Results & Discussions.....	76
6.3.1	Fermentation Process Case Study	76
6.3.2	Multi-agent Optimization Results	79
6.3.3	BIO-CS Closed-loop Results	81
6.4	Conclusions.....	85
7	Other Developments: BIO-CS Implementation on HYPER Process	87
7.1	Introduction	87
7.2	Results & Discussions.....	88
7.2.1	Hybrid Performance (HYPER) Process	88
7.2.2	Open-loop Results.....	92
7.2.3	Closed-loop Results	94
7.3	Conclusions.....	98
8	BIO-CS as a Model Predictive Controller (MPC).....	100
8.1	Introduction	100
8.2	Approach.....	100
8.2.1	Similarities between BIO-CS and MPC	100
8.2.2	Differences between BIO-CS and MPC.....	101
8.2.3	Casting BIO-CS as a Model Predictive Controller (MPC)	102
8.3	Results & Discussions.....	104
8.3.1	Closed-loop Results	104

8.4	Conclusions.....	110
9	Overall Conclusions and Recommendations	112
10	References	114

List of Figures

Figure 2.1 PSE in the information age (adapted from Rawlings, 2017).....	9
Figure 3.1 Schematic representation of ants' rule of pursuit	15
Figure 3.2 BIO-CS closed-loop framework.....	18
Figure 3.3 Schematic representation of fermentation process	21
Figure 3.4 Open-loop simulation results: output concentration profiles for different D_{in} values.....	25
Figure 3.5 BIO-CS closed-loop simulation results (case I): concentration and input profiles for agent0 (left), agent1 (middle), and agent5 (right)	29
Figure 3.6 Closed-loop implementation results (case I): single <i>dynopt</i> (left), PI controller (middle), and BIO-CS (right).....	32
Figure 3.7 BIO-CS closed-loop simulation results (case II): concentration and input profiles for agent0 (left), agent2 (middle), and agent5 (right)	33
Figure 3.8 BIO-CS closed-loop simulation with -20% disturbance in C_{s0} (case III): concentration and input profiles for agent0 (left), agent1 (middle), and agent4 (right) ..	35
Figure 3.9 Closed-loop implementation results with disturbance (case III): single <i>dynopt</i> (left), PI controller (middle), and BIO-CS (right)	36
Figure 3.10 BIO-CS closed-loop simulation with +20% disturbance in C_{s0} (case IV): concentration and input profiles for agent0 (left), agent1 (middle), and agent7 (right) ..	38
Figure 3.11 BIO-CS closed-loop simulation with plant-model mismatch (cases V and VI): concentration and input profiles for agent0 (left); agent18 (k_1 -20%) (middle) and agent17 (m_s -20%) (right)	39
Figure 4.1 Schematic representation of the BIO-CS controller with adaptive component framework (in MATLAB®)	44
Figure 4.2 Architecture of the ANN-based adaptive component	45
Figure 4.3 Summary of the model conversion steps for BIO-CS controller design.....	49
Figure 4.4 Schematic of multiple-input-multiple-output (MIMO) system of IGCC-AGR process (in DYNsim®).....	50

Figure 4.5 Closed-loop BIO-CS simulation results: setpoint tracking – scaled outputs (y_1, y_2) [top], scaled inputs (u_1, u_2) [bottom]	52
Figure 4.6 Schematic representation of the ANN-based adaptive component employed in IGCC-AGR process application	53
Figure 4.7 Closed-loop BIO-CS + ANN simulation results: setpoint tracking – scaled outputs (y_1, y_2) [top], scaled inputs (u_1, u_2) [bottom]	54
Figure 5.1 Schematic representation of the BIO-CS controller framework in MATLAB® for online implementation in DYNsIM® plant simulation	56
Figure 5.2 Schematic representation of the MATLAB® - DYNsIM® Engine Link	57
Figure 5.3 Schematic representation of the BIO-CS controller with adaptive component framework	58
Figure 5.4 2-D representation of input-output spaces for open-loop simulations	60
Figure 5.5 Selected case studies for setpoint tracking: inside, Case I (left), and outside, Case II (right), of model identification region	61
Figure 5.6 Closed-loop BIO-CS and PI simulation results: setpoint tracking Case I – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]	62
Figure 5.7 Closed-loop BIO-CS simulation results: setpoint tracking Case II – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]	63
Figure 5.8 Closed-loop BIO-CS and BIO-CS + ANN simulation results: set-point tracking Case II – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]	65
Figure 5.9 Closed-loop BIO-CS and BIO-CS + ANN simulation results: setpoint tracking with new tuning Case II – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom].....	67
Figure 6.1 Schematic of the overall integrated framework of BIO-CS with multi-agent optimization	71
Figure 6.2 Schematic of the algorithm details for the integrated BIO-CS with MAO framework	74
Figure 6.3 Schematic of the integrated framework with BIO-CS and multi-agent optimization for the fermentation process.....	79

Figure 6.4 BIO-CS simulation for setpoint tracking: (a) output (y_1); (b) output (y_2); (c) input (u_1); and (d) input (u_2) trajectories	82
Figure 6.5 PI simulation for setpoint tracking: (a) output (y_1); (b) output (y_2); (c) input (u_1); and (d) input (u_2) trajectories	84
Figure 6.6 BIO-CS simulation for plant-model mismatch: (a) output (y_1); (b) output (y_2); (c) input (u_1); and (d) input (u_2) trajectories	85
Figure 7.1 Schematic representation of the HYPER process	88
Figure 7.2 HYPER process transfer function models	91
Figure 7.3 Block diagram of BIO-CS implementation with observer	92
Figure 7.4 Open-loop simulation results: effect of step change on electric load - (a) outputs (y_1, y_2), and (b) inputs (u_1, u_2)	93
Figure 7.5 Open-loop simulation results: effect of step change on fuel valve - (a) outputs (y_1, y_2), and (b) inputs (u_1, u_2)	94
Figure 7.6 Closed-loop simulation results: setpoint tracking case - (a) outputs (y_1, y_2), (b) input (u_1), and (c) input (u_2)	96
Figure 7.7 Closed-loop simulation results: disturbance rejection case - (a) outputs (y_1, y_2), (b) input (u_1), (c) input (u_2), and (d) disturbances (u_3, u_4)	98
Figure 8.1 Schematic of agent-based BIO-CS framework	102
Figure 8.2 Schematic of BIO-CS as MPC framework	103
Figure 8.3 BIO-CS closed-loop simulation results for setpoint tracking (case I): output (top) and input (bottom) profiles	106
Figure 8.4 BIO-CS as MPC closed-loop simulation results for setpoint tracking (case I): output (top) and input (bottom) profiles	107
Figure 8.5 BIO-CS closed-loop simulation results for setpoint tracking (case II): output (top) and input (bottom) profiles	109
Figure 8.6 BIO-CS as MPC closed-loop simulation results for setpoint tracking (case II): output (top) and input (bottom) profiles	110

List of Tables

Table 3.1 Base set of parameters used for the fermentation process	23
Table 3.2 BIO-CS case studies: results and parameters.....	28
Table 3.3 Performance comparison among implemented control methods.....	31
Table 4.1 Summary of the notations used for describing process and ANN variables ..	46
Table 5.1 Selected case studies for implementation purposes	59
Table 5.2 BIO-CS and ANN parameters used for implementation purposes.....	64
Table 5.3 Summary of Integrated Squared Error (ISE) values for controller implementations in different case studies.....	66
Table 6.1 Base case set of constants and parameters used for the fermentation process model	78
Table 6.2 MAO implementation results	81
Table 7.1 Base set of process variables and BIO-CS parameters	90
Table 8.1 BIO-CS and BIO-CS as MPC parameters selected for implementation purposes	105
Table 8.2 Summary of comparison between controller implementations for Case I....	106
Table 8.3 Summary of comparison between controller implementations for Case II...	108

Development of Biomimetic-based Controller Design Methods for Advanced Energy Systems

1 Introduction

Between 2015 and 2040, world energy consumption is expected to increase by 28% as reported by the U.S. Energy Information Administration (EIA), with more than half of the increase attributed to U.S. and Asia (including China and India), where strong economic growth drives the increasing demand for energy [1]. The utilization of fossil fuels such as coal is likely to continue so that meeting such world's energy demand is possible. With such growing energy demand in the world, generating clean forms of energy is a challenging task. To address this challenge, the design of advanced control approaches for energy systems is a necessity of the time. The developed methods must account for strong interactions among multiple process variables and plant-model mismatches associated with highly nonlinear models that characterize advanced energy systems. Over the last decade, biologically-inspired strategies for advanced control have been proposed for tackling nonlinear systems of different nature. For example, inspired by the ant's rule of pursuit idea, a Generalized Sampled Local Pursuit (GSLP) algorithm was proposed to address problems such as the optimal control of a container crane system, minimum time control of a bridge crane, and collective movement of robots [2]. In addition, influenced by the swarm intelligence, a stigmergic algorithm has been developed to solve inverse thermal problems using a distributed approach for constructing solutions [3]. These techniques have emerged as attractive alternatives for advanced control applications that could be leveraged for chemical process control.

In particular, the behavior of the natural groups such as ants, bees and swarms demonstrates that self-organization and cooperation by following simple rules of interaction can result in a wide range of optimal patterns [2, 4]. This idea from biological systems is used here as motivation for the development of advanced optimal control strategies for chemical process control with particular inspiration from the ant's rule of pursuit. In this rule, suppose that an ant finds food by walking around at random. This pioneer ant would then trace a wiggly path back to the nest and start "group recruitment". The subsequent ants (or agents) would one after the other straighten the trail a little

starting from the original path until the agents' paths converge to a line connecting the nest and the food source, despite the individual ant's lack of sense of geometry [5]. Thus, by cooperating in large numbers, ants accomplish tasks that would be difficult to achieve individually. Inspired by this rule of pursuit idea, the objective in this work is to design and implement biomimetic optimal control strategies for complex chemical processes. The main focus of implementation is the Acid Gas Removal section of a coal-based Integrated Gasification Combined Cycle (IGCC-AGR) plant with CO₂ capture. Other implementation results also demonstrate the potential of the developed strategy to address a nonlinear fermentation process model and a transfer function model derived from the cyber-physical fuel cell-gas turbine Hybrid Performance (HYPER) system at NETL. The specific aims of this research are summarized as follows:

- 1) Develop a deterministic biologically-inspired optimal control strategy (using a gradient-based optimal control solver, *dynopt*, in MATLAB® for the intermediate problems).
- 2) Incorporate adaptive components into the developed controller architecture.
- 3) Implement the biomimetic-based approach for the IGCC-AGR process.
- 4) Integrate controller design method with Multi-Agent Optimization (MAO) framework (employing techniques such as the Efficient Ant Colony Optimization, EACO, among others).

By addressing these aims, the specific contributions of this dissertation are: (i) design of a Biologically-Inspired Optimal Control Strategy (BIO-CS) to address a nonlinear fermentation process model for setpoint tracking, disturbance rejection and plant-model mismatch; (ii) formulation of a novel biologically-inspired, advanced control approach with adaptive capabilities; (iii) implementation of the proposed framework for the IGCC-AGR process simulation in DYNOSIM® software [6] to tackle setpoint tracking and plant-model mismatch cases; and (iv) application of the BIO-CS on a transfer function model derived from the HYPER system at NETL to tackle coupling effects among different energy systems components. Other theoretical developments in this thesis include: (i) integration

of the proposed BIO-CS with Multi-Agent Optimization (MAO) techniques as an alternative for the combination of Real-Time Optimization (RTO) with Model Predictive Control (MPC); and (ii) formulation of BIO-CS as MPC for demonstrating the similarities and differences between agent-based and model-based control methods.

1.1 Research Outputs

The contributions of this thesis have resulted in the following products:

A. Journal Publications

1. Mirlekar, G., Gebreslassie, B., H., Li, S., Diwekar, U., M., and Lima, F., V., 2018, Biomimetic agent-based optimization and advanced control framework for nonlinear chemical processes. *In preparation for publication*.
2. Mirlekar, G., Al-Sinbol, G., Perhinschi, M., G., and Lima, F., V., 2018. A biologically-inspired approach for adaptive control of advanced energy systems. Submitted for publication in *Computers & Chemical Engineering*.
3. Mirlekar, G., Li, S., and Lima, F., V., 2017. Design and implementation of a Biologically-Inspired Optimal Control Strategy (BIO-CS) for chemical process control. *Industrial & Engineering Chemistry Research*, 56(22), p. 6468–6479.
4. Li, S., Mirlekar, G., Ruiz-Mercado, G., J., and Lima, F., V., 2016. Development of chemical process design and control for sustainability. *Processes*, 4(3), 23; doi:10.3390/pr4030023.
5. Perhinschi, M., G., Al-Sinbol, G., Bhattacharyya, D., Lima, F., V., Mirlekar, G., and Turton, R., 2015. Development of an immunity-based framework for power plant monitoring and control. *Advanced Chemical Engineering Research*, 4(1), p.15-28.

B. Book Chapter

1. Lima, F., V., Li, S., Mirlekar, G., Sridhar, L., N., and Ruiz-Mercado, G., J., 2016. Modeling and advanced control for sustainable process systems. In *Sustainability in the Analysis, Synthesis and Design of Chemical Engineering Processes*, G. Ruiz-Mercado and H. Cabezas (eds.), Elsevier.

C. Conference Proceedings

1. Mirlekar, G., Gebreslassie, B., H., Li, S., Diwekar, U., M., and Lima, F., V., 2018, July. An integrated biomimetic control strategy with multi-agent optimization for nonlinear chemical processes. Accepted in the *Proceedings of the IFAC ADCHEM Symposium*.
2. Mirlekar, G., Pezzini, P., Bryden, M., Tucker, D., and Lima, F., V., 2017, May. A biologically-inspired optimal control strategy for hybrid energy systems. In *American Control Conference (ACC)*, IEEE.

D. Conference Presentations

1. Mirlekar, G., Al-Sinbol, G., Perhinschi, M., G., and Lima, F., V., "Development of a biologically-inspired approach for advanced adaptive control of clean energy systems". In AIChE Annual Meeting (Computing and Systems Technology Division), Minneapolis, MN, 2017.
2. Mirlekar, G., Pezzini, P., Bryden, K., M., Tucker, D. and Lima, F., V., "A biologically-inspired optimal control framework: application to the Hybrid Performance (Hyper) system." In AIChE Annual Meeting (Computing and Systems Technology Division Poster Presentation), Minneapolis, MN, 2017.
3. Bankole, T., Mirlekar, G., Al-Sinbol, G., Gebreslassie, B., Lima, F., V., Perhinschi, M., Diwekar, M., Turton, R., and Bhattacharyya, D., "Development of biomimetic approaches for intelligent control system design, monitoring and optimization of advanced energy systems." In AIChE Annual Meeting (Computing and Systems Technology Division Poster Presentation), Minneapolis, MN, 2017.
4. Mirlekar, G., and Lima, F., V., "Design and implementation of a Biologically-Inspired Optimal Control Strategy (BIO-CS) for advanced energy systems". In AIChE Annual Meeting (Computing and Systems Technology Division), San Francisco, CA, 2016.
5. Gebreslassie, B., H., Mirlekar, G., Lima, F., V., and Diwekar, U., M., "Optimal control design based on efficient ant colony algorithm, case study: chemical process control". In AIChE Annual Meeting (Computing and Systems Technology Division), San Francisco, CA, 2016.

6. Gebreslassie, B., H., Diwekar, U., M., Mirlekar, G., and Lima, F., V., "Optimal control based on Efficient Ant Colony Optimal Control (EACOC) algorithm: case study chemical process control". In AIChE Midwest Regional Conference, Chicago, IL, 2016.
7. Mirlekar, G., Gebreslassie, B., H., Diwekar, U., M., and Lima, F., V., "Design and implementation of a biomimetic control strategy for chemical processes based on efficient ant colony optimization". In AIChE Annual Meeting (Computing and Systems Technology Division), Salt Lake City, UT, 2015.
8. Lima, F., V., Li, S., Mirlekar, G., Sridhar, L., N., and Ruiz-Mercado, G., J., "Modeling and advanced control for sustainable process systems". In AIChE Annual Meeting, Salt Lake City, UT, 2015.

E. Computational Software Tools

1. Computational Tool #1: Computational Modules in MATLAB® for Biomimetic Control Algorithms

The purpose of this computational tool is to control multiple outputs of an identified application model in MATLAB® using the BIO-CS algorithm. The design of the BIO-CS algorithm employs the DYNAMIC OPTimization (*dynopt*) solver and incorporates the Artificial Neural Network (ANN)-based adaptive component into the framework. The developed algorithm is also implemented on a DYNsIM® [6] simulation of the IGCC-AGR process employing the MATLAB®-DYNsIM® link.

2. Computational Tool #2: Biomimetic Controller Design Toolbox in MATLAB®

The specific goals of the developed biomimetic controller toolbox are: 1) to find optimal setpoints for the given process considering the overall process objective employing the multi-agent optimizer; 2) to take the process to those desired setpoints optimally by implementing the BIO-CS controller. The applicability of the toolbox is illustrated via the implementation on a MIMO control structure of the nonlinear, high-dimensional fermentation process in MATLAB®.

The developed codes and computational tools mentioned above for the algorithm design are user friendly and were submitted to DoE as deliverables for the DoE funded project.

1.2 Thesis Organization

The outline for the remaining chapters of this thesis includes a generic literature review, which is presented first. Then, the design of the deterministic Biologically-Inspired Optimal Control Strategy (BIO-CS) followed with its implementation on a fermentation process case study. The incorporation of the ANN-based adaptive component into the developed BIO-CS algorithm is presented in Chapter 4. For the application of the proposed framework in MATLAB®, a subsystem associated with an Acid gas Removal (AGR) section of an Integrated Gasification Combined Cycle (IGCC) plant simulation is employed. The implementation of the developed approach is also demonstrated using different software platforms (MATLAB® for controller design and DYNsIM® for plant simulation) in Chapter 5. Next, a novel design is proposed in Chapter 6 which combines multi-agent-based optimization techniques with the BIO-CS framework for process control and optimization. The integrated method is then implemented for the multivariable control structure associated with the fermentation process. Other developments in this work are presented in Chapter 7 by showing the application of the BIO-CS controller to the Hyper Performance (HYPER) case study. Finally, the design of the proposed BIO-CS as a Model Predictive Controller (MPC) is discussed in Chapter 8 by revisiting the fermentation process example. This chapter also summarizes both similarities and differences between agent-based and model-based controllers in general, followed by the overall conclusions of the thesis and recommendations for future work in Chapter 9.

2 Literature Review

This chapter presents a big picture overview of the reviewed literature to perform this research, following the main topics: Process Systems Engineering (PSE), process control, and advanced energy systems applications and challenges. Specific topics will be further discussed in the literature review in their respective chapters.

2.1 Process Systems Engineering

PSE is the field that encompasses the activities required in the engineering of systems involving physical, chemical, and/or biological process operations [7]. PSE has been evolving into a specialized field at the interface between chemical engineering, applied mathematics and computer science. In this field, advanced methods and tools are being developed to deal with the complex and the multi-objective nature of decision-making during the operation of chemical processes [8]. The following are some of the key topics that emerged as major challenges in the PSE area over the last decades [9]: (i) process and product design; (ii) process control; (iii) process operations; (iv) process modeling; (v) process integration; and (vi) process optimization and simulation tools. In summary, PSE complements many subareas of chemical engineering and provides a platform for the optimal development of chemical engineering processes in the 21st century [7]. The present work aims at making a contribution to the field of PSE, especially in the areas associated with advanced process control of energy systems. The proposed work is aligned to the 2040 vision of process systems engineering that was presented recently at the Symposium for the Retirement of George Stephanopoulos at MIT [10] (see Figure 2.1). In the next section, a brief literature review on process control is given.

