Applications of Fractional Calculus In Chemical Engineering

uOttawa

By

Xin Shen

Thesis submitted to

the Faculty of Graduate and Postdoctoral Studies

in partial fulfillment of the requirements

for the Master of Applied Science

In

Department of Chemical and Biological Engineering

Faculty of Engineering

University of Ottawa

Abstract

Fractional calculus, which is a generalization of classical calculus, has been the subject of numerous applications in physics and engineering during the last decade. In this thesis, fractional calculus has been implemented for chemical engineering applications, namely in process control and in the modeling mass transfer in adsorption.

With respect to process control, some researchers have proposed fractional $PI^{\lambda}D^{\mu}$ controllers based on fractional calculus to replace classical PI and PID controllers. The closed-loop control of different benchmark dynamic systems using optimally-tuned fractional $PI^{\lambda}D^{\mu}$ controllers were investigated to determine for which dynamic systems this more computationally-intensive controller would be beneficial. Four benchmark systems were used: first order plus dead time system, high order system, nonlinear system, and first order plus integrator system. The optimal tuning of the fractional $PI^{\lambda}D^{\mu}$ controller for each system was performed using multi-objective optimization minimizing three performance criteria, namely the ITAE, OZ, and ISDU. Conspicuous advantages of using $PI^{\lambda}D^{\mu}$ controllers were confirmed and compared with other types of controllers for these systems. In some cases, a PI^{λ} controller was also a good alternative to the $PI^{\lambda}D^{\mu}$ controller with the advantage of being less computationally intensive.

For the optimal tuning of fractional controllers for each benchmark dynamic system, a new version of the non-dominated sorting genetic algorithm (NSGA-III) was used to circumscribe the Pareto domain. However, it was found that for the tuning of $PI^{\lambda}D^{\mu}$ controllers, it was difficult to circumscribe the complete Pareto domain using NSGA-III. Indeed, the Pareto domain obtained was sometimes fragmentary, unstable and/or susceptible to user-defined parameters and operators of NSGA-III. To properly use NSGA-III and determine a reliable Pareto domain, an investigation on the effect of these user-defined operators and parameters of this algorithm was performed. It was determined that a reliable Pareto domain was obtained with a crossover operator with a significant extrapolation component, a Gaussian mutation operator, and a large population. The findings on the proper use of NSGA-III can also be used for the optimization of other systems.

Fractional calculus was also implemented in the modeling of breakthrough curves in packed adsorption columns using finite differences. In this investigation, five models based on different assumptions were proposed for the adsorption of butanol on activated carbon. The first four models are based on integer order partial differential equations accounting for the convective mass transfer through the packed bed and the diffusion and adsorption of an adsorbate within adsorbent particles. The fifth model assumes that the diffusion inside adsorbent particles is potentially anomalous diffusion and expressed by a fractional partial differential equation. For all these models, the best model parameters were determined by nonlinear regression for

different sets of experimental data for the adsorption of butanol on activated carbon. The recommended model to represent the breakthrough curves for the two different adsorbents is the model that includes diffusion within the adsorbent particles. For the breakthrough experiments for the adsorption of butanol on activated carbon F-400, it is recommended using a model which accounts for the inner diffusion within the adsorbent particles. It was found that instantaneous or non-instantaneous adsorption models can be used. Best predictions were obtained with fractional order diffusion with instantaneous adsorption. For the adsorption of butanol on activated carbon Norit ROW 0.8, it is recommended using an integer diffusion model with instantaneous adsorption. The gain of using fractional order diffusion equation, given the intensity in computation, was not sufficient to recommend its use.

Résumé

Le calcul fractionnel, qui est une généralisation du calcul classique, a fait l'objet de nombreuses applications en physique et en ingénierie au cours de la dernière décennie. Dans cette thèse, le calcul fractionnel a été mis en œuvre pour des applications de génie chimique, à savoir dans le contrôle de processus et pour le transfert de matière dans un lit d'adsorption.