Figure 2.1 PSE in the information age (adapted from Rawlings, 2017)

2.2 Process Control

Process control emerged in the chemical industry from the developments of the servo-regulatory theory and practice in other engineering domains such as mechanical engineering and electrical power engineering. Specifically, the servo-regulator or proportional-integral-derivative (PID) controller is the basic technique that can serve the purpose for control as long as the processing units are simple and interactions among them are mild [7]. However, with increasing process complexity (e.g., material recycle, energy integration), strong interaction among control loops and operational constraints, serious issues may arise when using such controllers. To address these challenges, the following topics need to be explored: (i) use of reliable process models for making critical decisions on the control structure; (ii) introduction of digital computers and technologies; (iii) applications of Model Predictive Control (MPC); and (iv) plant-wide control of industrial chemical processes and the associated academic research. In recent years, biologically-inspired methods have been studied as a promising alternative for advanced process control techniques. These methods have potential to address some of these challenges

associated with advanced chemical/energy systems. In the literature, optimal control has been researched extensively as a process control method for various systems applications. In particular, optimal control approaches deal with finding a control law for a given process such that a certain optimality criterion is achieved. Typical approaches include dynamic programming techniques by employing calculus of variations, the Pontryagin's maximum principle (a necessary condition), or by solving the Hamilton-Jacobi-Bellman equation (a sufficient condition) [11, 12]. Nonlinear programming (NLP) techniques can also be used to solve such problems provided the system of differential equations is converted to nonlinear algebraic equations [13]. In addition, one of the recent approaches include stochastic maximum principle for optimal control under uncertainty [14]. Such technique is implemented for the treatment of a pathogenic disease [15]. Time dependent uncertainties in optimal control problems are also addressed by employing a coupled maximum principle-nonlinear programming numerical optimization algorithm [16]. The optimal control in batch distillation [17] and the implementation of deterministic as well as stochastic optimal control methods for biodiesel production in a batch reactor are studied in detail in the past [18, 19]. However, optimal control concepts have not been explored in combination with biologically inspired techniques in the studied literature. In the following section, a brief overview on the advanced energy systems applications considered here for demonstration purposes, along with their control challenges are discussed.

2.3 Advanced Energy Systems Applications and Challenges

The first advanced energy system addressed in this work is the nonlinear fermentation process to produce ethanol with the desired level of performance and stability at steady state. In particular, ethanol is one of the most promising alternative fuels, either as fuel itself or for blending with gasoline, for taking the United States one step closer to energy independence and sustainability. The fermentation process dynamics are represented by a nonlinear first-principles model with constraints. A fermentation process model from the literature has shown high yields for ethanol production, but with the existence of oscillatory behavior caused by the Hopf bifurcations [20, 21]. The control problems such

as setpoint tracking of product concentration, disturbance rejection related to the inlet substrate concentration and plant-model mismatches associated with parametric uncertainties are critical for the successful operation of this fermentation process.

Secondly, one of the main challenges in chemical process control consists of designing controllers for complex nonlinear processes, such as the acid gas removal (AGR) section of the Integrated Gasification Combined Cycle (IGCC) process. The IGCC-AGR uses the physical solvent Selexol to selectively remove CO₂ and H₂S from the shifted syngas. This process is particularly important since the IGCC plants with CO₂ capture should be operated optimally without violating operational and environmental constraints [22]. In the literature, very few studies can be found on a systematic approach for the controller design for this process. Some of the most recent published papers demonstrate the selection of primary and secondary controlled variables for an IGCC- AGR plant focusing on the control structure design for such plants [23, 24]. Optimization studies have also been conducted for steady-state simulation of an IGCC plant [22]. The literature on the application of advanced controllers to IGCC processes is also scarce. In addition, the representative plant considered in most of the advanced control studies is comprised of simplified models due to computational time restrictions [25]. Hence, the current status of the process control field explicitly shows the need of computationally efficient optimal controller design methods for such complex commercial-scaled chemical/energy plants.

Thirdly, the hybrid energy system discussed in this work requires advanced control techniques to address their unique challenges associated with strong interactions among different energy system components, without violating their operational constraints [26]. Specifically, transient coupling among diverse energy devices is present when such devices are combined into the same system. These challenges provide opportunities for the design of novel control strategies for hybrid power systems [27, 28]. The hybrid energy system considered here for application purposes is derived from the Hybrid Performance (HYPER) project at the National Energy Technology Laboratory (NETL) in Morgantown (WV). The HYPER project is comprised of one-of-the-kind facilities in the U.S. that were

designed for the evaluation of future hybrid systems, including the development of modeling and control approaches focused on a gas turbine-fuel cell hybrid system. In the analyzed system configuration, the dynamic coupling of different energy components corresponds to the main control challenge that needs to be addressed. Previous studies were conducted to tackle different control problems encountered in the HYPER process such as failure mitigation in the emergency shut-down procedure and multi-input multi-output coupling [29, 30, 31]. These studies were mainly focused on classical feed-forward control approaches and the characterization of the transient dynamics of the process. However, research studies on the control of the system in operation for setpoint tracking and disturbance rejection scenarios are scarce. The application of advanced control methods to address coupling effects without compromising the system performance is critical for the future of the HYPER project.

2.4 Literature Review Summary

On the basis of the literature review performed above and the motivation from the advanced energy systems applications, the following main control challenges and gaps were identified: (a) nonlinearities and multiple steady states present in process models; (b) development of fast and accurate reduced models for use in model-based control of highly complex processes; (c) solving equation systems of various sizes considering a multiple-software automation environment [32]; (d) adaptation to plant-model mismatch scenarios; (e) hybrid nature of processes resulting in strong interactions among variables; (f) challenging operational constraints; and (g) multi-variable nature of control problems. These challenges have been addressed by the developments detailed in the chapters that follow.

3 Development of Deterministic Biomimetic Controller Design

3.1 Introduction

This chapter is focused on the development of an advanced biomimetic strategy for optimal controller design of chemical and energy processes. Biologically-inspired strategies or biomimetics are the human-made designs that imitate nature. Many biological systems have been a source of inspiration for advanced control methods because of the success of these systems in solving difficult problems encountered in nature. In particular, the ant's rule of pursuit is an excellent example of how biological systems can efficiently solve problems encountered in nature by cooperative behavior. Inspired by such biological events, this work introduces a novel optimal control strategy for nonlinear chemical processes with constraints. The obtained algorithm is implemented to address problems in the field of chemical process control for the first time.

In the past, ideas from biological entities have motivated many researchers for the development of advanced optimal control strategies to overcome various engineering challenges. Specifically, studies have been conducted to develop the optimal control method entitled generalized sampled local pursuit (GSLP), which is inspired by the ant's rule of pursuit, to tackle problems in the field of mechanical engineering. For this purpose, researchers have used RIOTS as their solver in MATLAB to address the intermediate optimal control problems [2, 4]. Another example included the inspiration from the *Escherichia coli* bacterium that has a control system that enables the microorganisms to search for food and try to avoid noxious substances. This approach was used to control the height of the liquid in a tank [33]. Also, a chemotactic algorithm inspired by bacterial cell behavior was applied to the motor control of a mobile robot [34]. Even though the design of biologically-inspired control algorithms has been a field of active research in the past two decades, most of this research was focused on problems in fields other than chemical engineering. The optimal control of complex chemical processes has unique challenges associated with the nature of these processes, such as nonlinearities present in chemical process models, operational constraints and minimum time set point tracking.

These challenges provide the motivation to design novel intelligent and biologically-inspired advanced control strategies. Previously, to overcome some of these challenges, intelligent but not biologically-inspired approaches, such as the iterative learning control (ILC) algorithm has been developed to improve tracking performance in batch processes [35]. This approach was also applied to a batch cooling crystallization process [36]. Therefore, studies on the development and implementation of biologically-inspired advanced controllers for chemical/energy processes is scarce. To fill this gap, in this work, we propose a biologically-inspired optimal control strategy (BIO-CS) that mimics the ant's rule of pursuit idea. The performance of the proposed approach is illustrated through chemical/energy processes with constraints. For illustration purposes, a nonlinear fermentation process example case study is employed in this chapter.

3.2 Approach

3.2.1 Optimal Control Background

The optimal control problems in this work consider models that are characterized by the following system of differential and algebraic equations (DAEs):

$$\frac{dx}{dt} = f(x(t), u(t)); \quad g(x(t), u(t)) = 0 \quad (3.1)$$

in which $x(t) \in R^n$ are state variables, $u(t) \in R^m$ are input variables and t is time. Both sets of variables are constrained as: $x(t)^{lb} \leq x(t) \leq x(t)^{ub}$ and $u(t)^{lb} \leq u(t) \leq u(t)^{ub}$, where lb and ub stand for lower and upper bounds, respectively. f and g represent the nonlinear models relating the state and control variables of the chemical process in focus. The optimal control objective function for the given model can be defined as: $\min \int_{t_i}^{t_f} J(x(t), t) dt$. Here, J is the objective function to be minimized; t_i and t_f are the initial time and the final time, respectively. This defined optimal control problem will be used for the design of the proposed control strategy.

3.2.2 Proposed Framework

The proposed control strategy corresponds to a multi-agent-based algorithm that combines the ant's rule of pursuit idea with optimal control concepts for the calculation of

optimal trajectories of individual agents. In this rule, suppose that an ant finds food by walking around at random. This pioneer ant would then trace a wiggly path back to the nest and start group recruitment. The subsequent ants (or agents) would one after the other straighten the trail a little starting from the original path until the agent paths converge to a line connecting the nest and the food source, despite the individual ant's lack of sense of geometry [5]. Thus, by cooperating in large numbers, ants accomplish tasks that would be difficult to achieve individually. Figure 3.1 shows a schematic representation of the rule of pursuit for the ants.

Figure 3.1 Schematic representation of ants' rule of pursuit

Starting from the GSLP algorithm mentioned above, the following significant changes are made to enable the design of the proposed BIO-CS controller for chemical process control: (i) open-loop simulation profiles of the chemical process in focus are employed as the first agent's trajectory; (ii) the gradient-based optimal control solver RIOTS in GSLP is replaced by *dynopt* (described below) for higher computational efficiency (five times faster on average based on performed computational studies); (iii) the optimal control solver is called in an intelligent manner to allow the accommodation of the setpoint tracking problem (see catching up and free running parts in the algorithm structure below); (iv) the optimal control problem is formulated as minimum time control for fast setpoint tracking; and (v) Integrated Time Absolute Error (ITAE) is used as the stopping criterion of the algorithm. The steps of the resulting BIO-CS algorithm are detailed below.

Figure 3.2 outlines the proposed framework for the BIO-CS algorithm, in which the following steps are involved:

Start: Define the setpoint (y_{sp}) for the desired output variables and the initial conditions for all of the variables in the given dynamic process model of the system in focus;

(i) Generate agent0 trajectory for the inputs, $u_{k=0}(t)$, and outputs, $a_{k=0}(t)$, based on the previous knowledge of the system (e.g., open-loop simulation profiles for inputs and outputs);

(ii) Select algorithm parameters: pursuit time (Δ), sampling time (δ) and time horizon (T) such that $0 < \delta < \Delta < T$;

(iii) Specify $a_k(t)$ as the leader's trajectory and discretize such trajectory considering the following time steps: $t = \Delta + i\delta$, for $i = 0, 1, 2, \dots$ until, $t = T$;

(iv) Start the follower, $a_{k+1}(t)$, journey by trajectory generation according to the leader's position. Two scenarios are considered in this step depending on the leader's position:

1. Catching up: If the leader is not at the final position, i.e., $a_k(t = \Delta + i\delta) \neq a_k(t = T)$, solve the optimal control problem that takes the follower agent from its position at $a_{k+1}(t = i\delta)$ optimally to the leader's position at $a_k(t = \Delta + \delta i)$ and apply the obtained control law for one sample time δ ;
2. Free running: If $a_k(t = \Delta + i\delta) = a_k(t = T)$, then solve the optimal control problem that takes the follower agent from its position at $a_{k+1}(t = i\delta)$ optimally to the y_{sp} and apply the obtained control law for the remaining period of time;

(v) Compute the error of the follower agent with respect to the setpoint given by the following formula: $ITAE = \int_0^T t \frac{|a_{k+1}(t) - y_{sp}|}{y_{sp}} dt$; in which, ITAE stands for the Integrated Time Absolute Error;

(vi) Based on the termination threshold value (ε) stipulated for the algorithm, check if $|ITAE_{K+1} - ITAE_K| < \varepsilon$;

1. If yes, the follower agent trajectory converged to optimal solution;

2. If no, then repeat steps (iii)-(vi) by defining current follower as the next leader agent or $k = k + 1$;

(vii) Apply the obtained optimal control solution, $u_b(t) = u_{k+1}(t)$, as the input to the chemical process plant/simulation for one time step, or sampling time (δ);

(viii) Obtain the current state values, $x(t)$, from the process measurements or state estimator/observer and also calculate the error of the current output, $y(t)$, w.r.t. y_{sp} to close the control loop. Update the initial conditions based on the current operating point and move the final point according to a horizon of T (from current time t to a new time $t + T$) for the next control trajectory calculations by the algorithm.

Thus, the proposed BIO-CS algorithm consists of 8 steps described in further details here. In this algorithm, we consider the leader-follower nomenclature typically employed in the multi-agent-based control literature [2, 37, 4]. In essence, for a given dynamic system model, it is assumed that there is an available initially feasible trajectory for the output variables, which is obtained through prior knowledge of the system. In chemical processes, this trajectory could be obtained as the output profiles corresponding to the open-loop simulation results and employed as agent0, a_0 , or leader's trajectory for the initialization of the algorithm. The combination of intermediate setpoints can be used for the initialization or evolutionary approaches such as genetic algorithm could also be employed for this purpose.

Figure 3.2 BIO-CS closed-loop framework

Next, the parameters that define the leader-follower local interactions are specified, such as the pursuit time, Δ , and the sampling time, δ . Then, the optimal control trajectories for each follower are computed numerically using the *dynopt* solver (discussed in details below) by considering the leader's output trajectory position at Δ time units as the current target. The procedure described in step (iv) above is followed for the generation of the entire trajectory of the follower agent. Specifically, for the catching up part of the algorithm, a dynamic objective function is used such that the setpoint changes for each intermediate optimal control problem according to the profile obtained from the leader's trajectory. For the free running portion of the algorithm, the optimal control problem is solved for each agent in a single segment until it reaches the specified setpoint. Once this step is

completed, the Integrated Time Absolute Error (ITAE) for consecutive agents is computed considering the entire time horizon of the trajectories. If the absolute difference between the ITAE values of two consecutive agents is below a certain threshold (ε), then the algorithm is terminated at that particular agent. Otherwise, the current follower becomes the leader and the next agent begins its journey following the path of the new leader.

Remark 1: The main difference of the proposed BIO-CS algorithm when compared to model predictive control (MPC) approaches corresponds to its potential for faster computational time due to the algorithm nature. In particular, the multi-agent nature of BIO-CS enables: (i) the algorithm termination at a suboptimal solution corresponding to a specific agent; and (ii) the possibility of parallelizing the optimal control problems associated with different agents. These aspects would be critical for cases in which the computational time for the solution of the optimization problem becomes prohibitive. The formulation of BIO-CS as MPC and the similarities and differences between the two are further discussed in Chapter 8.

Remark 2: For the process application addressed in the work, we assumed that all the states are measured for step (viii). In case this assumption does not hold, a state estimator/observer, such as a pole placement-based, extended Kalman filter (EKF), or moving horizon estimator (MHE) would have to be employed. The combination of the BIO-CS with a state observer is addressed in Chapter 7 that describes an observer designed for the HYPER process application [38].

In the past, various optimal control solvers have been developed to address dynamic optimization problems for nonlinear systems [39, 40, 41, 42]. In this work, the gradient-based optimal control solver named *dynopt* (a freely available MATLAB® optimal control toolbox) is employed in the proposed algorithm framework to solve the intermediate problems associated with the local interaction of the agents. This solver has higher computational efficiency when compared to some of its counterparts (e.g., RIOTS) and capabilities to deal with nonlinear process models with constraints. Specifically, *dynopt* is a set of MATLAB® functions that use the orthogonal collocation on finite elements method

for the determination of optimal control trajectories. As inputs, this toolbox requires the dynamic process model as a set of Ordinary Differential Equations (ODEs) or DAEs, the objective function to be minimized, and the set of equality and inequality constraints that characterize the system. For the purposes of optimization, the constrained nonlinear minimization subroutine *fmincon* in MATLAB® is employed. The optimal control problem is solved by a simultaneous method called total discretization, in which both the control and the state variables are discretized using polynomials (e.g., Lagrange) of which coefficients become the decision variables in a much larger nonlinear programming (NLP) problem. In the toolbox, the basis functions are assumed to be known and the coefficients of their linear combinations are optimized [43]. In the next section, the fermentation process that serves as the initial application system for the proposed algorithm is described.

3.3 Results & Discussions

3.3.1 Fermentation Processes Case Study

In this subsection, the applicability of the proposed strategy to a chemical system is illustrated through a fermentation process case study. As shown in Figure 3.3, we consider a homogeneous, perfectly-mixed continuous culture fermentor for bioethanol production. Here, the microorganisms (*Zymomonas mobilis*) convert the substrate into a product, in this case ethanol [21, 20, 44, 45]. In this process, the outlet flow from the fermentation reactor contains the ethanol product, the unreacted substrate as well as the biomass. Biomass plays the role of catalyst for substrate conversion and is also a product of the fermentation process, while the substrate is a solution of glucose for feeding the microorganisms. Moreover, ethanol is the desired product of the process and also an inhibitor for the enzymatic reactions [46, 47].

Figure 3.3 Schematic representation of fermentation process

The challenges of the process model that describes this system include steady-state multiplicity and the oscillatory behavior caused by the system nonlinearities reported in the literature [46, 47, 48, 49, 50, 51, 52, 45]. These oscillations result in the reduction of product concentration values and the less efficient utilization of substrate at high values of dilution rates. An effective controller is therefore needed to take the process optimally to a selected steady state by eliminating the undesired oscillations. The design and implementation of such optimal controller for this fermentation process is a challenging task, mainly due to these highly nonlinear dynamics and the constraints imposed on the process variables. Motivated by these challenges, the proposed BIO-CS controller is implemented for this process to achieve the desired productivity with reduced oscillations. In this system, the dynamic nonlinear process model, characterized by the mass balances for the substrate (C_s), biomass (C_x), key component (C_e) and the product (C_p) concentrations, is expressed as:

$$\begin{aligned} \frac{dC_s}{dt} &= P \left(\frac{-1}{Y_{SX}} \right) \left(\frac{C_s C_e}{K_S + C_s} \right) - m_s C_x + D_{in} C_{s0} - D_{out} C_s \\ \frac{dC_x}{dt} &= P \left(\frac{C_s C_e}{K_S + C_s} \right) + D_{in} C_{x0} - D_{out} C_x \end{aligned} \quad (3.2)$$

$$\frac{dC_e}{dt} = [k_1 - k_2 C_p + k_3 C_p^2] \left(\frac{C_s C_e}{K_s + C_s} \right) + D_{in} C_{e0} - D_{out} C_e$$

$$\frac{dC_p}{dt} = P \left(\frac{1}{Y_{PX}} \right) \left(\frac{C_s C_e}{K_s + C_s} \right) + m_p C_x + D_{in} C_{p0} - D_{out} C_p$$

in which Table 3.1 provides the base case values and descriptions of the parameters and constants used in this fermentation problem. The initial values for these parameters and constants were taken from the literature [20, 21]. Thus, the fermentation model addressed here is represented by a set of four nonlinear differential equations adapted from the literature [21, 45, 44]. This nonlinear dynamic model consists of four output variables and one input/manipulated variable. The output variables considered are C_s , C_x , C_e , C_p and the manipulated variable is D_{in} (inlet dilution rate). Here, D_{in} is expressed as the ratio of the inlet flow rate and the volume of the fermentor. The key component concentration (C_e) includes RNA and the proteins present in the biomass that play an important role in biomass growth rate [20, 21]. In the given indirect inhibition structural model, the concentration of the key component is affected by both substrate and product concentrations. In particular, the key component formation is inhibited by the product concentration and is also a function of substrate concentration [46, 47]. In this case, the small values of C_{x0} and C_{e0} in the inlet concentrations are considered for the purpose of initializing the simulations. In addition, the inlet concentration of substrate, C_{s0} , is considered as a disturbance in the control studies performed below. For the controller design purposes, a single-input-single-output (SISO) control structure is selected based on previous studies in the literature [53], in which the manipulated input is the dilution rate (D_{in}), while the controlled output is the product concentration (C_p). For the control studies, constraints are placed on both the manipulated ($0 \text{ h}^{-1} < D_{in} < 3.0 \text{ h}^{-1}$) and the output variables ($C_s, C_x, C_e, C_p > 0 \text{ kg/m}^3$). Although the dilution rate values at steady state are under 1 h^{-1} for all performed simulations, the upper bound on the dilution rate is kept large to avoid more aggressive actions from the controller during transient. Here, the goal of the BIO-CS implementation is to maintain the outlet product concentration, C_p , at a desired steady state by controlling the input variable optimally. The fermentation model

discussed above is programmed in MATLAB® according to the *dynopt* specifications and then the proposed BIO-CS controller is implemented for the optimal control of the process. For simulating this process, the given system of differential equations is integrated using *ode15s* in MATLAB®. In the next subsection, the open-loop dynamics of the process as well as the BIO-CS simulation results for setpoint tracking and disturbance rejection scenarios are discussed in detail.

Table 3.1 Base set of parameters used for the fermentation process

Parameter	Description	Value
k_1	Empirical constant (h^{-1})	16.0
k_2	Empirical constant ($m^3/kg\ h$)	0.497
k_3	Empirical constant ($m^6/kg^2\ h$)	0.0038
m_s	Maintenance factor based on substrate ($kg/kg\ h$)	2.16
m_p	Maintenance factor based on product ($kg/kg\ h$)	1.1
Y_{SX}	Yield factor based on substrate (kg/kg)	0.0244
Y_{PX}	Yield factor based on product (kg/kg)	0.0526
K_S	Monod constant (kg/m^3)	0.5
C_{s0}	Inlet substrate concentration (kg/m^3)	150.3
C_{x0}	Inlet biomass concentration (kg/m^3)	0.08
C_{p0}	Inlet product concentration (kg/m^3)	0
C_{e0}	Inlet key component concentration (kg/m^3)	0.02
V_F	Fermentor volume (m^3)	0.003
D_{in}	Inlet fermentor dilution rate (h^{-1})	0.5
D_{out}	Outlet fermentor dilution rate (h^{-1})	0.5
ρ	Ethanol density (kg/m^3)	789
P	Maximum specific growth rate (h^{-1})	1.0

3.3.2 Open-loop Results

In this subsection, the dynamics of the fermentation process model are analyzed before the implementation of the controller. For the first set of simulations, open-loop scenarios

are considered in which the manipulated variable, D_{in} , is kept constant at different values. Figure 3.4 depicts the concentration profiles of the four components of the system for the open-loop simulations. These plots show the effect of D_{in} values when varied from 0.05 h^{-1} to 0.5 h^{-1} on the concentration profiles. Note that the D_{in} values higher than 0.1 in general are more prone to sustain oscillations in the output trajectories until they reach the steady state. The time required to reach steady state and the steady-state value itself are different for each D_{in} scenario. This analysis also shows that the steady-state concentration of product, C_p , varies significantly with D_{in} . In addition, the oscillatory behavior that arises from the interaction between cell growth, substrate consumption, and the ethanol production portrayed in Figure 3.4 is in agreement with the literature [46, 47, 49]. The potential existence of multiple steady states and the oscillations are undesirable in terms of system's performance and stability. The mitigation of these issues motivates the need for optimal control of this system.

3.3.3 Closed-loop Results

Setpoint tracking

Case I:

The results of the first implementation of the proposed BIO-CS algorithm for the fermentation system are described here. For such implementation, the fermentation control problem is formulated with an objective function associated with the setpoint tracking of product concentration as: $J = \min \int_0^t (C_p - C_{p,sp})^2 dt$.

Figure 3.4 Open-loop simulation results: output concentration profiles for different D_{in} values

As the first step in the algorithm, the agent0 trajectory for C_p is generated by the open-loop simulation profile described above with constant $D_{in} = 0.5 \text{ h}^{-1}$ as shown in Figure 3.5 (left). This open-loop case is particularly of interest because the concentration profiles for higher D_{in} values present additional challenges associated with the inefficient use of substrate and product inhibition effects. Such challenges ultimately reflect in lower steady-state product concentration when compared to other cases. To improve the

system performance, the proposed BIO-CS framework is implemented to control the product concentration optimally with minimal oscillations and fast response time to reach a desired steady state. For the implementation, the algorithm parameters are defined as $\Delta = 10 \text{ h}$, $\delta = 2 \text{ h}$ and $T = 20 \text{ h}$. The setpoint, $C_{p,sp}$, is fixed at 65 kg/m^3 , which is higher than the open-loop steady-state concentration for the corresponding D_{in} . This selected higher setpoint is in the range of product concentrations achievable with the given inlet concentrations, which are specified as $C_{s0} = 150.3 \text{ kg/m}^3$, $C_{x0} = 0.08 \text{ kg/m}^3$, $C_{e0} = 0.25 \text{ kg/m}^3$ and $C_{p0} = 4 \text{ kg/m}^3$. In this case, the inlet concentration of C_p is selected to be greater than 0 as previous studies in the literature indicated that the injection of some components of the product stream in the inlet favors system stability [20, 21]. Given this problem setup, starting from the agent0 as the initial trajectory, a series of agents whose dynamics are given by the same fermentation process model are simulated by employing the BIO-CS framework outlined in Figure 3.2. In this case, the plant/process simulation model and the controller model are assumed to be identical. Also, for the catching up part of the algorithm, a dynamic objective function is used such that the followers track the setpoint obtained from the C_p profile of the leaders for each intermediate optimal control problem. For the free running part, the setpoint is fixed at 65 kg/m^3 .

Figure 3.5 depicts the BIO-CS closed-loop simulation results for different agents. Note that the agent1 trajectory in Figure 3.5 (middle) already shows progress towards achieving the desired setpoint particularly in free running part, i.e., from 10 h to 20 h . However, this agent's C_p trajectory still has oscillations and small overshoots with respect to the setpoint. As the number of agents increases, these oscillations are mitigated in both the catching up and the free running parts of the concentration profiles. As a result, Figure 3.5 (right) depicts an agent5 trajectory for C_p a lot smoother and with no overshoot from 10 h to 20 h when compared to the agent1 profile. For this case study, the termination criterion threshold for the algorithm is stipulated as $|ITAE_{K+1} - ITAE_K| < \varepsilon = 0.1$, in which the ITAE for the C_p trajectory is calculated over the time horizon of 0 h to 20 h according to step (v) of the algorithm framework. As expected, the ITAE value

decreases as the number of agents increases until the simulation is terminated at agent5 when the difference between the ITAE value of agent4 and agent5 is below the termination threshold. The calculated ITAE value differences are reported in Table 3.2. Also, the optimal solution given by agent5 output trajectory shows approximately 95% error improvement when compared to agent1 solution. Similar progression in terms of smoothness and less oscillations can be observed in the input trajectories as displayed in Figure 3.5 (bottom middle and right). It is worth noting that potential issues may arise in practice due to the low value of the flow rates employed in this and other case study simulations. If that is the case, to overcome such issues, the user may set the bounds of the input flow rates in the controller optimization problem formulation in order to satisfy valve manufacturing constraints. This case indicates that the cooperative work among the agents results in optimized and smooth trajectories connecting the start to the desired setpoint. During the computations, 2 minutes (in CPU time) on average was taken to generate each agent trajectory from start to end. The *dynopt* computational time to solve a single intermediate optimal control problem was 20 CPU seconds (on average) during the calculation of each agent's trajectory. As a result, the total computational time to generate the BIO-CS optimal solution was 10 minutes which is also reported in Table 3.3. Due to the fermentation system dynamics, this obtained computational time is not an issue for online implementation as process sampling time of 2 h is assumed. All the simulation studies in this work were carried out on an Intel Core i7 (Sandy bridge) 3.40 GHz processor.

Figure 3.6 shows the results of the implementation of the BIO-CS controller in comparison to a single *dynopt* and a classical feedback PI controller. As depicted in Figure 3.6 (left), the product concentration profile of the single *dynopt* implementation presents a highly oscillatory behavior until it reaches the setpoint. For the single *dynopt* method, the optimal control problem is solved considering the entire time horizon, i.e., 20 h, subdivided in 10 intervals of 2 h length. This discretization is performed following the BIO-CS method. The

number of collocation points and other *dynopt* parameters used in this case are also same as BIO-CS implementation for consistency.

Table 3.2 BIO-CS case studies: results and parameters

Case studies		Setpoint tracking		Disturbance rejection	
		I	II	III	IV
Inlet concentrations (kg/m^3)	C_{p0}	4	0	0	0
	C_{e0}	0.25	0.02	0.02	0.02
Time horizon (h)	t_f	20	30	30	30
Threshold value	ε	0.1	100	50	50
ITAE calculation (t_i to t_f)		0-20	0-30	14-30	14-30
Difference of $ITAE_k$ values for agents ($k = 0, 1, 2, \dots, 7$)	$ ITAE_1 - ITAE_0 $	39959.57	76302.04	69233.21	65858.00
	$ ITAE_2 - ITAE_1 $	497.07	11262.21	1794.07	4358.15
	$ ITAE_3 - ITAE_2 $	19.51	5758.23	1410.31	3511.19
	$ ITAE_4 - ITAE_3 $	1.67	1007.52	30.02	4061.87
	$ ITAE_5 - ITAE_4 $	0.04	79.00	-	468.90
	$ ITAE_6 - ITAE_5 $	-	-	-	89.61
	$ ITAE_7 - ITAE_6 $	-	-	-	34.20

Figure 3.5 BIO-CS closed-loop simulation results (case I): concentration and input profiles for agent0 (left), agent1 (middle), and agent5 (right)

The PI controller results depicted in Figure 3.6 (middle) display sluggish response for the concentration profile of the product. In particular, the PI controller design is based on the classical feedback PI formulation with $K_c = -0.00099$ (proportional gain) and $\tau = 40$ min (integral time) obtained by extensive trial and error simulations. The sampling time used for the controller implementation is 0.01 hour. For the tuning of the PI controller parameters, extensive simulations were run considering different tuning scenarios (e.g., high and low gains) without improved performance. The addition of the derivative component to the controller framework was also attempted for improved response, but the product concentration profiles sustained oscillations within a range of $\pm 5 \text{ kg/m}^3$ with

respect to the setpoint for the entire time horizon. Even though the rise time is similar for all the controllers, the settling time for the PI controller is much longer of approximately 50 h (see longer horizon in Figure 3.6 (middle)). On the other hand, the BIO-CS and single *dynopt* implementations bring the process close to the desired setpoint in around 3 h and 18 h , respectively. The approximate time to reach steady state and average computational time in CPU minutes for each method are also reported in Table 3.3 for comparison. Note that the computational time obtained for the BIO-CS implementation is shorter when compared to single *dynopt*. The computational time reduction and improved performance of the BIO-CS algorithm can be explained by the fact that the proposed algorithm corresponds to a systematic and intelligent manner of calling the optimal control solver to obtain the desired system performance. In particular, the single *dynopt* method solves a series of optimal control problems that have larger errors with respect to the setpoint at each time interval. On the other hand, the BIO-CS approach considers the previous agent trajectory as the setpoint for the next agent. The smaller errors between these trajectories enable the solution of easier and faster optimal control problems. Specifically, once the setpoint is achieved (in this case in the first two samples), the optimal control problem becomes trivial and thus can be solved in a faster manner. Finally, the optimal control laws for the proposed algorithm depicted in Figure 3.6 (right-bottom) are also smoother when compared to the single *dynopt* implementation. Therefore, this case shows the advantages of the proposed algorithm vs. other approaches including offset free and faster setpoint tracking with smoother optimal control solution profiles.

Case II:

In the second setpoint tracking case, the inlet concentrations of the fermentation problem are changed from the previous case by adjusting $C_{p0} = 0 \text{ kg/m}^3$ and $C_{e0} = 0.02 \text{ kg/m}^3$. This scenario is considered to check the robustness of the proposed controller under more challenging operating conditions. For this scenario, all of the controller parameters including the agent0 trajectory are considered same as in the previous case study. Also, the final time horizon considered is now of 30 h to allow sufficient time for the process to

reach the steady state. To accommodate the effects of increasing this final time on the ITAE calculations, the termination criteria for the BIO-CS trajectory computations is changed to $\varepsilon = 100$. Figure 3.7 depicts the BIO-CS implementation results for this case with agent5 as the optimal agent. Here once again, the improvement of the agents' trajectories as described in Case-I is observed from Figure 3.7 (left to right). From the comparison of Figure 3.7 (right) and Figure 3.5 (right), it is observed that the beginning part of the input profile is altered to accommodate the change in the operating conditions. This input eventually adjusts itself optimally to keep the product concentration at the steady value. The change in the inlet concentrations also affects the performance of the BIO-CS in terms of overshooting and settling time. Moreover in this case, the optimal solution shows roughly 95% error improvement when compared to agent1 trajectory. As this case is more challenging than Case-I, poor performance was also expected and observed for the PI controller and single *dynopt* implementations. The details on the results for the BIO-CS, PI and single *dynopt* implementations are documented in Table 3.2 and Table 3.3.

Table 3.3 Performance comparison among implemented control methods

Method	Approx. time to steady state (h)				Avg. computational time for all cases (CPU min.)
	Setpoint tracking		Disturbance Rejection		
	Case I	Case II	Case III	Case IV	
Single <i>dynopt</i>	18	8	8	7	50
PI	50	40	25	25	0.5
BIO-CS	3	6	5	5	10

Figure 3.6 Closed-loop implementation results (case I): single *dynopt* (left), PI controller (middle), and BIO-CS (right)

Figure 3.7 BIO-CS closed-loop simulation results (case II): concentration and input profiles for agent0 (left), agent2 (middle), and agent5 (right)

Disturbance rejection

Case III:

To further challenge the proposed controller, a case is simulated by imposing a disturbance in the inlet substrate concentration, C_{s0} . This disturbance would reflect the abrupt change in the inlet substrate concentration after a given period of time due to operational changes coming from an upstream process. For this scenario, a disturbance is generated as a step decrease in the value of C_{s0} by 20% (30 kg/m^3) at the time of 14 h and the other inlet concentrations from Case-II are used. To keep the system at the

desired operating point, the proposed BIO-CS approach is implemented to obtain an optimal system performance by mitigating the effects of the disturbance. For such implementation, the initial feasible trajectory for agent0 is generated from the open-loop simulation with the disturbance as shown in Figure 3.8 (left). Also, the BIO-CS algorithm parameters are selected as the same as in Case-II and the setpoint for the product concentration, $C_{p,sp}$, is kept at 65 kg/m^3 . Figure 3.8 shows the results of the BIO-CS implementation for this case. Note that agent1 shown in Figure 3.8 (middle) struggles to keep up with the setpoint immediately after the introduction of the disturbance, i.e., between 14 h and 23 h due to the insufficient availability of the substrate for product formation. The subsequent agents learn from their leaders and as a result of this cooperative work, the oscillations are mitigated in agent4 profile as depicted in Figure 3.8 (right), yielding a smoother trajectory for the output product concentration after the disturbance. Particularly, the optimal solution in agent4 shows an approximate 6% error reduction when compared to agent1. Figure 3.8 (bottom) plots show the evolution of the input profiles of these agents for this particular case. Note that D_{in} is adjusted optimally to minimize the disturbance effect at agent4.

In Figure 3.9, the plots with the results for the single *dynopt* and the classical PI controller implementations are shown for comparison with the BIO-CS. Note that here once again, the performance of the single *dynopt* implementation shown in Figure 3.9 (left) is poorer than BIO-CS before and after the disturbance due to similar reasons as explained in Case I above. The PI controller in this case is designed with $K_c = -0.00099$ (proportional gain) and $\tau = 40$ (integral time). As depicted in Figure 3.9 (middle), this controller displays once again sluggish and slow response for the concentration profiles, with a settling time of approximately 35 h after the application of the disturbance. Also, after the introduction of such disturbance in C_{s0} , the oscillations in the product concentration profile are amplified when compared to the setpoint tracking case in Figure 3.6 (middle). Thus, the BIO-CS implementation results in Figure 3.9 (right) indicate the potential of the proposed strategy to tackle disturbances in process operation.

Figure 3.8 BIO-CS closed-loop simulation with -20% disturbance in C_{s0} (case III): concentration and input profiles for agent0 (left), agent1 (middle), and agent4 (right)

Figure 3.9 Closed-loop implementation results with disturbance (case III): single *dynopt* (left), PI controller (middle), and BIO-CS (right)

Case IV:

A case is simulated in which an additive disturbance is introduced in the inlet substrate concentration, C_{s0} , while the process is in operation. For this purpose, a step increase in the value of C_{s0} by 20% (30 kg/m^3) at the operating time of 14 h is imposed. For the implementation here, the BIO-CS algorithm parameters, inlet concentrations and the setpoint for the product concentration, C_p , are considered the same as in the previous case. Also, the initial trajectory for the BIO-CS, or agent0, is produced by the open-loop simulation profile with the disturbance as depicted in Figure 3.10 (left). Figure 3.10 also

shows the evolution of the product concentration and the input profiles for agent1 and agent7. For agent1 depicted in Figure 3.10 (middle), after the introduction of the disturbance, the product concentration profiles move away from the setpoint as expected. With the agents' progression, the cooperative work among the agents enables once again the mitigation of the disturbance effects, as shown for agent7 in Figure 3.10 (right). There is also an error reduction of 7% for the agent7 solution when compared to agent1. Although the disturbance is additive in this case, the optimal output profile for agent7 shows reduction after the introduction of the disturbance due to the fact that the increased inlet substrate concentration causes higher inhibition effects resulting in less product formation [49, 50, 54]. For this case, the input trajectories also show convergence towards smoother profiles as displayed in Figure 3.10 (bottom). The BIO-CS results for this case when compared to the classical PI controller and single *dynopt* implementations also display the superior behavior of the proposed approach as in the previous cases (see Table 3.3 for performance comparison). Therefore, the setpoint tracking and disturbance rejection scenarios discussed demonstrate the potential of the BIO-CS controller for addressing challenges present in the operation of nonlinear chemical and energy processes.

Plant-model mismatch

Case V and VI:

Finally, two cases are simulated considering plant-model mismatch scenarios. In particular, we consider parametric uncertainties in: (i) the kinetic model by changing k_1 (kinetic empirical constant) by -20%; and (ii) the system properties by altering m_s (maintenance factor based on substrate) by -20%. In both examples, these parameters are modified in the plant model while keeping the controller model with their original values.

Figure 3.10 BIO-CS closed-loop simulation with +20% disturbance in C_{s0} (case IV): concentration and input profiles for agent0 (left), agent1 (middle), and agent7 (right)

For improved performance, the BIO-CS parameters are re-tuned for the controller implementations as $\Delta = 1 \text{ h}$ and $\delta = 0.2 \text{ h}$. The obtained results for both scenarios are shown in Figure 3.11. Here, agent0 is assumed to be same as for Case I. As the number of agents progresses, the product concentration profiles for both cases become smoother and achieve the desired setpoints, thus showing successful implementations of the controller. Note that the increased number of agents for the termination of the algorithm (assuming a threshold of $\varepsilon = 2$) in comparison to the other performed cases above, 18 and 17 for the k_1 and m_s changes, respectively (also see Figure 3.11, middle and right), is due to the additional challenges introduced by the plant-model mismatches.

Figure 3.11 BIO-CS closed-loop simulation with plant-model mismatch (cases V and VI): concentration and input profiles for agent0 (left); agent18 (k_1 -20%) (middle) and agent17 (m_s -20%) (right)

3.4 Conclusions

This chapter introduced an agent-based optimal control strategy for nonlinear chemical and energy processes that is inspired by the rule of pursuit for ants. In the developed BIO-CS algorithm, starting from an initially feasible trajectory, optimal paths for the agents are generated. The gradient-based optimal control solver, *dynopt*, was employed for the calculation of optimal control laws associated with the leader-follower interactions. The proposed algorithm corresponds to a systematic and intelligent manner of calling the

optimal control solver by considering such interactions and the cooperation among the agents to obtain the desired system performance. The proposed strategy was successfully applied to the fermentation process to produce ethanol. The results of this implementation for setpoint tracking, disturbance rejection, and plant-model mismatch cases showed that the output concentration profiles of the process converged to smooth trajectories with reduced oscillations, thus optimizing the system performance. The developed BIO-CS controller also showed performance that was superior to that of the classical PI and single *dynopt* controllers in terms of offset-free and faster responses. The proposed BIO-CS approach has the following noteworthy features that enables its implementation for process control of chemical and energy processes: (i) capability of dealing with process models of different complexities, such as first-principles-based or input-output-data-based models; (ii) flexibility of employing optimal control solvers of different nature, for example, gradient-based or probabilistic-based solvers; (iii) possibility of handling a variety of objective functions associated with set point tracking or minimum time optimal control; and (iv) versatility in defining the termination criteria, for example, integrated absolute error or integrated squared error, could be used in addition to ITAE. Thus, the application of the proposed biologically inspired optimal control strategy for chemical/energy processes was successfully demonstrated for different scenarios.