En ce qui concerne le contrôle des procédés, certains chercheurs ont proposé des contrôleurs d'ordres fractionnels $PI^{\lambda}D^{\mu}$ pour remplacer les contrôleurs PI et PID classiques. Le contrôle en boucle fermée de différents systèmes dynamiques de référence utilisant des contrôleurs $PI^{\lambda}D^{\mu}$ fractionnels optimisés a été étudié pour déterminer pour quels systèmes dynamiques ce contrôleur plus intensif en calcul serait bénéfique. Quatre systèmes de référence ont été utilisés: système de premier ordre avec retard pur, système d'ordre élevé, système non linéaire et système de premier ordre avec intégrateur. Le réglage optimal du contrôleur $PI^{\lambda}D^{\mu}$ fractionnel pour chaque système a été effectué en utilisant une optimisation multi-objectif minimisant trois critères de performance, à savoir l'ITAE, l'OZ et l'ISDU. Les avantages de l'utilisation des contrôleurs $PI^{\lambda}D^{\mu}$ ont été confirmés et comparés aux nombreux autres types de contrôleurs pour ces systèmes. Dans certains cas, un contrôleur PI^{λ} était également une bonne alternative au contrôleur $PI^{\lambda}D^{\mu}$ avec l'avantage d'être moins intensif en calcul.

Pour l'optimisation optimale des contrôleurs fractionnels pour chaque système dynamique de référence, l'algorithme génétique du tri des données non-dominés (NSGA-III) a été utilisé pour circonscrire le domaine de Pareto. Cependant, il a été déterminé que pour le réglage des contrôleurs $PI^{\lambda}D^{\mu}$, il était difficile de circonscrire le domaine de Pareto complet en utilisant NSGA-III. En effet, le domaine de Pareto obtenu était parfois fragmentaire, instable et/ou sensible aux paramètres et aux opérateurs de NSGA-III. Pour utiliser correctement NSGA-III et déterminer un domaine de Pareto fiable, une étude sur l'effet des opérateurs et des paramètres de cet algorithme a été réalisée. Il a été déterminé qu'un domaine de Pareto fiable a été obtenu avec un opérateur de croisement avec une composante d'extrapolation significative, un opérateur de mutation gaussien et une population de grande taille. Les résultats sur l'utilisation correcte de NSGA-III peuvent également être utilisés pour l'optimisation d'autres systèmes complexes.

Le calcul différentiel fractionnel a également été mis en œuvre pour la modélisation de la courbe de percée dans une colonne d'adsorption en utilisant la méthode des différences finies. Dans cette étude, cinq modèles basés sur différentes hypothèses ont été proposés. Les quatre premiers modèles sont basés sur des équations aux dérivées partielles d'ordre entier représentant le transfert de matière convectif du butanol à travers le lit garni et sa diffusion et son adsorption dans des particules d'adsorbants. Le cinquième modèle suppose que la diffusion à l'intérieur

des particules d’adsorbants est une diffusion potentiellement anormale et exprimée par une équation différentielle partielle d’ordre fractionnel. Pour tous ces modèles, les meilleurs paramètres du modèle ont été déterminés par régression non linéaire pour différents ensembles de données expérimentales pour l'adsorption du butanol sur le charbon activé. Le modèle recommandé pour représenter les courbes de percée pour les deux adsorbants différents est le modèle qui inclut la diffusion au sein des particules d'adsorbant. Pour l'adsorption du butanol sur le charbon activé F-400, il est recommandé d'utiliser un modèle avec adsorption non-instantanée ou un modèle avec un ordre fractionnel de diffusion avec adsorption instantanée. Pour l'adsorption du butanol sur charbon activé Norit ROW 0.8, il est recommandé d'utiliser un modèle de diffusion d’ordre entier avec adsorption instantanée. Le gain de l'utilisation de l'équation de diffusion d'ordre fractionnel pour les systèmes d’adsorption, compte tenu de l'intensité du calcul, n'est pas suffisant pour recommander son utilisation.

Statement of Contributions of Collaborators

I hereby declare that I am the sole author of this thesis and that no part of this work has been submitted or accepted for any other degree.