4 Incorporation of Adaptive Component into Biomimetic Controller Design

4.1 Introduction

This chapter focuses on the incorporation of an adaptive component into the biomimetic controller developed in the previous chapter. This work is performed in collaboration with Dr. Perhinschi's group. The proposed framework is then implemented for tackling a subsystem of an Acid Gas Removal section of an Integrated Gasification Combined Cycle (IGCC-AGR) process simulation considering realistic scenarios involving setpoint tracking and plant-model mismatch that could be encountered in practice. The IGCC-AGR system uses the physical solvent Selexol to selectively remove CO₂ and H₂S from the syngas feed. This process is critical as the IGCC plants with CO₂ capture should be operated optimally without violating operational and environmental constraints. A 2×2 (two-input-two-output) control structure associated with the CO₂ absorber process unit of the IGCC-AGR is selected for the controller design work. A dynamic process model derived from the input-output data of the selected subsystem is employed in the BIO-CS controller framework. Specifically, the existence of plant-model mismatches for the control of multivariable systems has not yet been addressed in the BIO-CS design. To fill this gap, BIO-CS is extended to incorporate an adaptive component into the controller formulation. In particular, the Artificial Neural Network (ANN)-based adaptive mechanisms developed in Dr. Perhinschi's group are considered for the formulation of the adaptive component. The details of the developed approach including the implementation case study are explained below.

In the past, adaptive control of process systems has been widely studied in the literature. Such literature was mainly focused on developing self-tuning and pole placement approaches for adapting the control strategy through online parameter estimation according to operating conditions [55, 56, 57]. Additionally, learning-based algorithms have been proposed to assist PID controller tuning by using output error information for direct adaptive control of chemical process systems [58]. Also, iterative learning control, repetitive control, and run-to-run control, collectively referred to as learning-type adaptive

control mechanisms, have been studied to address repetitive and run-based processes [59]. To determine plant-model mismatches for multiple-input multiple-output systems, an autocovariance-based estimation methodology for operation under model-based control has also been investigated in the literature [60]. ANNs have also been studied as adaptive control elements for the improvement of power plant performance [61]. The application of neural networks in process control has been performed in the past considering a state-space bounded derivative network with extrapolation capability for automatic control [62, 63]. Recently, immunity-based and neural network-based adaptive mechanisms have been proposed for the detection, identification, and evaluation of abnormal conditions in power plants [64, 65]. These techniques can serve as effective tools for advanced monitoring and control of energy systems. However, the incorporation of an adaptive component into biomimetic and agent-based controllers has been rarely explored in the literature. In this chapter, a novel controller framework is developed that combines an ANN-based method with BIO-CS to address advanced energy systems, such as IGCC with carbon capture.

4.2 Approach

4.2.1 Optimal Control Background

The analyzed optimal control problem associated with a multivariable control structure consists of finding the vector of input variables that minimizes the tracking error with respect to the setpoint while suppressing the control activity, subject to state and input constraints. In general, this optimal control problem is formulated as follows:

$$\min_{u(t)} J = \int_{t_i}^{t_f} \left(\|y(t) - y_{sp}\|_Q^2 + \|u(t) - u^-(t)\|_R^2 \right) dt \quad (4.1)$$

$$\text{such that, } \dot{x}(t) = Ax(t) + Bu(t); \quad y(t) = Cx(t) + Du(t)$$

$$\text{subject to, } x(t)^{lb} \leq x(t) \leq x(t)^{ub}; \quad u(t)^{lb} \leq u(t) \leq u(t)^{ub}$$

in which matrices A, B, C, and D represent the reduced mathematical models obtained from the original nonlinear system that relates the state, output and input variables of the process in focus that were defined in the previous chapter. In the definitions of the

inequality constraints, superscripts lb and ub stand for lower and upper bounds, respectively. The optimal control objective function for the given model, denoted by J , is minimized over a period of time. Here, t_i and t_f are the initial and final times, respectively. In the objective function, Q and R stand for the weighting matrices for output and input variables, respectively. The objective function (J) generally consists of simultaneously maintaining multiple outputs at their desired setpoints, y_{sp} , with the addition of the input suppression term that considers past input moves, $u^-(t)$. The function terms are minimizations of the squared errors between the variables and their desired/past values.

The solution of this optimal control problem provides optimal trajectories for each input of the multiple-input-multiple-output (MIMO) system addressed to satisfy the defined objective function and constraints. In this work, orthogonal collocation methods and the MATLAB® function *fmincon* are used to solve the constrained optimization problem. Specifically, the gradient-based MATLAB® toolbox entitled *dynopt* (described above) is called following the ant's rule of pursuit idea according to the BIO-CS algorithm design. The procedure for designing the BIO-CS controller in MATLAB® was explained in the previous chapter. This design is modified here to accommodate a specified maximum number of agents, k_{max} , if computational time restrictions for the agents are present. Note that the chosen termination criterion is selected due to the computational time limitations imposed by the online implementation of BIO-CS on the process simulation software environment that is discussed in the next chapter. For the error, particular users may choose different criterion, such as Integrated Squared Error (ISE), if needed.

4.2.2 Integrated BIO-CS Framework with Adaptive Component

The schematic representation of the integrated BIO-CS framework with ANN-based adaptive component for online implementation in MATLAB® is outlined in Figure 4.1. This framework is mainly subdivided in two main parts: (i) BIO-CS controller design described previously; and (ii) incorporation of the adaptive component laws into the BIO-CS control laws. The details of the latter part are presented below. Here, the BIO-CS control laws are computed as u_b trajectories for the given sample time horizon as discussed

previously. Then, employing these trajectories, the adaptive component control adjustments, u_a , are calculated in MATLAB® at each sample time using the information associated with the tracking errors, e , and available outputs, y , from the process simulation as the inputs to the adaptive mechanism. For the implementation in the MATLAB® plant simulation, the BIO-CS control actions, u_b , are augmented with the adaptive control laws, u_a , using user-defined gains. The details on the computation of adaptive control laws, u_a , are given next.

Figure 4.1 Schematic representation of the BIO-CS controller with adaptive component framework (in MATLAB®)

The adaptive control laws are designed to augment the baseline BIO-CS laws under challenging operating conditions, such as occurrences of plant-model mismatches. In this case, the Artificial Neural Network (ANN)-based mechanism is selected for the generation of adaptive control laws. This adaptive method relies on the capability of the ANNs to approximate tracking errors due to uncertainties, perturbations, modeling inaccuracies, and abnormal operating conditions. Specifically, a single-hidden-layer (SHL) ANN [66] with online learning capacity is employed in this study. The conventional architecture of the ANN generally used for such application purposes is presented in Figure 4.2.

Figure 4.2 Architecture of the ANN-based adaptive component

As shown in Figure 4.2, in general, each artificial neuron, j , inside the neural network is an information processing unit with multiple inputs. Such neuron j receives the inputs, ξ , and it produces the network output, ϕ , that propagates to the exterior of the network. In particular, the artificial neural network output is computed as the weighted sum of all inputs modified by an activation function σ . The general form of the equation used for such function transformation is given in the hidden layer of the neural network, as outlined by the green ellipse in Figure 4.2. As depicted in Figure 4.2, $[\xi_1, \xi_2, \dots, \xi_{q-1}, \xi_q]$ are the inputs, $[w_1, w_2, \dots, w_{r-1}, w_r]$ are the weights, ϕ_j is the output, and θ_w is the bias term corresponding to each j neuron. Table 4.1 summarizes all the notations and the possible bounds on numbers that describe the process and ANN variables.

In addition to the generation of the output at the neuron individual level, an adjustment of the weights needs to be carried out as part of the learning process. For this purpose, a data historian is used to train the neural network and weights are adjusted accordingly. In case of online learning ANN, these weights are updated at each time once the data becomes available. This feature of the ANN is advantageous to address plant-model mismatches since the data itself is generated as a function of time and used for dynamic

adaptation to situations. Next, the specific details on the online learning ANN employed in this study, such as the activation function and laws used for updating the weights are discussed.

Table 4.1 Summary of the notations used for describing process and ANN variables

Description	Notation	Possible bound
Process inputs	$[u_1, u_2 \dots \dots u_{m-1}, u_m]$	$m = [1, 2 \dots, m]$
Process outputs	$[y_1, y_2 \dots \dots y_{p-1}, y_p]$	$p = [1, 2 \dots, p]$
Process states	$[x_1, x_2 \dots \dots x_{n-1}, x_n]$	$n = [1, 2 \dots, n]$
ANN inputs	$[\xi_1, \xi_2 \dots \dots \xi_{q-1}, \xi_q]$	$q = [1, 2 \dots, m + n + p]$
ANN outputs	$[\phi_1, \phi_2 \dots \dots \phi_{j-1}, \phi_j]$	$j = [1, 2 \dots, m]$
ANN weights	$[w_1, w_2 \dots \dots w_{r-1}, w_r]$	$r = [1, 2 \dots, r \geq q]$

Assume a general structure where ξ_i inputs, $i = 1, 2, \dots, q$; and w_s weights, $s = 1, 2, \dots, r$; are in the hidden layer, and m outputs (ϕ_j) are present in the ANN. Any ANN output ϕ_j , $j = 1, 2, \dots, m$ is computed using the following relationship:

$$\phi_j = \sum_{s=1}^r \left[w_{sj} \sigma \left(\sum_{i=1}^q v_{si} \xi_i + \theta_{vs} \right) + \theta_{wj} \right] \quad (4.2)$$

in which w_{sj} are the interconnection weights between the hidden layer and the output layer, v_{si} are the interconnection weights between the input layer and the hidden layer, and θ_{vs} and θ_{wj} are bias terms. The activation function, σ , used in this study is a sigmoid function represented as follows, along with its derivative, $\dot{\sigma}$:

$$\sigma(\vartheta) = \frac{1}{1 + e^{-\beta\vartheta}} ; \quad \dot{\sigma}(\vartheta) = \frac{d\sigma}{d\vartheta} = \beta * \sigma(1 - \sigma) \quad (4.3)$$

where the activation potential β is a design parameter. In the ANN with online learning capability, the interconnection weights are updated at each time instant according to the following update laws:

$$\begin{aligned}\dot{w} &= -\gamma_w[(\sigma - \dot{\sigma} * v^T \xi)e^T + \lambda_w * \|e\| w] \\ \dot{v} &= -\gamma_v[\xi * e^T * w^T * \sigma + \lambda_v * \|e\| v]\end{aligned}\tag{4.4}$$

in which e is the tracking error and γ_w , γ_v , λ_w , λ_v are learning rates or design parameters of the adaptive component. Also, $\dot{w}$ and $\dot{v}$ are the updated interconnection weights that are used for the next control move calculations. The values for the design parameters mentioned above are selected by the user. The stability of the ANN-based adaptive component considering PID as a baseline controller was demonstrated in the final DoE report associated with this project [67]. During the implementation of the proposed controller framework (BIO-CS with ANN), no stability issues have occurred. However, further stability guarantees for BIO-CS are yet to be proved and thus is a recommendation for future work.

Next, the application of BIO-CS with ANN-based adaptive component to an advanced energy system is demonstrated via the IGCC-AGR subsystem.

4.3 Results & Discussions

4.3.1 Model Development for BIO-CS Design

The implementation of BIO-CS on an advanced energy system is demonstrated using a subsystem of the AGR section of the IGCC process simulation as an example case study. In this subsection, the steps involved in the development of the data-driven dynamic process model of the IGCC-AGR system are described. The obtained process model is embedded in the BIO-CS controller. In this work, the classical autoregressive model with exogenous inputs (ARX) method is used for deriving the data-driven dynamic model in MATLAB® [68, 69]. In the ARX model structure, the dynamic characteristics of the process under study are captured in the observed (measured) variables of the process, namely the output signal or controlled variable, $y(\tau)$, and the input signal or manipulated variable,

$u(\tau)$. The input-output relationship in this structure is described by the following linear difference equation model:

$$\begin{aligned}
y(\tau) + a_1 y(\tau - 1) + \dots + a_{n_a} y(\tau - n_a) \\
&= b_1 u(\tau - 1) + \dots + b_{n_b} u(\tau - n_b) + e(\tau) \\
\theta &= [a_1 a_2 \dots a_{n_a} \dots \dots b_1 b_2 \dots b_{n_b}]^T \\
A(q) &= 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a} \\
B(q) &= b_1 q^{-1} + \dots + b_{n_b} q^{-n_b} \\
u &\rightarrow \left[\frac{B(q)}{A(q)} \right] \rightarrow y
\end{aligned} \tag{4.5}$$

in which θ is the vector of the model parameters that are identified by a least-squares method using the collected input-output data set. Also, the coefficients a and b characterize the $A(q)$ and $B(q)$ polynomials of orders n_a and n_b in the z domain. Here, the q operator is employed to be consistent with the conventional definition of the z -transform. In particular, the ARX model development is conducted by employing the *arx* function available in MATLAB®. The input to this function is the generated data set and the output of this function is a discrete-time polynomial in the z domain. To obtain the mathematical model based on the ARX technique, data sets that capture the dynamics of the process are generated in DYNsim®. For this purpose, simulated step tests are designed by individually moving the input variables as steps and collecting the resulting data for the output variables. The collected data can then be normalized depending on the different magnitudes of the input and the output variables. The obtained ARX model in MATLAB® is further processed following the steps outlined in Figure 4.3 to generate continuous-time state-space matrices in the form of A, B, C, D. These steps are necessary as BIO-CS requires in its formulation the process model in the form of continuous-time differential equations. For implementation in the BIO-CS controller, the ARX polynomial is first converted to a discrete-time state-space model using the *idss* function in MATLAB® and then to a continuous-time state-space model employing the *d2c* function.

Figure 4.3 Summary of the model conversion steps for BIO-CS controller design

IGCC-AGR process description

In this subsection, the subsystem of IGCC-AGR process example is described. Integrated Gasification Combined Cycle (IGCC) is a technology that uses a gasifier to turn coal and other carbon-based fuels into synthesis gas (syngas). Impurities are removed from the syngas in the Acid Gas Removal (AGR) section prior to the power generation combined cycle [22]. A combined-cycle power plant employs both gas and steam turbine interactions and is typically 40% more efficient than a traditional single-cycle plant with the same fuel [70]. Specifically, the waste heat from the gas turbine is recovered to be used for steam generation for feeding the nearby steam turbine, which generates extra power. IGCC plants are advantageous in comparison to conventional coal-fired power plants due to their high thermal efficiency, low greenhouse gas emissions and capability to process low grade coal [22, 71]. For the implementation case study, a 2×2 (two-input-two-output) control structure associated with the CO₂ absorber process unit of the IGCC-AGR is selected for the plant simulated in DYNsIM[®] as shown in Figure 4.4. The subsystem of the IGCC-AGR process addressed here involves an absorption column that uses the physical solvent Selexol to selectively remove CO₂ from the syngas feed [72]. This process is critical to the optimal operation of the IGCC plants with CO₂ capture under operational and environmental constraints. The description of the variables associated with the selected control system are also depicted in Figure 4.4.

IGCC AREA 1000 CO2 ABSORBER

SIMSPD 100 %

Figure 4.4 Schematic of multiple-input-multiple-output (MIMO) system of IGCC-AGR process (in DYNsIM®)

As the first step in the controller design, the data-driven dynamic process model is derived using the steps described in the previous subsection. The obtained model is able to fit approximately 85% of the data, confirming its accuracy. The obtained process model consists of a continuous-time state-space representation as given below,

$$\begin{aligned} A &= \begin{bmatrix} -0.0005 & 0 \\ 0 & -0.0017 \end{bmatrix}; B = \begin{bmatrix} -0.0151 & -0.0008 \\ 0.0243 & -0.0148 \end{bmatrix} \\ C &= \begin{bmatrix} 0.0317 & 0 \\ 0 & 0.0657 \end{bmatrix}; D = \begin{bmatrix} -0.0146 & -0.0008 \\ 0.0505 & -0.0308 \end{bmatrix} \end{aligned} \quad (4.6)$$

This state-space model is employed as the BIO-CS controller model inside the framework. In addition, the objective function associated with the 2x2 MIMO system considered consists of simultaneously maintaining the 2 outputs at their desired setpoints with the input suppression term as follows:

$$\min_{u_1(t), u_2(t)} J = \int_{t_i}^{t_f} [(y_1 - y_{1,sp})^2 + (y_2 - y_{2,sp})^2 + (u_1 - u_1^-)^2 + (u_2 - u_2^-)^2] dt \quad (4.7)$$

4.3.2 BIO-CS Stand-alone Implementation

In this subsection, the results for the BIO-CS stand-alone implementation in the IGCC-AGR CO₂ absorber subsystem are explained. For such implementation, the selected control system is simulated in MATLAB® employing the BIO-CS control actions as the input to the plant. The results for such implementation are shown in Figure 4.5, in which the variables u_1 and u_2 represent the optimal control trajectory used as inputs obtained from the BIO-CS stand-alone implementation. The output variables y_1 and y_2 of the selected 2x2 (two-input-two-output) control structure obtained after simulating the process are also depicted in Figure 4.5. Note that the plant simulation model and the model used inside the BIO-CS framework for this case are identical. The BIO-CS stand-alone implementation is able to handle this scenario efficiently by taking both outputs of the system to the desired setpoints. Next, to address scenarios involving plant-model mismatch, the incorporation of the adaptive component into this framework is discussed.

Figure 4.5 Closed-loop BIO-CS simulation results: setpoint tracking – scaled outputs (y_1, y_2) [top], scaled inputs (u_1, u_2) [bottom]

4.3.3 BIO-CS + ANN Implementation

ANN structure for IGCC-AGR application

In this subsection, the specific details on the ANN-based adaptive component for the IGCC-AGR process application are discussed. The ANN structure considered for implementation for the IGCC-AGR process is outlined in Figure 4.6. In particular, the control system addressed here consists of two process input-output pairs represented by two neurons inside the network. As shown in Figure 4.6, each neuron has two ANN inputs that are processed in the hidden layer for generating the ANN outputs, which are eventually used as the adaptive control actions to augment the BIO-CS control laws as previously depicted in Figure 4.1. The selection of the number of inputs to each neuron

and the number of weights are based on user's choice. For this case study, 2 inputs and 5 weights are selected for each neuron based on achieved simulation performance.

Figure 4.6 Schematic representation of the ANN-based adaptive component employed in IGCC-AGR process application

The testing of the integrated framework is carried out under abnormal conditions that are simulated by altering the original A matrix of the state-space model. Additional error of 60% is introduced to the first element of the matrix A when simulating the developed state-space model for the BIO-CS with adaptive component implementation. Specifically, the first element of matrix A is changed from -0.0005 to -0.0002 for the MATLAB® plant simulation. The control results for the tracking performance of BIO-CS with adaptive component (ANN) are reported in Figure 4.7. In this Figure 4.7, when the abnormal conditions (including the matrix element errors) are imposed, the BIO-CS with adaptive component (ANN) application takes the appropriate control actions to compensate for the introduced challenges.

Figure 4.7 Closed-loop BIO-CS + ANN simulation results: setpoint tracking – scaled outputs (y_1, y_2) [top], scaled inputs (u_1, u_2) [bottom]

4.4 Conclusions

In this chapter, the BIO-CS algorithm (discussed in Chapter 3) was integrated with an ANN-based adaptive component for implementation on advanced energy systems. These systems are typically characterized by nonlinear and multivariable nature, and are represented here by the IGCC-AGR process with plant-model mismatch. In this study, data-driven and reduced process models for use in the agent-based BIO-CS were derived. The ANN-based adaptive component was developed to augment the baseline BIO-CS control laws in order to address the plant-model mismatch and provide additional compensation. This adaptive augmentation is also capable of increasing system robustness under more realistic scenarios. Therefore, the incorporation of the adaptive component into the biomimetic controller has been successfully accomplished.

5 Implementation of Biomimetic-based Method in DYNsIM[®] plant

5.1 Introduction

This chapter is focused on the implementation of the Biologically-Inspired Optimal Control Strategy (BIO-CS) with adaptive component developed in Chapter 4 for the DYNsIM[®] (software used for dynamic simulations of chemical processes) plant that represents the AVESTAR-WVU Center. This work is performed in collaboration with Dr. Perhinschi's group. The proposed strategy is implemented to address a subsystem of the IGCC-AGR process discussed previously. The proposed controller is designed in MATLAB[®] and the implementation of the control laws in the DYNsIM[®] plant is carried out using the MATLAB[®]-DYNsIM[®] link that is also described in this chapter. In particular, the implementation of the proposed framework for the IGCC-AGR process simulation in DYNsIM[®] is addressed considering setpoint tracking and plant-model mismatch cases. The details of the implementation of the developed approach for the IGCC-AGR subsystem is explained below.