Dr. Jules Thibault is the supervisor of this thesis. He provided continual guidance throughout this work and made editorial comments and corrections to the written work presented. The responsibilities of the author, Xin Shen, in order to fulfill the requirements of this thesis, were as follows:

Chapter 2:

The computer codes for the fractional controllers were initially written in VBA and MATLAB languages by Xin Shen and then converted to Python computer language to allow using the mainframe computer at Queens University. The codes of NSGA-III in Python language was obtained from the open-source DEAP evolutionary computation framework collectively created by a group of developers on the platform Github. Xin Shen modified the NSGA-III codes in Python language to fit the requirements of research, analyzed the reasons of the incompleteness and instability of the Pareto domain, conducted experiments to determine the most appropriate settings of NSGA-III for the optimization of fractional controllers. The paper was written by Xin Shen with editorial comments from Dr. Thibault.

Chapter 3:

Xin Shen did planned and conducted all computational experiments to determine the Pareto domain for analysis. The analysis was performed using the Netflow ranking algorithm to determine the highest ranked solutions of the Pareto domain. The paper was written by Xin Shen with editorial comments from Dr. Thibault.

Chapter 4:

Conjunctly with my supervisor, five different models were derived and coded for the adsorption of butanol on activated carbon. The programming of these models was coded in Python by Xin Shen and in Fortran by Dr. Thibault. Isotherm and breakthrough curve data were obtained from the thesis of a former Ph.D. student, Niloofar Abdehagh and from the literature. The paper was written by Xin Shen with editorial comments from Dr. Thibault.

Acknowledgement

In the past two years, I have changed research topics two times, searching for the most powerful applications of fractional calculus in chemical engineering. The problems considered were more difficult and challenging than initially anticipated and were a good metrics for my perseverance. However, Dr. Thibault gave me support and inspiration when I was mired in optimization problem for half year and I almost thought that I could not figure out and solve the problem at that time. Thanks to his guidance and patience, I was able to identify the major obstacles, which were the causes of the problems encountered. Without the help of Dr. Thibault, I would not have been able to obtain the research results presented in this thesis. I would, therefore, like to express my deep and sincere appreciation to Dr. Thibault.

In this research, the Centre for Advanced Computing and Compute Canada provided clusters for advanced computation, which were truly instrumental for running time-consuming computer programs. To transfer my computer codes from VBA and MATLAB into Python and then to the supercomputers, Nabil Benabbou helped me writing the executable scripts that were necessary to execute my Python programs on the mainframe computers. The help from the personnel the Centre for Advanced Computing and Compute Canada is greatly appreciated.

To accelerate my programming, the open-source and multi-processing evolutionary computation framework DEAP was adopted. Thanks to the contributors of DEAP from all over the world who built an excellent tool for researchers. The help received from users on the different internet forums who answered my questions about multi-processing, Python and MATLAB is appreciated. Their assistance greatly helped me overcome many of the intricate technical problems that I encountered.

I would also like to thank my parents who always believed in my potential even when I gave up pharmaceutical engineering to transfer to chemical engineering. This transfer provided me the fantastic opportunity to broaden my outlook in science and engineering and allowed me accumulating valuable experience for my future life. I value and cherish this special experience, which is rarely accessible to most Chinese students. As the years go by, I really looking forward spending more time with my parents and give back to them all the love and help they deserved.

At the University of Ottawa, I have met many students of different backgrounds from many countries. I am fortunate to consider many of them as my friends. They have contributed to make me think differently and insightfully. Compared with three years ago, the deep connection with these wonderful friends contributed to carve my personality and add colors to my life. I am grateful to all my friends who shared their experience and stories with me.