5.2 Approach

5.2.1 BIO-CS Implementation Framework

The implementation of BIO-CS for the DYNsIM[®] plant simulation is explained in this subsection. For this implementation, the MATLAB[®]-DYNsIM[®] link developed at West Virginia University (WVU) is employed for establishing the communication between the BIO-CS controller (in MATLAB[®]) and the plant model (in DYNsIM[®]). Specifically, the connection between MATLAB[®] and DYNsIM[®] is established using an Open Platform Communications (OPC) server. A schematic representation of the BIO-CS controller framework for online implementation on the DYNsIM[®] plant simulation is shown in Figure 5.1. In this framework, a data-driven (e.g., autoregressive model with exogenous inputs, ARX) and reduced model derived from the original plant simulation serves as the BIO-CS controller model in MATLAB[®]. The BIO-CS algorithm presented in Chapter 3 is then followed for control trajectory (u_b) calculations over a sample time horizon. The generated

BIO-CS subroutine can be called from MATLAB[®] by the DYN[®] plant using the MATLAB[®]-DYN[®] link at each feedback sample time associated with the closed-loop implementation. Once the implementation is performed for a sample time, the generated outputs (y) are employed for the computation of the optimal control trajectories (u_b) for the consecutive time horizon. For the implementation of the BIO-CS control laws, the selection of the feedback sample time is critical for the controller performance. The feedback sample time is defined as the frequency in which a feedback signal is sent from the BIO-CS controller to the DYN[®] plant. This sample time should be long enough to allow the BIO-CS calculations, while the DYN[®] plant simulation is on hold. The BIO-CS computes optimal inputs trajectories in the form of data points that are discretized according to the DYN[®] engine time step and implemented using the MATLAB[®]-DYN[®] link, which is discussed in more details below.

Figure 5.1 Schematic representation of the BIO-CS controller framework in MATLAB[®] for online implementation in DYN[®] plant simulation

A high fidelity, nonlinear model of the IGCC-AGR process in DYN[®] software is used here. As briefly mentioned above, for simulation purposes, the MATLAB[®]-DYN[®] link is employed to perform the data exchange between MATLAB[®] and DYN[®] using object

linking and OPC data access protocol. This engine link uses a data mapping file that assigns each desired DYNsIM[®] point to an OPC data point and organizes those points into OPC groups, based on user selection. The data mapping file naming convention into OPC groups is considered while coding the control algorithms in MATLAB[®], in order to access the values of the DYNsIM[®] plant data points and send the commands to MATLAB[®] to run the controller. Once the computation of optimal control trajectories is completed in MATLAB[®], the control actions are converted to OPC data points that are accessible to the DYNsIM[®] simulation. The schematic representation of the MATLAB[®]-DYNsIM[®] engine link is illustrated in Figure 5.2 [61]. Specifically, for the implementation for the chosen application example, the feedback sample time is selected as 300 seconds based on extensive trial and error simulations. In this case, the BIO-CS parameters for the implementation are defined as pursuit time, $\Delta = 150$ sec, and discretization time, $\delta = 0.25$ sec. The value of δ selected for the controller design is the same as the DYNsIM[®] engine time step. These parameters are fixed for all the cases described in the result sections.

Figure 5.2 Schematic representation of the MATLAB[®] - DYNsIM[®] Engine Link

5.2.2 Integrated BIO-CS Framework with Adaptive Component

The schematic representation of the integrated BIO-CS framework with ANN-based adaptive component for online implementation in DYNsIM[®] is outlined in Figure 5.3. This framework is similar to the one presented in Chapter 4 with the difference that the control implementation is performed in the DYNsIM[®] plant simulation of IGCC-AGR process. For such implementation, the BIO-CS control actions, u_b , are augmented with the adaptive control laws, u_a , using user defined gains and executed via the MATLAB[®]-DYNsIM[®] link. The description on the design of the adaptive control laws/actions is provided in Chapter 4. These control actions are augmented with the baseline BIO-CS control laws and then implemented on the process simulation. In this case, before designing the BIO-CS controller with adaptive mechanism for the chosen system, it is first necessary to evaluate the output space for selecting setpoints that are achievable during process operation. The performed input-output state-space analysis is described next.

Figure 5.3 Schematic representation of the BIO-CS controller with adaptive component framework

5.3 Results & Discussions

5.3.1 Input-Output State-space Analysis

In this subsection, the input-output analysis associated with the selected control structure of the process is discussed. For this purpose, input-output operability analysis can be employed [73]. In the IGCC-AGR subsystem case, the available input and achievable output sets are subdivided in two categories as depicted in Figure 5.4. In Figure 5.4 (left), the outer space outlined by the green dashed line represents the space limited by the physical input constraints. The model is simulated in open loop considering the boundaries of this physical space. The values of the corresponding output space for the same region are the steady-state points obtained from these simulations (see green dashed lines in Figure 5.4, right). The inner spaces limited by the dashed blue color denote the identification spaces, i.e., when open-loop simulations are run considering the identification range of the input variables (see Figure 5.4, blue dashed lines). The operating point shown by the red triangle in Figure 5.4 is a nominal point considered to indicate a specific input-output relationship example in open loop. For the controller design purposes, the output setpoints can be selected from within the identification space built using the model identification limits or within the physical limit space. The case studies initially selected for BIO-CS implementation purposes are given in Table 5.1. Also, the corresponding operating points in the output space associated with the selected case studies are depicted in Figure 5.5.

Table 5.1 Selected case studies for implementation purposes

Case studies	$y_{1,sp}$	$y_{2,sp}$	Operating point location
Case I	2.8	41	Inside Identification Region
Case II	6.0	40	Outside Identification Region

*Id S: Model Identification Space *PL S: Physical Limit Space *OP: Operating Point

Figure 5.4 2-D representation of input-output spaces for open-loop simulations

As depicted in Figure 5.5 (left or right), the “+” sign represents the starting point for the simulation and denotes the current operating steady state for the IGCC-AGR process. The red triangle symbolizes the new steady state/setpoint that needs to be achieved by implementing the BIO-CS controller. The BIO-CS controller computes the optimal input trajectories that take the system from its current steady-state position, +, to the new desirable steady state, Δ .

Figure 5.5 Selected case studies for setpoint tracking: inside, case I (left), and outside, case II (right), of model identification region

5.3.2 BIO-CS Stand-alone Implementation Results

Case I (inside identification region)

This case considers an operating setpoint inside the identification region of the model used in the BIO-CS framework. Figure 5.6 shows the implementation results associated with Case I considering the stand-alone BIO-CS formulation. Additionally, a Proportional-Integral (PI) controller for the same control structure is designed for comparison purposes. The results of the PI implementation for this case study are also given in Figure 5.6. The y trajectories depicted in Figure 5.6 are the actual outputs from the DYNsIM[®] plant and the inputs, u , are the BIO-CS profiles computed in MATLAB[®] and then supplied to the DYNsIM[®] plant. As shown in Figure 5.6, the controllers are able to track the setpoints of both outputs successfully. In addition, the inherent plant-model mismatch that exists between the MATLAB[®] and DYNsIM[®] plant models is addressed effectively by employing the BIO-CS framework. Note that when comparing results, the BIO-CS controller is able to track the setpoints in a faster manner than the PI implementation. In particular, BIO-

CS takes on average 250 min to reach the setpoint for y_1 and 70 min in case of y_2 . On the other hand, the PI controller requires around 300 min to bring both outputs to their desired setpoints with slight offset in case of y_2 . The Integrated Squared Error (ISE) improvement for BIO-CS when compared to PI is observed to be 50% in case of y_1 and 85% in case of y_2 (also see Table 5.3, where a summary of results is presented).

Figure 5.6 Closed-loop BIO-CS and PI simulation results: setpoint tracking case I – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]

Case II (outside identification region)

Next, a case is simulated in which the operating setpoint is outside the identification region of the model used for BIO-CS stand-alone implementation. Such case imposes extra challenges to the BIO-CS controller such as an additional plant-model mismatch. As a

result, small offsets are observed for both outputs as shown in Figure 5.7. In this case, the controller struggles to keep the system at the desired operating point and additional oscillations are observed in process operation. The effect of such oscillations is quantified in terms of the ISE given in Table 5.3. This scenario provides the motivation for the proposed biomimetic adaptive controller approach. For such scenarios, the proposed BIO-CS framework with the adaptive component plays an important role in mitigating the additional plant-model mismatch. Using the information of the tracking errors, outputs, and/or available states, the developed adaptive controller is expected to bring the system back to the desired setpoint. The incorporation of the adaptive component into the BIO-CS framework is performed in the next subsection.

Figure 5.7 Closed-loop BIO-CS simulation results: setpoint tracking case II – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]

5.3.3 BIO-CS + ANN Implementation Results

Case II (Revisited)

The case study performed in the previous subsection is revisited to see if the performance could be improved by employing the proposed framework. In particular, Case II, in which the operating setpoint lies outside the identification region of the model used for BIO-CS stand-alone implementation is addressed. The design parameters associated with BIO-CS and ANN used for implementation purposes are summarized in Table 5.2. The results for the scenario with additional plant-model mismatch are depicted in Figure 5.8. The proposed BIO-CS framework with the adaptive component is able to bring the system back to the desired setpoints and enables faster setpoint tracking for both the outputs when compared to the BIO-CS stand-alone implementation. In particular, the proposed controller requires about 150 min and 80 min to bring y_1 and y_2 to their desired setpoints, respectively. On average, the developed strategy is 25 min and 2-3 min faster in terms of settling time for y_1 and y_2 , respectively, than the BIO-CS stand-alone case. From the results, it is also observed that the BIO-CS with adaptive component framework enables the reduction of tracking errors with respect to setpoint for both the outputs (with tracking error improvement of 11% for y_1 and 33% for y_2) when compared to the BIO-CS stand-alone implementation (see Table 5.3).

Table 5.2 BIO-CS and ANN parameters used for implementation purposes

BIO-CS parameters		ANN parameters	
Pursuit time, Δ	150 sec	Activation potential, β	1
Discretization time, δ	0.25 sec	Learning rates, $\gamma_w, \gamma_v, \lambda_w, \lambda_v$	0.1
Feedback sample time, T	300 sec	Number of inputs, q	2
Maximum number of agents, k_{max}	2	Number of weights, r	5
ITAE termination criteria, ε	0.1	Number of outputs, j	2

Figure 5.8 Closed-loop BIO-CS and BIO-CS + ANN simulation results: set-point tracking case II – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]

Improved case study (with new ANN tuning)

Case II is now simulated for faster setpoint tracking considering more aggressive tuning of the ANN-based adaptive component. In particular, the augmentation gains for the ANN-based adaptive mechanism are altered to achieve this goal. The gains associated with the ANN-based control action augmentation to the BIO-CS control moves are increased considering extensive trial and error simulations to achieve the desired performance. The results of the new implementation are depicted in Figure 5.9. In this case, faster setpoint tracking is achieved for y_1 with tracking error improvement in terms of ISE of 54% when compared to BIO-CS (see Table 5.3). However, note that the input trajectories generated

are not as smooth when compared to the previous case shown in Figure 5.8. Such tradeoff between faster setpoint tracking and smoothness in input profiles could be decided by the user according to the customized tuning employed. The oscillations in the output trajectories could be due to plant-model mismatch or the strong interactions among different variables and control loops associated with the process that are not controlled by the proposed framework. Such oscillations could be reduced by employing a controller model that further captures the process nonlinearities and/or by designing lower level control loops. Therefore, this case study shows that the proposed biomimetic adaptive controller provides itself as an alternative for addressing advanced energy systems applications.

Table 5.3 Summary of Integrated Squared Error (ISE) values for controller implementations in different case studies

Case I				
Integrated Error (ISE)	Squared	PI	BIO-CS	Improvement w.r.t. PI
y_1		1579.8	793.9	50%
y_2		261.8	38.5	85%
Case II				
Integrated Error (ISE)	Squared	BIO-CS	BIO-CS + ANN	Improvement w.r.t. BIO-CS
y_1		18851.2	16817.4	11%
y_2		249.3	168.1	33%
Case II (with new ANN tuning)				
Integrated Error (ISE)	Squared	BIO-CS	BIO-CS + ANN (new tuning)	Improvement w.r.t. BIO-CS
y_1		18851.2	8839.2	54%
y_2		249.3	243.6	0.02%

Figure 5.9 Closed-loop BIO-CS and BIO-CS + ANN simulation results: setpoint tracking with new tuning case II – outputs (y_1, y_2) [top], inputs (u_1, u_2) [bottom]

5.4 Conclusions

In this chapter, the implementation of the biomimetic-based control method on DYN[®] simulation of the IGCC-AGR process was performed successfully. In particular, the BIO-CS controller with ANN-based adaptive component was designed in MATLAB[®]. The MATLAB[®]-DYN[®] link that connects the two software packages, for the process simulation in DYN[®] and the biomimetic controller in MATLAB[®], was employed as a multiple-software automation environment. The results of the implementation of the

proposed approach demonstrated its performance improvement in terms of tracking errors and response time when compared to classical PI and stand-alone BIO-CS applications for different case studies. Thus, the proposed framework provides an integrated approach for biomimetic control with adaptive methods that can be used in a variety of process systems engineering applications.

6 Integration of Controller Design Method with Multi-agent Optimization Framework

6.1 Introduction

This chapter is focused on the development of a framework for the integration of a Biologically-Inspired Optimal Control Strategy (BIO-CS) with a multi-agent optimization approach. This work is performed in collaboration with Dr. Diwekar's group. In particular, the BIO-CS developed in Chapter 3 has shown to have unique features for handling process model nonlinearities as well as flexibility of employing different optimal control solvers and termination criteria when compared to traditional control methods. In case of optimization, techniques that imitate ant colony optimization with improved efficiency have been studied in the past under the name of Efficient Ant Colony Optimization (EACO) for molecular design and solvent selection case studies [74]. In addition, the abilities of heuristic-based methods such as EACO, Efficient Genetic Algorithm (EGA) and Efficient Simulated Annealing (ESA) were used to develop homogenous Multi-agent Optimization (MAO) techniques by establishing communication protocols between the algorithm procedures and the global information sharing environment [75]. However, the combination of biomimetic control strategies and agent-based optimization methods for nonlinear systems have not yet been addressed in an integrated fashion. In particular, in the context of process systems engineering, control studies are necessary to address setpoint tracking, disturbance rejection and plant-model mismatch challenges associated with process dynamics. Additionally, optimization plays an important role in identifying the optimal steady states or operating conditions for the processes that will satisfy the overall process objective (e.g., economic, productivity). To bridge this gap and combine process control and optimization techniques, in this chapter, BIO-CS is integrated with MAO to design a novel framework that leads to optimal dynamic process operations. The proposed combination results in a unique biomimetic framework for optimal control of nonlinear chemical processes. In summary, the developed framework yields optimal setpoints or a trajectory of setpoints for a nonlinear, multivariable system considering an

overall process objective by employing MAO. This system is then optimally controlled by BIO-CS to achieve the desired output setpoints.

The applicability of the proposed method is demonstrated using a fermentation process model [47] for bioethanol production. In particular, in this multivariable system, finding the optimal setpoint associated with production rate (or profitability) and the simultaneous control of product concentration and temperature of the fermentor are critical for optimal performance. The proposed framework is implemented for the fermentation process to address these challenges. Specifically, scenarios of setpoint tracking and plant-model mismatch are considered. The results of the developed method are compared to a gradient-based Sequential Quadratic Programming (SQP) technique [13] and a classical proportional-integral (PI) controller in terms of optimization and control studies, respectively. In this chapter, the proposed framework and its successful implementation for a case study are discussed in detail.

6.2 Approach

6.2.1 Proposed Integrated Framework

The proposed framework for the integration of the BIO-CS controller with the multi-agent optimizer considering process systems applications is shown in Figure 6.1. As depicted in this figure, for a given process, the MAO acts in a supervisory layer that considers the minimization or maximization of an overall objective for the whole process. As a result of this optimization, the optimal setpoints or trajectory of setpoints are obtained for the controlled/output variables in different control loops that represent sections of the process simulation. After this optimization, BIO-CS controllers are designed for the coupled control loops to take the process to these desired/optimal operating setpoints. In the particular case of optimization, the MAO search for decision variables in a solution space and then implement those solutions on the process. Note that the process model in this step is only employed to simulate the process and calculate the objective function values. The interaction between the MAO and the process simulation is depicted in Figure 6.1 (upper

part). The same model, or a reduced version of it, may be used for the design of the model-based BIO-CS controllers. Specifically, as shown in Figure 6.1 (lower part), the BIO-CS compute and implement optimal control laws on the process simulation for simultaneously tracking the multiple outputs of the process to their desired setpoints. For a given dynamic system model, it is assumed that the process control loops have been already identified through control structure selection techniques. These control loops or islands are thus simultaneously controlled using BIO-CS and integrated optimally through the MAO approach. The proposed framework is developed in MATLAB® by employing in-house MAO and BIO-CS algorithms. Also, the MATLAB® function *ode15s* is used for process simulation purposes. A schematic with the algorithm details associated with this integrated framework is depicted in Figure 6.2. The two main components of the proposed framework (MAO and BIO-CS) are discussed in the next subsections.

Figure 6.1 Schematic of the overall integrated framework of BIO-CS with multi-agent optimization

6.2.2 Multi-agent Optimizer (MAO)

The design of the multi-agent optimization approach for process systems applications is explained here in details. In particular, homogeneous MAO techniques are considered for implementation purposes in this chapter. Inside homogeneous MAO, multiple agents compute solutions for the optimization problem. The developed MAO routine involves the following steps (depicted in Figure 6.2, inside red dotted rectangle):

- (i) Select MAO algorithm from the available pool (EACO, EGA, ESA, and SQP) based on the user's choice (e.g., EACO);
- (ii) Define parameters for the algorithm initialization;
- (iii) Generate multiple agents (1, 2, ..., z) of the selected algorithm to obtain solutions for the decision variables by exploiting the capabilities of the chosen algorithm representing each agent;
- (iv) Simulate the process using the solution of the decision variables obtained from previous step for each agent and compute objective function values;
- (v) Share the information among the agents globally for coordination and comparison of the obtained solutions;
- (vi) Check the optimality criteria (e.g., tolerance for the objective function value difference at consecutive iterations);
 - 1. If satisfied, then the MAO converged to an optimal solution;
 - 2. If not satisfied, then repeat steps (ii)-(vi) by defining different parameters for the agents.

The obtained optimal solution corresponds to the setpoints or trajectory of setpoints for the outputs of the process that can then be used in the implementation of the BIO-CS controllers. Note that in this framework the agents/algorithms involved in the optimization are only dependent on the process model for the calculation of the objective function

values. The MAO parameters are independent of the process simulation. The model dependency particularly associated with step (iv) of the algorithm is further described next. The mathematical formulation of the optimization problem for process applications is given by,

$$\begin{aligned}
\min J &= \sum_{i=0}^k h(y_i, u_i) \\
\text{subject to, } &u^{lb} \leq u_i \leq u^{ub} \\
\text{in which, } &x_{i+1} = f(x_i, u_i) \\
&y_i = g(x_i, u_i)
\end{aligned} \tag{6.1}$$

in which J denotes the objective function that consists of the summation of the function of the i discrete controlled/output variables over the predefined time horizon with k symbolizing the number of discretization points. The solution of this nonlinear optimization yields the values of the y variable that are feasible to satisfy the optimized objective function for a given time horizon. The first step in the implementation for a process model consists of the selection of decision variables (u) and the number of intervals for discretization (k) for the time horizon. The number of intervals is chosen based on the tradeoff between desired computational efficiency and accuracy. The next step is to discretize the selected decision variable ranges based on the number of intervals. Then, the decision variable values at the discretization points are computed by the optimizer agents based on the algorithm capabilities. Subsequently, the values of the variables involved in the objective function are obtained from the process simulation by implementing the decision variable values computed by the agents at each discretization point. The objective function values at each discretization point are then added together to provide a cumulative J value which is further used in subsequent steps of the algorithm for information sharing and checking the optimality criteria. The mathematical details on the homogeneous MAO that utilizes the potential of multiple agents in terms of

coordination, parallelization and diversity by global information sharing can be found in the literature [75].

Figure 6.2 Schematic of the algorithm details for the integrated BIO-CS with MAO framework

6.2.3 BIO-CS Controller

In this subsection, a summarized version of the BIO-CS algorithm employed for process control applications is discussed briefly (see Chapter 3 for more details). The BIO-CS involves the following steps (also depicted in Figure 6.2, inside black dotted rectangle):

- (i) Start with initial conditions for a given dynamic process model and generate an initial feasible input trajectory (corresponding to an initial guess for agent0). Then select the BIO-CS agents' interaction parameters such as pursuit time (Δ), discretization time (δ) and sampling time (T);
- (ii) Specify the obtained trajectory as the leader agent trajectory;

(iii) Generate the follower agent trajectory by employing optimal control solvers (e.g., gradient-based solver) in an intelligent manner. The follower agent communicates with the leader by predefined algorithm parameters to compute its own trajectory;

(iv) Compute the Integrated Time Absolute Error (ITAE) for the follower trajectory over a user defined period of time and then check if this ITAE value lies within a certain threshold (ε);

1. If yes, the BIO-CS converged to an optimal control solution;
2. If no, then specify current follower trajectory as the next leader and repeat steps (ii)-(iv).

(v) Retain the optimal control/input profile from BIO-CS for implementation over a predefined sampling time horizon;

(vi) Simulate the process by employing the obtained control laws for a sampling time horizon and then send the feedback signal containing current outputs ($y(t)$) from the process to update the conditions for the next sampling period and close the loop.

For the implementation of BIO-CS on multivariable control loops, the optimal control problem is defined as follows:

$$\begin{aligned}
 \min_{u(t)} \varphi &= \int_{t_i}^{t_f} \left(\|y(t) - y_{sp}\|_Q^2 + \|u(t) - u^-(t)\|_R^2 \right) dt \\
 \text{such that, } \quad &\dot{x}(t) = f(x(t), u(t)) \\
 &y(t) = g(x(t), u(t)) \\
 \text{subject to, } \quad &x(t)^{lb} \leq x(t) \leq x(t)^{ub} \\
 &u(t)^{lb} \leq u(t) \leq u(t)^{ub}
 \end{aligned} \tag{6.2}$$

in which the optimal control objective function, denoted by φ , is minimized over a period of time. Here, t_i and t_f are the initial and final times, respectively. The descriptions of the other symbols are the same as in previous BIO-CS chapters. The objective function (φ)

generally consists of simultaneously maintaining multiple outputs at their desired setpoints, y_{sp} , with the addition of the input suppression term that considers past input moves, $u^-(t)$. These terms are minimizations of the squared errors between the variables and their desired/past values. Thus, the solution of this optimal control problem provides optimal trajectories for each input of the Multi-Input-Multi-Output (MIMO) system addressed to satisfy the objective function and constraints. Next, the case study for the implementation of the integrated framework on a nonlinear system considering a fermentation process as an example is explained.

6.3 Results & Discussions

6.3.1 Fermentation Process Case Study

To demonstrate the applicability of the proposed framework to a nonlinear chemical system, an extension of the fermentation process example presented in Chapter 3 is employed as the implementation case study. For the model extension, to prevent ethanol (end-product) inhibition and improve the productivity and efficiency of the fermentation process, an in situ ethanol-removal membrane is used so that the ethanol is removed as it is being produced. The extended mathematical model also takes into consideration the temperature effect on kinetics parameters, mass and heat transfer, in addition to the kinetic equations modified from the indirect inhibition structural model developed in the literature. In summary, the fermentation process model comprises of seven Ordinary Differential Equations (ODE) and two Algebraic Equations (AE) (see details in reference [47]). The main challenge of this system lies in the nonlinearities for control and optimization studies. For implementation purposes, a MIMO control structure that consists of a two-input-two-output system is chosen from this process. Selected model equations showing the input-output relationships relevant to this study are given below:

$$\frac{dC_p}{dt} = P \left(\frac{f(T)}{Y_{PX}} \right) \left(\frac{C_s C_e}{K_s + C_s} \right) + m_p C_x + D_{in} C_{p0} - D_{out} C_p - \frac{\alpha(C_p - C_{pm})}{V_F} \quad (6.3)$$

$$\begin{aligned}\frac{dC_{pm}}{dt} &= \frac{A_M P_M (C_p - C_{pm})}{V_M} + D_{m,in} C_{pm0} - D_{m,out} C_{pm} \\ \frac{dT_r}{dt} &= D_{in} (T_{in} - T_r) - f(C_x) f(T_r) + K_F (T_r - T_j) \\ \frac{dT_j}{dt} &= D_{j,in} (T_{w,in} - T_j) + K_j (T_r - T_j) \\ D_{m,out} &= D_{m,in} + \frac{A_M P_M (C_p - C_{pm})}{V_M \rho_r}\end{aligned}$$

in which, first two equations represent the mass balances for the product concentration on the fermentor side (C_p) and membrane side (C_{pm}), respectively. These equations also relate concentrations of other species involved in the reaction system in the fermentor, such as biomass (C_x), key component (C_e) and substrate (C_s). The next two equations show the energy balance in terms of temperature of the fermentor (T_r) and the jacket (T_j), respectively. The last equation is the algebraic equation considered in this fermentation process model in which the parameter $\alpha = A_M P_M$. The definitions of all the constants and parameters involved in this model and their nominal values were obtained from the literature [47] and are summarized in Table 6.1.

In this case, the membrane dilution rate, $D_{m,in}$, as well as the cooling water dilution rate in the jacket, $D_{j,in}$, are chosen as the manipulated variables for the regulation of ethanol concentration, C_{pm} , and fermentor temperature, T_r , respectively. The bound constraints on the manipulated variables are placed as $0 \text{ h}^{-1} \leq D_{j,in}, D_{m,in} \leq 1.5 \text{ h}^{-1}$. The selected control loops for the fermentation process representing multiple islands along with their integration using multi-agent optimization is depicted in the schematic in Figure 6.3. In the next subsection, the MAO analysis and the BIO-CS implementation results for setpoint tracking and plant-model mismatch scenarios are discussed.

Table 6.1 Base case set of constants and parameters used for the fermentation process model

Parameter	Description	Value
C_{p0}	Inlet product concentration (kg/m^3)	0.01
C_{pm0}	Inlet product concentration (kg/m^3)	0.01
D_{in}	Inlet fermentor dilution rate (h^{-1})	0.1
D_{out}	Outlet fermentor dilution rate (h^{-1})	0.1
$D_{m,in}$	Inlet membrane dilution rate (h^{-1})	0.5
$D_{m,out}$	Outlet membrane dilution rate (h^{-1})	0.5
$D_{j,in}$	Inlet cooling water dilution rate (h^{-1})	0.5
T_{in}	Inlet temperature of reactants (°C)	30
$T_{w,in}$	Inlet temperature of cooling water (°C)	28
K_S	Monod constant (kg/m^3)	0.2
K_F	Heat transfer constant (h^{-1})	1.8324
K_j	Heat transfer constant (h^{-1})	0.0714
m_p	Maintenance factor based on product ($kg/kg\ h$)	1.1
Y_{PX}	Yield factor based on product (kg/kg)	0.0526
V_F	Fermentor volume (m^3)	0.003
V_M	Membrane volume (m^3)	0.0003
P_M	Membrane permeability (m/h)	0.1283
A_M	Area of membrane (m^2)	0.24
ρ_r	Reactant density (kg/m^3)	1080
P	Maximum specific growth rate (h^{-1})	1.0

Figure 6.3 Schematic of the integrated framework with BIO-CS and multi-agent optimization for the fermentation process

6.3.2 Multi-agent Optimization Results

For the implementation of the multi-agent optimizer to address the fermentation process, the manipulated variables associated with the control loops, i.e., $D_{m,in}$ and $D_{j,in}$ are selected as the decision variables. Also, the time horizon for the optimization is chosen to be 20 h with the length of intervals of 4 h each. This results in 5 discretization points for each of the decision variables excluding the initial point. Inside the optimizer, agents are employed to calculate the values of these decision variables at each discretization point and implement those values in the process simulation using *ode15s* in MATLAB. From this process simulation, the objective function values as a function of the controlled and decision variables are computed for each corresponding discretization point. These discrete values are then combined to calculate the cumulative objective function value. For the implementation of the homogeneous MAO, an optimization problem is formulated considering an overall objective of maximization of production rate, J , which is related to the system profit, defined as follows:

$$\max J = C_p D_{out} V_F + C_{pm} D_{m,out} V_M \quad (6.4)$$

It is important to note that J is a function of three variables that are associated with state/decision variables of the system ($C_p, C_{pm}, D_{m,out}$). The implementation results of the homogeneous MAO technique with EACO, EGA and ESA as the selected algorithms for the fermentation process are summarized in Table 6.2. Note that each selected homogeneous MAO only considers agents with similar features, i.e., agents differ only in terms of the algorithmic parameters and the initialization. In this case, the number of agents are considered to be 5 for implementation purposes. These results are also compared to the stand-alone gradient-based SQP (employing *fmincon* in MATLAB) considering the same parameters and the result is also given in Table 6.2. The cumulative J values in case of heuristic-based MAO approaches are on average 5.5% higher than their SQP counterpart, which could result in significant economic benefits in the long run. The agents of each MAO technique search the solution space for decision variables extensively using their own capabilities that leads to the optimal value for the objective function. However, the computational time efficiency of the SQP implementation is higher due to the probabilistic sampling used for the solutions in case of heuristic-based methods vs. the directional search method employed in the gradient-based approach (SQP). Such longer computational time should not be an issue if the MAO is running offline multiple times or periodically during process operation, especially for this biochemical system with a time scale in order of hours. Given the performance vs. computational time tradeoff, the obtained optimal setpoint values from homogeneous MAO with multiple EACO as algorithmic agents are selected. Specifically, the controlled variable values obtained at the end of the optimization time horizon as shown in Table 6.2 are supplied as setpoints for the control studies. Note that the implementation results in Table 6.2 do not consider parallel computing of agents or heterogeneous MAO. Next, the closed-loop controller implementation results with these setpoints are discussed.

Table 6.2 MAO implementation results

Outputs	EACO	EGA	ESA	SQP
$C_{pm} (kg/m^3)$	36.87	37.00	36.90	37.01
$T_r(^{\circ}C)$	28.66	28.84	28.84	28.67
$J (kg/h)$	0.1252	0.1252	0.1252	0.1183
$CPU\ time\ (s)$	332.14	4941.82	6086.31	24.94

6.3.3 BIO-CS Closed-loop Results

Setpoint tracking

The implementation results of the BIO-CS controllers that are designed for the selected control loops are discussed here. The goal of the BIO-CS controllers is to take the system to the optimal setpoints obtained from the MAO calculations. The BIO-CS implementation results for the chosen control structure are shown in Figure 6.4. The BIO-CS parameters considered for this implementation are: pursuit time (Δ) of 1 h, discretization time (δ) of 0.1 h and threshold value (ε) of 0.1. The setpoints for C_{pm} of $36.87\ kg/m^3$ and T_r of $28.66^{\circ}C$ are selected from the results of homogeneous MAO with EACO as multiple agents. As depicted in Figure 6.4, BIO-CS provides optimal control trajectories that reach the desired output setpoints within 9 h for C_{pm} and 4 h for T_r successfully with smooth input profiles. The comparison of the obtained results with classical PI controllers are considered next. The PI controller results (obtained by extensive trial and error tuning) depicted in Figure 6.5 display slower and oscillatory response with slightly higher overshoot for C_{pm} compared to the BIO-CS implementation. In particular, the product concentration on the membrane side reach the steady-state shortly after the simulation time horizon of 20 h. The observed oscillations translate to production rate losses due to operation away from the optimal conditions. Specifically, the cumulative production rate calculated is approximately 15% higher for the BIO-CS implementation when compared to the PI controller. Thus, the BIO-CS implementation brings the system to its desired

setpoints in an optimal manner with reduced overshoot when compared to the PI controller performance.

Plant-model mismatch

The next case is simulated considering a plant-model mismatch scenario. In particular, the constant Y_{pX} is changed in the plant model, but not in the controller model thus affecting the process outputs as depicted in first equation of the process model. Specifically, the value of Y_{pX} is increased from 0.0526 to 0.0631 kg/m^3 which is approximately a 20% change from its original value.

Figure 6.4 BIO-CS simulation for setpoint tracking: (a) output (y_1); (b) output (y_2); (c) input (u_1); and (d) input (u_2) trajectories

This scenario essentially simulates the effect when increasing the yield factor based on product, affecting the product concentration of the fermentation process. Initially, without re-running the MAO, the BIO-CS controller with plant-model mismatch is implemented by using the setpoints from the previous case study. For this scenario, the cumulative production rate now considering the mismatch over the given time horizon is calculated to be approximately 0.1001 kg/h . Due to such mismatch, this production rate of the system is no longer optimal. Therefore, the MAO is re-run to obtain the optimal operating conditions that maximize the system production rate and then the BIO-CS is implemented to mitigate the effect of the model mismatch for providing optimal system performance for the new conditions. In practice, MAO would run periodically or even online depending on the process time scale. For homogeneous MAO with EACO agents, the values of the optimal setpoints for C_{pm} and T_r are obtained as 33.13 kg/m^3 and $28.67 \text{ }^\circ\text{C}$, respectively, with maximum cumulative production rate of 0.1109 kg/h .

Figure 6.5 PI simulation for setpoint tracking: (a) output (y_1); (b) output (y_2); (c) input (u_1); and (d) input (u_2) trajectories

The optimal cumulative production rate obtained from MAO is on average 10% higher than the case without re-running MAO, which would result in loss of productivity that translates into reduced profit. The BIO-CS implementation results shown in Figure 6.6 illustrate the successful controller performance. Therefore, the proposed integrated framework of BIO-CS with MAO is able to tackle the additional challenges imposed on the process effectively.

Figure 6.6 BIO-CS simulation for plant-model mismatch: (a) output (y_1); (b) output (y_2); (c) input (u_1); and (d) input (u_2) trajectories

6.4 Conclusions

In this chapter, the BIO-CS algorithm was integrated with MAO for implementation on nonlinear, multivariable processes to obtain optimal system performance. Specifically, a multivariable control structure derived from a nonlinear fermentation process example was addressed. The results of the homogeneous MAO considering an agent pool of heuristic-based algorithms such as EACO, EGA and ESA were compared to a gradient-based SQP method in terms of objective function value and computational time efficiency. In addition, BIO-CS control studies using the outcome of MAO were performed for the

process considering setpoint tracking and plant-model mismatch scenarios. The BIO-CS controller showed superior performance to that of the classical PI controller in terms of improved and faster responses. The performed studies provide an integrated approach for biomimetic agent-based control with optimization methods that can be employed in a variety of process systems engineering and energy applications.

7 Other Developments: BIO-CS Implementation on HYPER Process

7.1 Introduction

In this chapter, additional developments associated with the HYPER process at NETL were carried out in collaboration with Dr. Tucker's team. As outcomes of these efforts, a conference paper related to this chapter was published and a CRADA was written between WVU and NETL for future collaborations. Specifically, the energy system considered here for application purposes is derived from the HYbrid PERformance (HYPER) project at the National Energy Technology Laboratory (NETL) in Morgantown (WV). The HYPER project is comprised of one-of-the-kind facilities in the U.S. that were designed for the evaluation of future hybrid systems, including the development of modeling and control approaches focused on a gas turbine and fuel cell hybrid system. In particular, hybrid energy systems require advanced control techniques to address their unique challenges associated with strong interactions among different energy system components, without violating their operational constraints [26]. Specifically, transient coupling among diverse energy devices is present when such devices are combined into the same system. These challenges provide opportunities for the design of novel control strategies for hybrid power systems [27]. In the analyzed system configuration, the dynamic coupling of different energy components corresponds to the main control challenge that needs to be addressed. Previous studies were conducted to tackle different control problems encountered in the HYPER process such as failure mitigation in the emergency shut-down procedure and multi-input-multi-output coupling [29, 30, 31]. These studies were mainly focused on classical feed-forward control approaches and the characterization of the transient dynamics of the process. However, research studies on the control of the system in operation for setpoint tracking and disturbance rejection scenarios are scarce. The application of advanced control methods such as BIO-CS to address coupling effects without compromising the system performance is critical for the future of the HYPER project.

7.2 Results & Discussions

7.2.1 Hybrid Performance (HYPER) Process

The hybrid energy system application considered in this study is part of the HYPER process at the National Energy Technology Laboratory (NETL), in Morgantown, WV. This process consists of a cyber-physical fuel cell-gas turbine hybrid power system. The main advantage of this technology is to increase the total system efficiency by using less amount of fuel when compared to stand-alone gas turbine or fuel cell cycles. Figure 7.1 depicts the layout of the cycles and the components that are used in such hybrid configuration. As shown in Figure 7.1, the traditional gas turbine cycle design is modified to couple heat exchangers, pre and post-combustors and a fuel cell stack in the same system environment.

Figure 7.1 Schematic representation of the HYPER process

In the HYPER process, the gas turbine hybrid cycle is physically available, whereas the pre-combustor and the fuel cell stack are simulated by a model (validated based on a real system) [29]. The cathode inlet mass flow, temperature and pressure are fed into a 1-D fuel cell model while the turbine is running, and the gas turbine combustor is controlled by the fuel cell model using the calculation of the waste heat produced by the fuel cell

during operation. In the physical gas turbine system, the compressed air is warmed up by a heat exchanger using the exhausted air coming out of the turbine and then proceeds to a big size volume that reproduces the fuel cell cathode pressure drop. When the gas turbine combustor is not driven by the fuel cell model, the fuel valve can be independently controlled and used as a disturbance to simulate heat variations in the turbine [30]. In this work, we focus on the transfer function models derived from the physical gas turbine-fuel cell hybrid system. These models provide a realistic representation of the coupling between the actuator response and the real measurements of the system. The studies associated with the application of advanced control methods such as the proposed BIO-CS controller to these models are needed to facilitate the implementation of the controller to the actual system at NETL in the future. The control problem addressed here corresponds to a highly coupled scenario using two inputs and two outputs. The transfer function models depicted in Figure 7.2 are used for simulation and controller design purposes. These models are obtained from system identification techniques applied to the process. Inside the BIO-CS framework, the input-output transfer function model that serves as the controller model is converted to a continuous-time state-space model employing *tf2ss* function in MATLAB. The resulting plant and disturbance models addressed here consist of 4 states, 2 inputs and 2 outputs each. In this study, linear transfer functions are considered as the controller and the plant models. The implementation of the BIO-CS controller to address nonlinear and complex models that represent the HYPER plant is expected to be subject of future studies. The definitions of all variables used in the model are given in Table 7.1. It is important to note that the given plant model was obtained by identification around specific operating conditions, i.e., u_1 between 30 and 50 kW while u_2 between 40 % and 80 %. For simplification purposes, all variables are scaled between -1 and 1 during the controller design. In the next subsection, these variables are then rescaled to their original units for plotting purposes. Figure 7.2 (bottom) also shows the relationship between the disturbances and the plant outputs in the form of transfer functions. Note that the disturbance causing variables are

independent of the plant inputs. The disturbance model is used to simulate the effect of disturbances on process outputs directly. The differences between the plant and the disturbance models show the need of a state observer for controller design to address disturbance rejection scenarios. For such scenarios, as the BIO-CS is not aware of the disturbance model, the plant model inside the controller needs estimates of the current states that are affected by the disturbance for the calculation of the control moves.