Table of Contents

Abstract ii
Résumé iv
Statement of Contributions of Collaborators vi
Acknowledgement vii
Table of Contents viii
List of Figures xi
List of Tables xiv
Chapter 1 1
Introduction 1
1.1 Introduction 1
1.2 Research Objectives 4
1.2.1 Performance of Fractional $PI^{\lambda}D^{\mu}$ Controllers 4
1.2.2 Proper Choice of NSGA-III Optimization Parameters 4
1.2.3 Modelling of Adsorption Breakthrough Curves 5
1.3 Thesis Structure 5
1.4 References 6
Chapter 2 9
Multi-Objective Tuning of Fractional $PI^{\lambda}D^{\mu}$ Controllers Based on Pareto Domain Deconvolution and Net Flow Algorithm 9
2.1 Introduction 9
2.1.1 Background 9
2.2 Fractional Controller [1] 12
2.2.1 Fractional Controller 12
2.2.2 Numerical Method [10], [11] 13
2.2.3 Controller Performance Criteria 13
2.3 Benchmark Systems 14
2.3.1 First Order Plus Dead Time System (FOPDTS) 14
2.3.2 Higher-Order System (HOS) 15
2.3.3 Nonlinear System (NLS) 15
2.3.4 First Order Plus Integrator System (FOPIS) 15
2.4 Controller Tuning 15
2.4.1 Multi-Objective Optimization Problem 16
2.4.2 NSGA-III [13], [14] 17
2.4.3 Reduction of Fractional $PI^{\lambda}D^{\mu}$ Controllers 17
2.5. Analysis of the Pareto Domain 20

2.5.1 Net Flow Algorithm [17] 20
2.5.2 Pareto Domain Deconvolution 23
2.6. Results and Discussion 24
2.6.1 First Order Plus Dead Time System (FOPDTS) 24
2.6.2 Higher Order System (HOS) 33
2.6.3 Nonlinear System (NLS) 34
2.6.4 First Order Plus Integrator System (FOPIS) 36
2.7 Response of the Best Pareto-optimal Controllers for Different Systems 40
2.8 Conclusion 42
2.9 Nomenclature 43
2.10 References 45
Chapter 3 47
Practical Methodology to Initialize Multi-Objective Genetic Algorithm through Evolution History Analysis: A Case Study for the Tuning Fractional Controllers by NSGA-III 47
3.1 Introduction 48
3.2 Fractional Controller Multi-Objective Optimization Problem 49
3.2.1 Fundamental Concept of Multi-Objective Optimization 49
3.2.2 Fractional $PI^{\lambda}D^{\mu}$ Fractional Controller and Its Design [13] 50
3.2.3 Challenges to Circumscribe the True Pareto Domain 51
3.3 Important User-defined Factors to Initialize NSGA-III 53
3.3.1 Crossover Operator [6] 53
3.3.2 Mutation Operator 54
3.3.3 Population Size and Decision Space Size 55
3.3.4 Division of the Objective Space 56
3.4 Evolution History Analysis 57
3.5 Computational Experiments: Results and Discussion 57
3.5.1 Crossover 58
3.5.2 Mutation Rate and Mutation Operator 61
3.5.3 Population Size and Decision Space Size 64
3.5.4 Division of the Objective Space 65
3.6 Conclusion 66
3.7 Acknowledgement 67
3.8 Nomenclature 67
3.9 References 68
Chapter 4 71
Modelling Adsorption Breakthrough Curves 71
4.1 Introduction 72
4.2 Materials and Methods 73

4.2.1 Materials 73
4.2.2 Methods 73
4.3 Description of the different models 75
4.3.1 Problem statement 75
4.3.2 Different Mathematical Models 76
4.3.3 Initial and boundary conditions 80
4.4 Numerical Methods 81
4.5 Results and discussion 82
4.5.1 Experimental results with activated carbon F-400 82
4.5.2 Experimental results with activated carbon Norit ROW 0.8 87
4.6 Conclusion 91
4.7 Nomenclature 91
4.8 References 93
Appendix 95
Appendix 1 Ranking of the Pareto domains 95
Appendix 2 Dynamic Response for the Best Pareto-Optimal Fractional and Linear Controllers 103
Appendix 3 NSGA-III and Net Flow Algorithms 105

List of Figures

Chapter 2:

Figure 1	Control block diagram used to simulate the dynamic response of the closed-loop process	14
Figure 2	Multi-objective optimization problem for the tuning of the fractional $PI^{\lambda}D^{\mu}$ controllers	16
Figure 3	Different types of controllers in the decision space of fractional $PI^{\lambda}D^{\mu}$ controllers	18
Figure 4	Decision variable space of the normalized Pareto domain for the fractional $PI^{\lambda}D^{\mu}$ controller used to control the FOPDTS without noise. The color scheme is based on the RGB color model using the three normalized decision variables of Figure 4b.	25
Figure 5	Objective space of the normalized Pareto domain for the fractional $PI^{\lambda}D^{\mu}$ controller used to control the FOPDTS without noise. The color scheme is based on the RGB color model using the three normalized decision variables of Figure 4b.	26
Figure 6	Non-dominated solutions in the collection of independent optimization results from five different types of controllers	27
Figure 7	Non-dominated solutions deconvoluted from the collection of independent optimization for circumscribing the Pareto domains of different types of controllers	28
Figure 8	Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPDTS without noise	29
Figure 9	Variation in the three objectives as a function of the order of the integral component for a representative Pareto-optimal solution ranked in the top 5%	31
Figure 10	Ranked Pareto domain for the control of FOPDTS with noise: (a) normalized integral time as a function of the normalized controller gain; (b) normalized derivative time as a function of the normalized controller gain; (c) fractional integral order λ as a function of derivative order μ; (d) 3-D plot of the three objective functions.	32
Figure 11	Normalized integral time as a function of the normalized controller gain in ranked Pareto domain for the higher order system without noise	34
Figure 12	Normalized integral time as a function of the normalized controller gain in ranked Pareto domain for the nonlinear system without noise	35
Figure 13	Decision variable space and objective space of the normalized Pareto domain for the fractional $PI^{\lambda}D^{\mu}$ controller used to control the FOPIS without noise. The color scheme is based on the RGB color model using the three normalized decision variables of Figure 4b.	38

Figure 14	Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPDTS without noise	39
Figure 15	Different fractional order derivative of white noise	40
Figure 16	Dynamic response following a unit step change in the set point and when the best Pareto-optimal fractional and integer order controllers were used: (a) FOPDTS without noise; (b) FOPDTS with noise; (c) FOPIS without noise; (d) FOPIS with noise	42

Chapter 3:

Figure 1	Optimization problem for the tuning of the fractional $PI^{\lambda}D^{\mu}$ controller	51
Figure 2	Plot of the first two decision variables of the Pareto domain of the Fractional controller for the FOPDT. The color of numbered regions corresponds to the probability of occurrence	53
Figure 3	Offspring Distribution of Blend-α Crossover Operator	54
Figure 4	Schematic diagram to illustrate the creation of reference points and reference lines using the association operator	56
Figure 5	Evolution history analysis	57
Figure 6	Plot of two decision variables K_C-τ_I of the Pareto Domain obtained when the crossover parameter $\alpha = 0$ (a) and $\alpha = 0.5$ (b) with a constant mutation parameter ($\xi_m = 20\%$)	59
Figure 7	Plots of the history of the median values of K_C (a) and τ_I (b) through the population evolution process	59
Figure 8	Population distribution for $\alpha = 0$ (a) and $\alpha = 0.5$ (b) at the 9th generation of the optimization process for a population size of 10000	61
Figure 9	Plot of two decision variables K_C-τ_I of the Pareto Domain obtained when the crossover parameter $\alpha = 0.8$ with a constant mutation parameter ($\xi_m = 20\%$)	61
Figure 10	Pareto domain for two separate runs with a mutation rate parameter $\xi_m = 60\%$ and a crossover parameter $\alpha = 0.5$	63
Figure 11	The first two decision variables associated to the population at the 500th generation in search of the Pareto domain for a uniform distribution mutation for $\xi_m = 60\%$ (a) and $\xi_m = 100\%$ (b)	63
Figure 12	Plot of the first two decision variables K_C-τ_I of the Pareto domain obtained with a normal distribution mutation parameter $\xi_m = 60\%$ (a) and $\xi_m = 100\%$ (b) while the crossover parameter remained constant at $\alpha = 0.5$	64
Figure 13	Pareto domain based on a population of 10000 individuals with a larger decision variable space size (a) and a population of 1000 individuals with a smaller decision variable space size (b)	65

Chapter 4:

Figure 1	Schematic diagram of the adsorption setup used for (a) isotherm experiments and (b) breakthrough experiments	74
Figure 2	Packed bed filled with adsorbent porous particles	75
Figure 3	Breakthrough curves of butanol with AC F-400 for three different flow rates. The continuous lines and the symbols correspond to the simulated (using Model 3) and experimental breakthrough curves, respectively	83
Figure 4	Sum of squares of the errors (SSE) as a function of k_L and D_e (a) and k_{ADS} and D_e (b) for Model 3	88
Figure 5	Sum of squares of the errors (SSE) as a function of k_L and D_e for Model 4	88
Figure 6	Sum of squares of the errors (SSE) as a function of D_e and γ for Model 5	89
Figure 7	Experimental and simulated breakthrough curves for the adsorption of butanol on AC Norit Row 0.8. Lines and symbols represent the predicted breakthrough curves with Model 4 and 5 and the experimental breakthrough curve, respectively.	90

List of Tables

Chapter 2:

Table 1	Reduction conditions for which a fractional $PI^{\lambda}D^{\mu}$ controller is reduced to a simplified controller. Note that in this investigation K_c and τ_I cannot be zero or infinite	19
Table 2	Parameters of the Net Flow Algorithm for benchmark dynamic systems used in this investigation	24
Table 3	Different types of controllers identified in the whole Pareto domain and the top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the FOPDTS with and without noise	30
Table 4	Different types of controllers identified in the whole Pareto domain and top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the HOS with and without noise	34
Table 5	Different types of controllers identified in the whole Pareto domain and top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the NLS with and without noise	35
Table 6	Different types of controllers identified in the whole Pareto domain and top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the FOPIS with and without noise	37
Table 7	Parameters of the best Pareto-optimal fractional and integer order controllers, ranked by Net Flow, for the four benchmark systems without and with noise	41

Chapter 3:

Table 1	Different optimization parameters used to obtain Pareto-optimal solutions presented in each figure of Section 5	58
Table 2	Non-Dominated solutions for different numbers of divisions for the combined population of 40000 individuals	66

Chapter 4:

Table 1	Some properties of the two butanol adsorbents	73
Table 2	Model parameters giving the minimum SSE for breakthrough butanol adsorption on activated carbon F-400 for the three different flow rates	85
Table 3	Model parameters giving the minimum SSE for breakthrough butanol adsorption on activated carbon Norit Row 0.8	85
Table 4	Table 4 Model parameters determined by the method of least squares to minimize the SSE for experiments with activated carbon Norit Row 0.8	86

Chapter 1
Introduction

1.1 Introduction

This thesis is based on interdisciplinary research combining chemical engineering and fractional calculus. It is particularly focused on the application of fractional $PI^{\lambda}D^{\mu}$ controllers, on the multi-objective optimization of fractional $PI^{\lambda}D^{\mu}$ controllers, and on the modeling of adsorption breakthrough curves using partial fractional differential equations.

Classical integer calculus has been used for several centuries in various fields for a myriad of applications [1]. On the other hand, fractional calculus also has a very long history, which dates back to 1695 when Leibniz and L'Hôpital exchanged ideas on semi-order derivatives [2]. The earliest application of fractional calculus can be found in the field of physics, where mathematician Niels Henrik Abel first applied fractional calculus in the solution of the tautochrone problem in 1823 [2]. However, applications of fractional calculus were very scarce before 1974. The first specialized conference on fractional calculus in 1974 [3] spurred the interest of researchers and gave rise to an increasing number of applications in the past decades [4], [5]. The first applications of fractional calculus in chemical engineering were reported in 1974 in a monograph written by chemist K.B. Oldham and mathematician J. Spanier [5]. They investigated heat transfer and diffusion mechanisms in different geometries using semi-derivative and semi-integral equations [6]. Nowadays, the application of fractional calculus has been extended into different areas of applied science, different branches of engineering, and finance [5]. In chemical engineering, applications of fractional calculus are mainly in fractional control of process systems, viscoelasticity of materials, anomalous transport phenomena, and bioengineering [5].