Table 7.1 Base set of process variables and BIO-CS parameters

BIO-CS parameter	Description	Nominal Values
Δ	Pursuit time (sec)	1
δ	Discretization time (sec)	0.2
T	Sampling time (sec)	2
ε	Threshold value	0.1
t_{cpu}	CPU time (sec)	1.8
Process Variables	Description	Nominal Values
y_1	Turbine speed (rpm)	40,500
y_2	Cathode airflow (kg/s)	0.75
u_1	Electric load (kW)	40
u_2	Cold-air bypass (%)	40
u_3 (disturbance)	Fuel valve (%)	50
u_4 (disturbance)	Hot-air bypass (%)	10

$$\begin{aligned}
& \text{Transfer Function Plant Model} \\
& \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{-0.25 \cdot e^{-0.1 \cdot s}}{(s + 0.225)} & \frac{-0.065 \cdot e^{-0.5 \cdot s}}{(s + 0.125)} \\ \frac{-0.22 \cdot e^{-0.56 \cdot s}}{(s + 0.69)} & \frac{-1.43 \cdot e^{-0.7 \cdot s}}{(s + 1.43)} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\
& \text{Transfer Function Disturbance Model} \\
& \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{0.17 \cdot e^{-0.1 \cdot s}}{(s + 0.125)} & \frac{0.03}{(s + 0.46)} \\ \frac{0.06 \cdot e^{-0.56 \cdot s}}{(s + 0.16)} & \frac{-1.23 \cdot e^{-0.64 \cdot s}}{(s + 1.43)} \end{bmatrix} \cdot \begin{bmatrix} u_3 \\ u_4 \end{bmatrix}
\end{aligned}$$

Figure 7.2 HYPER process transfer function models

Figure 7.3 shows the block diagram for the BIO-CS implementation, in which the measured outputs from the process, considering the effect of the disturbance (y_m), feed the state observer for the calculation of state ($\hat{x}$) and output ($\hat{y}_m$) estimates for the controller. For the plant/process simulation purposes, *ode23s* function in MATLAB® is used to integrate the system by implementing the optimal control actions obtained from the BIO-CS calculations. The control and state estimation implementations are conducted at the same sampling frequency. For the dynamic optimization involved in the optimal control computations, the interval of discretization is 1 sec, which is the same as the pursuit time, 4, of the BIO-CS. In each interval, the number of collocation points used for the discretization are 3 and 1 for states and inputs, respectively. Also, $T = 2 \text{ sec}$, and the number of finite elements considered here are 2. All the simulation studies in this study were carried out on an Intel Core i7 (Sandy bridge) 3.40 GHz processor. In the next subsections, the open-loop simulation analysis along with the closed-loop BIO-CS implementation results for setpoint tracking and disturbance rejection scenarios are discussed.

Figure 7.3 Block diagram of BIO-CS implementation with observer

7.2.2 Open-loop Results

The open-loop dynamics of the HYPER transfer function models are analyzed here prior to the implementation of the controller. For the first set of simulations, open-loop scenarios are considered in which a step change of +10 kW (15 %) is imposed on the electric load, u_1 , as the manipulated variable, and the cold-air bypass valve, u_2 , is kept constant. Figure 7.4 depicts the output profiles of the turbine speed and cathode airflow for the open-loop simulations. As this system is highly coupled, the step change in u_1 primarily decreases the turbine speed, which in turn affects the cathode airflow. If the turbine speed is not properly controlled and thus a deviation occurs, a perturbation in the cathode airflow would also be observed. This effect corresponds to an indirect relationship between the electric load and the cathode airflow. The open-loop analysis for a step change of 20 % in cold air bypass valve, u_2 , while keeping the electric load, u_1 , constant was also studied but not presented here due to space limitations. For this scenario, the cathode airflow decreases substantially as opposed to the turbine speed. This open-loop study shows that the cold-air bypass primarily affects the cathode airflow. The turbine speed change that also takes place is only due to the coupling effect between the cold-air bypass and the turbine inlet temperature. Hence, changing the cold-air bypass to regulate the airflow for the fuel cell affects the turbine inlet temperature and the

turbine inlet temperature change affects the speed. The existence of strong coupling among these variables is undesirable in terms of system's performance and stability. Traditionally, to address such highly coupled system, two separate PID controllers need to be designed to control each of the outputs. However, previous studies show that the presence of the coupling causes conflict between the PID controllers, degrading the system's performance. Therefore, the mitigation of these issues motivates the need for advanced control of this system. Next, the effects of the disturbances on the plant outputs are also analyzed in open-loop. For this scenario, the fuel valve, u_3 , nominal value is changed as a step to reach 48.4 % (-1.6 % from current operating point) while u_4 is kept constant. The open-loop simulation results in this case are shown in Figure 7.5. The change in the fuel valve directly affects the turbine speed, which indirectly changes the cathode airflow, once again due to the coupling between these variables. Thus, minimizing the disturbance effects is an additional challenge for the controller design.

Figure 7.4 Open-loop simulation results: effect of step change on electric load - (a) outputs (y_1, y_2), and (b) inputs (u_1, u_2)

Figure 7.5 Open-loop simulation results: effect of step change on fuel valve - (a) outputs (y_1, y_2), and (b) inputs (u_1, u_2)

7.2.3 Closed-loop Results

Setpoint tracking

To address the challenges presented in the previous subsection, the proposed BIO-CS algorithm is implemented for the hybrid energy system. The closed-loop simulation results of the setpoint tracking case are depicted in Figure 7.6. In this case, the goal is to keep the turbine speed, y_1 , at the set point of 40,500 rpm and change the setpoint for the cathode airflow from 0.75 kg/s to 0.55 kg/s at 20 s and then back to 0.75 kg/s at 40 s. For this implementation, the algorithm parameters and the initial conditions (at 0 s) for each variable are defined in Table 7.1. As the first step in the algorithm, the agent0 trajectory for y_1 and y_2 are generated based on previous knowledge of the system. For y_1 , this

trajectory consists of a constant setpoint at 40,500 rpm (for the entire time horizon) and in case of y_2 , the agent0 trajectory corresponds to the intermediate setpoints at specified discretization times after the setpoint is changed (at 20 s), $t = \Delta + i\delta$, of 0.71 (at 21.2 s), 0.67 (at 21.4 s), 0.63 (at 21.6 s), 0.59 (at 21.8 s), and 0.55 (at 22 s) kg/s. Additionally, intermediate setpoints of 0.59 (at 41.2 s), 0.63 (at 41.4 s), 0.67 (at 41.6 s), 0.71 (at 41.8 s), and 0.75 (at 42 s) kg/s are considered on the way back of y_2 to 0.75 kg/s starting from 40 s. Given this problem setup, a series of agents whose dynamics are described by the same process model are simulated by employing the BIO-CS framework. The proposed framework is implemented to reach the new setpoint optimally with fast response time and reduced oscillations, which ultimately translates to improved system performance. As depicted in Figure 7.6, the BIO-CS controller successfully keeps the turbine speed at its desired value while changing the setpoints for cathode airflow by manipulating the inputs optimally. In particular, for the setpoint changes in cathode airflow, new stable conditions are achieved within 3 seconds without overshoot or actuator saturation, and the effect on the turbine speed due to coupling is limited to no more than 100 rpm or 0.2 %. Also, the actuators' moves are calculated according to the discretization time of the controller to allow the mitigation of oscillations and the coupling effects on the turbine speed. For this implementation, the computational times for the generation of optimal control laws are kept within one sample, $T = 2$ s, according to the description of the algorithm, due to the fast time scale of the hybrid system addressed in this study. In general, 1.8 s on average is taken for the computation of trajectories associated with 2 agents. Thus, the BIO-CS controller always employs here suboptimal solutions due to the computational time taken by the controller to calculate the optimal profiles within the considered sampling time.

Figure 7.6 Closed-loop simulation results: setpoint tracking case - (a) outputs (y_1, y_2), (b) input (u_1), and (c) input (u_2)

Disturbance rejection

The next case is simulated considering the introduction of a disturbance into the process. In particular, the fuel valve, u_3 , is changed affecting both process outputs as shown in the open-loop analysis above. The BIO-CS is implemented to mitigate the effect of the disturbance, and thus minimize output setpoint deviations. Also, a state observer is designed to perform the estimation of the unmeasured states associated with the disturbance-output model, by using the output measurements that have the combined effects of disturbance and manipulated inputs. For simplification purposes, the observer is designed using standard pole placement functions in MATLAB®. The closed-loop simulation results for this case are shown in Figure 7.7. For this scenario, the goal is to keep both y_1 (at 40,500 rpm) and y_2 (at 0.75 kg/s) as close as possible to their desired

setpoints. In the BIO-CS implementation, the agent0 trajectory is considered to be the desired setpoint (entire time horizon) for both the outputs. Also, the BIO-CS algorithm parameters are assumed to be the same as in the previous case study (see Table 7.1). As depicted in Figure 7.7, the disturbance in the fuel valve is introduced at 30 s until about 40 s in the form of a step change of -1.6 % (same value as in the open-loop analysis), causing the outputs to move away from their setpoints during the disturbance time period. In such period, the BIO-CS controller detects the output deviations at each sampling time through the state observer and computes control actions to bring the system back to its original steady state. As expected from the open-loop studies, the turbine speed gets more affected by the disturbance than the cathode airflow. For this speed during the disturbance period, the offset w.r.t. the setpoint is reduced by about 500 rpm when compared to the open-loop simulation. The complete removal of this offset was attempted by a more aggressive tuning of the observer. However, such aggressiveness resulted in a more oscillatory behavior than for the presented case. This result suggests the need for further analysis on more advanced state estimation techniques. Also, the current oscillations observed for the turbine speed during the disturbance period are mainly due to the communication lag between the controller and observer during implementation (i.e., observer provides the states to the controller for calculations at the same time sample). The operation of the designed observer with faster sampling than the controller (due to its simple nature) will be investigated in the future. Once the disturbance phases out, the BIO-CS control laws stabilize the system by addressing the remaining coupling effects in the process.

Figure 7.7 Closed-loop simulation results: disturbance rejection case - (a) outputs (y_1, y_2), (b) input (u_1), (c) input (u_2), and (d) disturbances (u_3, u_4)

7.3 Conclusions

In this chapter describing other BIO-CS developments, the BIO-CS algorithm previously proposed was modified by incorporating computational time improvements to facilitate its implementation on hybrid energy systems. The developed algorithm was implemented on MIMO transfer functions derived from the fuel cell-gas turbine hybrid energy system, as part of the HYPER facilities at NETL. Setpoint tracking and disturbance rejection scenarios were considered. For the setpoint changes in cathode airflow, new stable conditions were realized within 3 seconds without overshoot or actuator saturation, and the effect on the turbine speed due to coupling was limited to no more than 100 rpm or 0.2 %. For the disturbance rejection case, the results showed offset (of about 500 rpm),

which was reduced when compared to the open-loop simulations. Also, oscillations in the turbine speed response (in the order of 200-300 rpm) were present due to the communication lag between the controller and observer. Therefore, the performed studies provide promising aspects and point out future challenges for biomimetic control of hybrid energy systems of the future.

8 BIO-CS as a Model Predictive Controller (MPC)

8.1 Introduction

This chapter presents the formulation of the proposed BIO-CS as a Model Predictive Controller (MPC). In particular, MPC is the most widely used advanced control technique because of its industrial acceptance and demonstrated capabilities to handle nonlinear and constrained multivariable systems [76]. MPC is a model-based control approach that also has potential to deal with objective functions of various natures. Books such as [77] and [76] provide exposure to a variety of typical MPC formulations and related practical / theoretical issues. In this chapter, the casting of BIO-CS as MPC is described in details along with the similarities and differences between BIO-CS and MPC in terms of their basic principles. In addition, the resulting formulation (BIO-CS as MPC) is compared to the agent-based BIO-CS approach in terms of computational time, time to reach steady state and tracking error. For this purpose, the fermentation process case study discussed in Chapter 6 is revisited for implementation.

8.2 Approach

The underlying principles behind BIO-CS (model-based, predictive) are similar to those of MPC methods. However, in the case of BIO-CS, the nonlinear optimal control problem is solved in an agent-based manner that leads to improved performance as system complexity increases in terms of various criteria such as tracking error and time to reach steady state. The similarities and differences between the proposed agent-based BIO-CS and typical MPC approaches are listed below.

8.2.1 Similarities between BIO-CS and MPC

1. Each agent in the BIO-CS framework behaves as an individual MPC, if considered a fixed setpoint trajectory for both methods.
2. The pursuit time (Δ) in the case of BIO-CS is similar to the prediction horizon (P) for an MPC formulation.

3. If the BIO-CS algorithm is terminated at an immediate follower, or agent1 ($a_{k=1}(t)$), and the corresponding input trajectory ($u_{k=1}(t)$) is implemented for the process, then the one-agent BIO-CS can be considered as an MPC. This holds if the same solver, typically gradient-based (e.g., *fmincon* in MATLAB®), is used to obtain the optimal input trajectory for the controller. In this case, the BIO-CS algorithm loses its agent-based nature that allows further improvement of the control trajectories at the benefit of computational time reduction.

8.2.2 Differences between BIO-CS and MPC

1. The agent-based or multi-agent nature of BIO-CS allows for continuous improvement of input/output trajectories by using a dynamic or evolving setpoint trajectory for its application as opposed to a fixed one at the setpoint value. Typically, a constant setpoint value is used during an MPC implementation. Such property requires the employment of a termination criterion for BIO-CS such as ITAE, ISE for the calculation of the trajectory for different agents connecting two different steady states of the system.
2. The prediction (P) and the control (M) horizons of MPC are usually different, while BIO-CS only has the pursuit time (Δ) as an equivalent parameter.
3. The possibility of parallelization of optimal control problems associated with each BIO-CS agent is expected to improve the computational time performance of the BIO-CS algorithm. In such cases, the follower agents would not have to wait for the computation of the complete leader's trajectory. Once the required information (corresponding setpoint) from the leader agent is made available, the follower could start its journey leading to improved computational time. Such parallelization is currently under investigation and is a recommendation for future work (see Chapter 9).

8.2.3 Casting BIO-CS as a Model Predictive Controller (MPC)

The schematics of the agent-based BIO-CS and BIO-CS as MPC algorithms employed for implementation purposes are shown in Figure 8.1 and Figure 8.2, respectively. These figures also show the BIO-CS leader-follower interactions for the given time horizon (T) as well as the pursuit time (Δ)/prediction horizon (P) and discretization/sample time (δ) considered for the controllers.

Figure 8.1 Schematic of agent-based BIO-CS framework

BIO-CS as MPC

Prediction horizon (P) = Control horizon (M) = Pursuit time (Δ)

Figure 8.2 Schematic of BIO-CS as MPC framework

From the comparison of Figure 8.1 and Figure 8.2, the features that differentiate BIO-CS as MPC from the original agent-based BIO-CS can be re-iterated and are listed as follows: (i) BIO-CS as MPC has only one agent and the obtained follower trajectory behaves as an individual MPC for implementation on the process plant simulation; (ii) a fixed setpoint value is given to the optimal control problem associated with BIO-CS as MPC implementation as opposed to the dynamic setpoint trajectory of BIO-CS; (iii) for the BIO-CS as MPC case, the prediction horizon (P) and the control horizon (M) are identical and selected as the pursuit time (Δ) of the BIO-CS algorithm. If these features mentioned above are true, then the BIO-CS algorithm can be cast as an MPC approach. The formulation of BIO-CS as MPC leads to improved computational time efficiency at the expense of tracking performance reduction. In the next section, the implementation

results of the BIO-CS as MPC strategy in comparison with the original agent-based BIO-CS framework are presented.

8.3 Results & Discussions

8.3.1 Closed-loop Results

Case I:

The extended fermentation process model described in Chapter 6 is used here for demonstration purposes. The process model consists of seven Ordinary Differential Equations (ODE) and two Algebraic Equations (AE) characterized by the mass balances for the substrate (C_s), biomass (C_x), key component (C_e) and the product concentrations at fermentor side (C_p) and membrane side (C_{pm}). This model also includes the energy balance in terms of temperature of the fermentor (T_r) and the jacket (T_j) (see Equations 3.2 and 6.3). For the controller design purposes, a two-input-two-output control structure is selected based on the studies shown in Chapter 6, in which the membrane dilution rate, $D_{m,in}$, as well as the cooling water dilution rate in the jacket, $D_{j,in}$, are chosen as the manipulated variables for the regulation of ethanol concentration, C_{pm} , and fermentor temperature, T_r , respectively. For the control studies, the same constraints are placed on both the manipulated ($0 \text{ h}^{-1} \leq D_{j,in}, D_{m,in} \leq 1.5 \text{ h}^{-1}$) and the output variables ($C_s, C_x, C_e, C_p, C_{pm} > 0 \text{ kg/m}^3$ and $T_r, T_j > 0^\circ\text{C}$) as discussed previously. Here, the goal of the controller implementation is to maintain the outlet product concentration, C_{pm} , and fermentor temperature, T_r , at a desired steady state, in this case 36.87 kg/m^3 and 28.66°C , respectively, by controlling the input variables optimally. For the simulations, the controller parameters for BIO-CS and BIO-CS as MPC are kept the same when possible for comparison as listed in Table 8.1.

Table 8.1 BIO-CS and BIO-CS as MPC parameters selected for implementation purposes

Parameter	BIO-CS	BIO-CS as MPC
Pursuit time (Δ) for BIO-CS or prediction (P) / control horizon (M) for MPC	1 h	1 h
Discretization / sample time (δ)	0.5 h	0.5 h
Threshold value (ϵ)	1	-
Time horizon (T)	2 h	2 h
Initial trajectory	Constant setpoint	Constant setpoint – fixed entire simulation time
Max agents (k)	10	1

Figure 8.3 and Figure 8.4 show the results of the implementations of the BIO-CS controller and BIO-CS as MPC, respectively, for the setpoint tracking scenario. These implementations are assessed in terms of the Integrated Squared Error (ISE) performance criterion and time to reach steady state with a summary of the results in Table 8.2. As depicted in Figure 8.3 (top), the product concentration for the BIO-CS, C_{pm} , and fermentor temperature, T_r , of the process reach the desired steady states at approximately 10.5 h and 2 h, respectively, with minimal oscillations. Also, the dilution rate profiles, $D_{m,in}$ and $D_{j,in}$, achieve steady-state values at around 20 h and 1.5 h, respectively. Similar performance is observed for the BIO-CS as MPC implementation as shown in Figure 8.4 with tracking errors (represented by ISE) increased by 18 and 0.8%, respectively for y_1 and y_2 when compared to the multi-agent BIO-CS (also see Table 8.2). The performance difference between the two formulations mostly occurs during the transient period (see for example magnitude of first peak in both figures). For the BIO-CS as MPC case, the computational time is reduced from approximately 0.4 min in BIO-CS to 0.2 min due to the single agent consideration when BIO-CS is cast as MPC.

Table 8.2 Summary of comparison between controller implementations for Case I

Comparison criteria	BIO-CS (Case I)	BIO-CS as MPC (Case I)	Improvement w.r.t. BIO-CS as MPC
Time to reach steady state (h) (y_1)	≈ 10.5	≈ 11	similar
Time to reach steady state (h) (y_2)	≈ 2	≈ 2	similar
Integrated Squared Error (y_1)	3164.52	3891.91	18%
Integrated Squared Error (y_2)	64.54	65.11	0.8%
Computational time for entire simulation (min)	≈ 0.4 (max agents = 10)	≈ 0.2 (max agent = 1)	-100% (or 2 times higher)

Figure 8.3 BIO-CS closed-loop simulation results for setpoint tracking (case I): output (top) and input (bottom) profiles

Figure 8.4 BIO-CS as MPC closed-loop simulation results for setpoint tracking (case I):
output (top) and input (bottom) profiles

Case II:

In the second case, a more challenging setpoint tracking scenario is simulated to test the performance of both controllers. In particular, the setpoint for y_1 is changed by approximately 40% from its operating steady state at every 50 h for the simulation horizon of 250 h. In addition, a setpoint step change of 5% is given for y_2 at 50 h and 100 h during the implementation. Such scenario simulates the more challenging dynamic operation of a process to obtain products with different concentrations associated with a highly nonlinear process model. Note that the upper bound constraint on the input variable, $D_{m,in}$, is relaxed to accommodate such dynamic operation of the fermentation

process. The results for the implementation of BIO-CS and BIO-CS as MPC are shown in Figure 8.5 and Figure 8.6, respectively. For such implementation, the algorithm parameters are kept the same as for Case I (see Table 8.1). It is observed that the times to reach steady state for the output variables are approximately the same once again for both controller implementations. Also, there is an improvement in ISE for both the outputs in case of BIO-CS when compared to the BIO-CS as MPC implementation (see Table 8.3). The agent-based nature of the BIO-CS enables such improvement in trajectories mainly during the transient parts of the simulation. Once again, the computational time is shorter for the BIO-CS as MPC formulation than the BIO-CS itself.

Table 8.3 Summary of comparison between controller implementations for Case II

Comparison criteria	BIO-CS (Case II)	BIO-CS as MPC (Case II)	Improvement w.r.t. BIO-CS as MPC
Integrated Squared Error (y_1)	4.9209×10^7	9.1321×10^7	46%
Integrated Squared Error (y_2)	2.0631×10^7	3.5152×10^7	41%
Computational time for entire simulation (min)	≈ 3 (max agents = 10)	≈ 1.5 (max agent = 1)	-100% (or 2 times higher)

Figure 8.5 BIO-CS closed-loop simulation results for setpoint tracking (case II): output (top) and input (bottom) profiles

Figure 8.6 BIO-CS as MPC closed-loop simulation results for setpoint tracking (case II): output (top) and input (bottom) profiles

8.4 Conclusions

This chapter showed how to cast BIO-CS as MPC and described the uniqueness of the proposed BIO-CS controller in comparison with typical MPC methods available in the literature. The implementations of the original BIO-CS and BIO-CS as MPC methods for the nonlinear fermentation process model showed that the BIO-CS has improved performs due to its agent-based nature when compared to BIO-CS as MPC approach in terms of ISE criterion. It is also concluded that there is a tradeoff between the

computational time and desired tracking error performance of the system for these controller implementations. In addition, due to its flexibility, BIO-CS allows the possibility of parallelization of the optimal control problems associated with different agents as well as the exploration of model-free control as future research topics.

9 Overall Conclusions and Recommendations

In summary, a Biologically-Inspired Optimal Control Strategy (BIO-CS) was developed in this work to tackle nonlinear, multivariable energy systems. In particular, the IGCC-AGR process with carbon capture, a fermentation process for bioethanol production, and the HYPER process were addressed. The performance of the proposed approach was compared with other methods in the literature. In addition, an ANN-based adaptive component was incorporated into the developed BIO-CS framework to handle plant-model mismatch scenarios that are encountered in practice. Also, multi-agent optimization techniques were integrated with the proposed controller algorithm to provide optimal solutions for dynamic systems. The performed studies thus provide novel approaches for biomimetic agent-based control with data-driven and optimization methods that can be employed in a variety of process systems engineering and energy applications.

Following are the recommendations for future consideration:

- Studies could be carried out for further BIO-CS algorithm development towards performance and computational time improvements. In particular, the agent-based nature of the proposed algorithm could be investigated in details for further advancements. For example, the computational time performance of the biologically-inspired methods could be improved by examining the parallel computation of agents' trajectories;
- The developed biomimetic controller formulation could be changed for improved performance and compared with other existing model-based controllers. Controller design methods with model-free nature could be pursued by investigating further data-driven (e.g., deep learning), multi-agent techniques and stochastic principles for incorporation into the biomimetic strategies. Also, theoretical studies could be performed in terms of deriving mathematical proofs for biomimetic control methods and their stability analyses;

- The biomimetic control methods could be implemented further to address online application to real/cyber-physical systems. For example, models of high level of complexity associated with the HYPER process at NETL or bio-processes could be addressed by employing such novel techniques. The investigation of more sophisticated state estimators (e.g. EKF, MHE) in conjunction with BIO-CS could also be carried out.
- Some of the additional approaches that would be suggested to make the algorithm solve faster for improving/optimizing the code structure are calling subroutines in the code in an efficient manner, employing multidimensional arrays, etc. In addition, the comparison with respect to linear MPC and nonlinear MPC (with various identified models) could be studied as future work.

10 References

- [1] U. E. I. Administration, "Today in energy," eia.gov, 2017.
- [2] D. Hristu-Varsakelis and C. Shao, "A bio-inspired pursuit strategy for optimal control with partially constrained final state," *Automatica*, vol. 43, pp. 1265-1273, 2007.
- [3] P. S. B. Finzell, *A stigmergic algorithm for solving inverse thermal systems. MS Thesis*, Ames, IA: Iowa State University, 2013.
- [4] D. Hristu-Varsakelis and C. Shao, "Biologically-inspired optimal control: learning from social insects," *International journal of control*, vol. 77, no. 18, pp. 1549-1566, 2004.
- [5] A. M. Bruckstein, "Why the ant trails look so straight and nice?," *The Mathematical Intelligencer*, vol. 15, no. 2, pp. 59-62, 1993.
- [6] SchneiderElectric, "SimSci DYNsim®," 2017. [Online]. Available: <http://software.schneider-electric.com/products/simsci/design/dynsim/>. [Accessed 01 12 2017].
- [7] G. Stephanopoulos and G. V. Reklaitis, "Process systems engineering: From Solvay to modern bio- and nanotechnology. A history of development, successes and prospects for the future," *Chemical Engineering Science*, vol. 66, pp. 4272-4306, 2011.
- [8] K. Klatt and W. Marquardt, "Perspectives for process systems engineering—Personal views from academia and industry," *Computers and Chemical Engineering*, vol. 33, pp. 536-550, 2009.
- [9] I. E. Grossmann and A. W. Westerberg, "Research Challenges in Process Systems Engineering," *AIChE Journal*, vol. 46, no. 9, pp. 1700-1703, 2000.

- [10] J. B. Rawlings, *Process Systems Engineering in the information age. 2040 Visions of Process Systems Engineering*, Boston, MA: Symposium for the Retirement of George Stephanopoulos at MIT, 2017.
- [11] F. L. Lewis, D. L. Vrabie and V. L. Syrmos, *Optimal control*, 3 ed., John Wiley, 2012.
- [12] D. E. Kirk, *Optimal control theory: an introduction*, Prentice Hall, 1970.
- [13] U. M. Diwekar, *Introduction to Applied Optimization*, Springer, 2008.
- [14] V. R. Ramirez and U. M. Diwekar, "Stochastic maximum principle for optimal control under uncertainty," *Computers and Chemical Engineering*, vol. 24, pp. 2845-2849, 2004.
- [15] V. R. Ramirez, F. N. Rivera, G. G. Alatorre and U. M. Diwekar, "Stochastic optimal control for the treatment of a pathogenic disease," *Computer Aided Chemical Engineering*, vol. 28, pp. 217-222, 2010.
- [16] V. R. Ramirez, S. H. Castro and U. M. Diwekar, "Time dependent uncertainties and optimal control," *Computer Aided Chemical Engineering*, vol. 18, pp. 763-768, 2004.
- [17] S. Ulas, U. M. Diwekar and A. Stadtherr M., "Uncertainties in parameter estimation and optimal control in batch distillation," *Computers and Chemical Engineering*, vol. 29, pp. 1805-1814, 2005.
- [18] P. T. Benavides and U. M. Diwekar, "Optimal control of biodiesel production in a batch reactor: Part I: Deterministic control," *Fuel*, vol. 94, pp. 211-217, 2012a.
- [19] P. T. Benavides and U. M. Diwekar, "Optimal control of biodiesel production in a batch reactor: Part II: Stochastic control," *Fuel*, vol. 94, pp. 218-226, 2012b.
- [20] L. Sridhar, "Global optimization of continuous fermentation involving *Zymomonas mobilis*," *Journal of Sustainable Bioenergy Systems*, vol. 3, pp. 64-67, 2013.
- [21] L. Sridhar, "Elimination of oscillations in fermentation process," *AIChE Journal*, vol. 57, no. 9, pp. 2397-2405, 2011.

- [22] D. Bhattacharyya, R. Turton and S. E. Zitney, "Steady-state simulation and optimization of an integrated gasification combined cycle power plant with CO₂ capture," *Industrial & Engineering Chemistry Research*, vol. 50, no. 3, pp. 1674-1690, 2010.
- [23] D. Jones, D. Bhattacharyya, R. Turton and S. Zitney, "Plant-wide control system design: Primary controlled variable selection," *Computers & Chemical Engineering*, vol. 71, pp. 220-234, 2014a.
- [24] D. Jones, D. Bhattacharyya, R. Turton and S. Zitney, "Plant-wide control system design: Secondary controlled variable selection. .," *Computers & Chemical Engineering*, vol. 71, pp. 253-262, 2014b.
- [25] F. V. Lima, R. Amrit, M. Tsapatsis and P. Daoutidis, "Nonlinear model predictive control of IGCC plants with membrane reactors for carbon capture," 2013.
- [26] A. Traverso, D. Tucker and C. L. Haymes, "Preliminary experimental results of IGFC operation using hardware simulation," *ASME J. Eng. Gas Turbines Power*, vol. 134, no. 7, pp. 91-104, 2011.
- [27] N. Zhou, C. Yang, D. Tucker, P. Pezzini and A. Traverso, "Transfer functions development for control of fuel cell turbine hybrid systems," *Int: Journal of Hydrogen Energy*, vol. 40, pp. 1967-1979, 2015.
- [28] M. L. Ferrari, E. Liese, D. Tucker, L. Lawson, A. Traverso and A. F. Massardo, "Transient modeling of the NETL hybrid fuel cell/gas turbine facility and experimental validation," *Journal of Engineering for Gas Turbines and Power*, vol. 129, pp. 1012-1019, 2007.
- [29] P. Pezzini, M. L. Ferrari, D. Tucker and A. Traverso, "Research and educational opportunities in hardware-in-the-Loop simulation of advanced power systems: an International Perspective," 2014.
- [30] P. Pezzini, D. Tucker and A. Traverso, "Control strategy for a direct-fired fuel cell turbine hybrid power system," 2014a.

- [31] P. Pezzini, D. Tucker and A. Traverso, "Avoiding compressor surge during emergency shutdown hybrid turbine systems," *Journal of Engg, for Gas Turbines and Power*, vol. 135/102602, pp. 1-10, 2013.
- [32] F. V. Lima, D. Bhattacharyya, R. Turton, P. Mahapatra and S. E. Zitney, "Control of integrated gasification combined cycle power plants with CO₂ capture. The Impact of Control Technology," 2nd edition, T. Samad and A.M. Annaswamy (eds.), IEEE Control Systems Society, 2014.
- [33] T. Tsuji, M. Suzuki, N. Takiguchi and H. Ohtake, "Biomimetic control based on a model of chemotaxis in E.coli," *Artificial Life*, vol. 16, no. 2, pp. 155-177, 2010.
- [34] K. Passino, "Biomimecry of bacterial foraging for distributed optimization and control," *IEEE control system*, vol. 22, no. 3, pp. 52-67, 2002.
- [35] J. H. Lee and K. H. Lee, "Iterative learning control applied to batch processes: An overview," *Control Engineering Practice*, vol. 15, pp. 1306-1318, 2007.
- [36] N. Sanzida and Z. K. Nagy, "Iterative learning control of a batch cooling crystallization process based on linear time- varying perturbation models," London, UK, 2012.
- [37] C. Shao and D. Hristu-Varsakelis, "Cooperative optimal control:broadening the reach of bio-inspiration," *Bioinspiration & Biomimeics*, vol. 1, pp. 1-11, 2006.
- [38] G. Mirlekar, P. Pezzini, M. Bryden, D. Tucker and F. V. Lima, "A biologically-inspired optimal control strategy for hybrid energy systems," Seattle, WA, 2017a.
- [39] J. E. Cuthrell and L. T. Biegler, "On the optimization of differential-algebraic process systems," *AIChE Journal*, vol. 33, pp. 1257-1270, 1987.
- [40] S. A. Dadebo and K. B. McAuley, "Dynamic optimization of constrained chemical engineering problems using dynamic programming," *Computers & Chemical Engineering*, vol. 19, pp. 513-525, 1995.
- [41] J. S. Logsdon and L. T. Biegler, "Accurate solution of differential-algebraic optimization problems," *Chemical Engineering Science*, vol. 28, pp. 1628-1639, 1989.

- [42] J. S. Logsdon and L. T. Biegler, "Decomposition strategies for large-scale dynamic optimization problems," *Chemical Engineering Science*, vol. 47, no. 4, pp. 851-864, 1992.
- [43] M. Cizniar, M. Fikar and M. A. Latifi, "MATLAB DYNamic OPTimisation Code, Version 4.1," KIRP FCHPT STU, Bratislava, Slovak Republic, 2006.
- [44] I. M. L. Jobses, G. T. C. Egbertsa, K. C. A. M. Luyben and J. A. Roels, "Fermentation kinetics of *Zymomonas mobilis* at high ethanol concentrations; Oscillations in continuous cultures," *Biotechnology and Bioengineering*, vol. 28, pp. 868-877, 1986.
- [45] I. M. L. Jobses and J. A. Roels, "The inhibition of the maximum specific growth and fermentation rate of *Zymomonas mobilis* by ethanol," *Biotechnology and Bioengineering*, vol. 28, no. 4, pp. 554-563, 1986.
- [46] F. V. Lima, S. Li, G. Mirlekar, L. N. Sridhar and G. J. Ruiz-Mercado, "Modeling and advanced control for sustainable process systems," in *Sustainability in the Analysis, Synthesis and Design of Chemical Engineering Processes*, G. R. a. H. Cabezas, Ed., Cambridge MA, Elsevier, 2016, pp. 115-138.
- [47] S. Li, G. Mirlekar, G. J. Ruiz-Mercado and F. V. Lima, "Development of chemical process design and control for sustainability," *Processes [Online]*, vol. 4, no. 3, 2016.
- [48] H. Wang, N. Zhang, T. Qiu, J. Zhao, X. He and B. Chen, "Analysis of Hopf points for a *Zymomonas mobilis* continuous fermentation process producing ethanol," *Industrial & Engineering Chemistry Research*, vol. 52, no. 4, pp. 1645-1655, 2013.
- [49] A. Mahecha-Botero, P. Garhyan and S. S. E. H. Elnashie, "Nonlinear characteristics of a membrane fermentor for ethanol production and their implications," *Nonlinear Anal Real World Appl.*, vol. 7, pp. 432-457, 2006.
- [50] P. Garhyan and S. Elnashaie, "Static/dynamic bifurcation and chaotic behavior of an ethanol fermentor," *Industrial & Engineering Chemistry Research*, vol. 43, no. 5, pp. 1260-1273, 2004.

- [51] A. J. Daugulis, P. J. McLellan and J. Li, "Experimental investigation and modeling of oscillatory behavior in the continuous culture of *Zymomonas mobilis*," *Biotechnology and Bioengineering*, vol. 56, no. 1, pp. 99-105, 1997.
- [52] A. B. Jarzbski, "Modeling of oscillatory behavior in continuous ethanol fermentation," *Biotechnology Letters*, vol. 14, pp. 137-142, 1992.
- [53] O. L. Quintero, A. A. Amicarelli, G. Scaglia and F. Sciascio, "Control based on numerical methods and recursive bayesian estimation in a continuous alcoholic fermentation process," *BioResources*, vol. 4, no. 4, pp. 1372-1395, 2009.
- [54] S. Y. Huang and J. C. Chen, "Analysis of the kinetics of ethanol fermentation with *Zymomonas mobilis* considering temperature effect," *Enzyme and microbial technology*, vol. 10, pp. 431-439, 1988.
- [55] D. E. Seborg, T. F. Edgar and D. A. Mellichamp, "Teaching process control in the 21(st) Century: What has changed?," Denver, CO, 2003.
- [56] D. E. Seborg, "A perspective on advanced strategies for process control," *Modeling Identification and Control*, vol. 15, no. 3, pp. 179-189, 1994.
- [57] D. E. Seborg, T. F. Edger and S. L. Shah, "Adaptive control strategies for process control: A survey," *AIChE Journal*, vol. 32, no. 6, pp. 881-913, 1986.
- [58] C. T. Chen, "Direct adaptive control of chemical process systems," *Industrial & Engineering Chemistry Research*, vol. 40, pp. 4121-4140, 2001.
- [59] Y. Wang, F. Gao and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, pp. 1589-1600, 2009.
- [60] S. Wang, J. M. Simkoff, M. Baldea, L. H. Chiang, I. Castillo, R. Bindlish and D. B. Stanley, "Autocovariance-based plant-model mismatch estimation for linear model predictive control," *Systems & Control Letters*, vol. 104, pp. 5-14, 2017.
- [61] G. Al-Sinbol, *Monitoring and control framework for advanced power plant systems using artificial intelligence techniques. Doctor of Philosophy Thesis*, Morgantown, WV: West Virginia University, 2017.

- [62] P. Turner and J. Guiver, "Introducing the bounded derivative network-superceding the application of neural networks in control," *Journal of Process Control*, vol. 15, pp. 407-415, 2005.
- [63] P. Turner, J. Guiver and B. Lines, "Introducing the state space bounded derivative network for commercial transition control," Denver, CO,, 2003.
- [64] M. Perhinschi and G. Al-Sinbol, "Artificial dendritic cell algorithm for advanced power system monitoring," *International Review of Automatic Control*, vol. 9, no. 5, pp. 330-340, 2016.
- [65] M. Perhinschi, G. Al-Sinbol, D. Bhattacharyya, F. V. Lima, G. Mirlekar and R. Turton, "Development of an immunity-based framework for power plant monitoring and control," *Advanced Chemical Engineering Research*, vol. 4, no. 1, pp. 15-28, 2015.
- [66] F. Nardi and A. Calise, "Robust Adaptive Nonlinear Control Using Single Hidden Layer Neural Networks," Sydney, Australia, 2000.
- [67] D. Bhattacharyya, F. V. Lima, R. Turton, M. Perhinschi, U. M. Diwekar, B. Temitayo, G. Mirlekar, G. Al-sinbol and B. H. Gebreslassie, "Development of Integrated Biomimetic Framework with Intelligent Monitoring, Cognition and Decision Capabilities for Control of Advanced Energy Plants," Department of Energy, 2018.
- [68] L. Ljung and T. Glad, Modeling of dynamic systems, Prentice Hall, 1994.
- [69] L. Ljung, System identification: Theory for the user, Prentice Hall, 1987.
- [70] N. Holt, *Integrated Gasification Combined Cycle power plants*, 3rd Edition: Encyclopedia of Physical Science and Technology ,Academic Press, 2001.
- [71] T. Bankole, D. Jones, D. Bhattacharyya, R. Turton and S. Zitney, "Optimal scheduling and its Lyapunov stability for advanced load-following energy plants with CO2 capture," *Computers & Chemical Engineering*, vol. 109, pp. 30-47, 2018.

- [72] P. Mobed, J. Maddala, P. Pednekar, D. Bhattacharyya and Rengaswamy R., "Optimal sensor placement for fault diagnosis using magnitude ratio," *Industrial and Engineering Chemistry Research*, vol. 54, no. 38, pp. 9369-9381, 2015.
- [73] J. C. Carrasco and F. V. Lima, "Novel operability-based approach for process design and intensification: application to a membrane reactor for direct methane aromatization," *AIChE Journal*, vol. 63, no. 3, pp. 975-983, 2017.
- [74] B. H. Gebreslassie and U. M. Diwekar, "Efficient ant colony optimization (EACO) for computer aided molecular design: Case study solvent selection problem," *Computers and Chemical Engineering*, vol. 78, pp. 1-9, 2015.
- [75] B. H. Gebreslassie and U. M. Diwekar, "Homogenous multi-agent optimization for process systems engineering problems with a case study of computer aided molecular design," *Chemical Engineering Science*, vol. 159, pp. 194-206, 2017.
- [76] J. B. Rawlings, D. Q. Mayne and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, Madison, WI: Nob Hill Publishing, 2017.
- [77] E. C. Camacho and Bourdons C., *Model Predictive Control*, London: Springer Verlag, 2007.
- [78] MathWorks, "MATLAB function library: information on fmincon," 2017. [Online]. Available: <http://www.mathworks.com/help/optim/ug/fmincon.html>. [Accessed 28 4 2017].
- [79] M. Cizniar, M. Fikar and M. A. Latifi, "Dynopt document and code library," 2017. [Online]. Available: <http://www.kirp.chtf.stuba.sk/moodle/course/view.php?id=187>. [Accessed 28 4 2017].
- [80] T. Bankole and D. Bhattacharyya, "Algorithmic modelling of structural connectivity for process plants," Boston, MA, 2016.
- [81] G. Al-Sinbol and M. Perhinschi, "Generation of power plant artificial immune system using the partition of the universe approach," *International Review of Automatic Control*, vol. 9, no. 1, pp. 40-47, 2016.

- [82] H. K. Khalil, *Nonlinear Systems*, 3 ed., Upper Saddle River, NJ: Prentice-Hall, 2002.
- [83] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, pp. 733-764, 2002.
- [84] R. Amrit, J. B. Rawlings and R. Biegler, "Optimizing process economics online using model predictive control," *Computers and Chemical Engineering*, vol. 58, pp. 334-343, 2013.
- [85] G. Al-Sinbol, M. Perhinschi, P. Pezzini, K. Bryden and D. Tucker, "Investigation of biomimetic adaptive mechanisms for hybrid power plant control," *International Review of Automatic Control*, vol. 10, no. 5, pp. 390-398, 2017.
- [86] P. Daoutidis, M. Zachar and S. S. Jogwar, "Sustainability and process control: A survey and perspective," *Journal of Process Control*, vol. 44, pp. 184-206, 2016.
- [87] G. Mirlekar, S. Li and F. V. Lima, "Design and implementation of a Biologically-Inspired Optimal Control Strategy (BIO-CS) for chemical process control," *Industrial & Engineering Chemistry Research*, vol. 56, no. 22, p. 6468–6479, 2017.