There are many reasons for the introduction and the implementation of fractional calculus in science, engineering and finance even though classical calculus is still being used for most applications:

(1) The introduction of fractional operators can simplify complex equations which express special nonlinearity that rely on special functions such as Bessel function, Hankel function, Hermite function, error function, complementary error function, Wright function, and Mittag-Leffler function [7]. These special functions often take their origin in the solution of integer order partial differential equations containing implicitly special nonlinearities. To express these special nonlinearities in the solutions, mathematicians defined these special functions as tools to solve partial differential equations. However, fractional calculus has shown to be a neat alternative to generalize and evince naturally these special nonlinearities. Examples can be

found in distributed systems which can express some equations using error functions using a concise semi-order derivative expression, which even to eventually reveal the physical meaning.

(2) Some problems can be reformulated using fractional order and solved more easily resorting to the definition and properties of fractional calculus. A good example is Abel's solution of the tautochrone problem, where the equation can be reformulated by a semi-order integral and derivative and solved using properties of fractional calculus [6].

(3) Fractional calculus can lead to new and innovate models with special nonlinearity interpolated between integer order derivatives and integrals. Viscoelasticity and the fractional $PI^{\lambda}D^{\mu}$ controller are representative examples belonging to this category. Indeed, the fractional $PI^{\lambda}D^{\mu}$ controller considered in this thesis is a new controller that can replace the classical PID controller. The latter is comprised of three components: proportional, integral, and derivative. Both the integral and derivative components are first-order operator of the instantaneous error. If it would be desired to use a controller that provides an intermediate response between the first order operator of the instantaneous error and the instantaneous error itself, then a fractional order operator can be introduced, which can interpolate between first order and zero order derivative and/or integral. Thus, a fractional $PI^{\lambda}D^{\mu}$ controller takes the advantage of an interpolated response between the classical PID controller and P controller.

(4) Most importantly, fractional calculus can describe a series of phenomena, which have self-similarity characterized by power form functions in a microscopic point of view. In addition, fractional calculus can have heredity with time or nonlocality with space in a macroscopic point of view [8]. This abstract description can be illustrated by anomalous diffusion. From a microscopic point of view, the trajectory of a single particle or molecule in an anomalous diffusion is self-similar. The probability of a particle to change location or/and the time interval between two jumps for any given particle can be modeled by power-law distribution, which is also a self-similar function from a mathematical point of view. The corresponding macroscopic behavior, the anomalous transportation, relies on the whole history of the trajectory of all particles, not only their current state, which is different from classical diffusion [9], [10]. This historical behavior is referred to as heredity in physics, non-Markovian process in mathematics, memory effect when history refers to time, and nonlocality when history refers to position. Such behavior corresponds to the natural physical meaning of fractional calculus for its convolution form of the definition of fractional integral and fractional derivative. Unlike other convolution examples, mostly encountered in electronic engineering, the weight function for both the fractional derivative and the fractional integral are power function expressions where the power is related to the order of the operator. The weight function in convolution expression plays the role of memory function or influence function controlling the microscopic self-

similarity. These phenomena which have power regularities (relationship with power functions and power distribution) commonly exist everywhere, from human activities to natural cosmic phenomena [8], [11].

In control engineering, as an innovation of system and control theory, different researchers have proposed different controllers and systems based on fractional calculus. In 1945, Bode developed a controller for feedback amplifiers to maintain linearity and stable gain, which made use of a fractional order derivative term in the Laplace domain [12]. In 1958, Tustin created an open-loop transfer function with a fractional order integral term to control the position of massive objects [12]. The first attempt proposing a fractional order controller was made by Oustaloup, who proposed three generations of CRONE controllers (Commande Robuste d'Ordre Non Entier, meaning Non-integer-order Robust Control) [12], [13]. Inspired by CRONE controllers, Podlubny generalized the classical PID control strategy by interpolating between integer order operators and proposed fractional $PI^{\lambda}D^{\mu}$ controllers in 1994 [12], [14]. Through the past decades, better performance of fractional $PI^{\lambda}D^{\mu}$ controllers compared to classical PI and PID controllers has been reported for various applications [15]–[17]. To implement this improved controller for different systems, a number of tuning methods have proposed to obtain optimal controller parameters: genetic algorithm, particle swarm optimization, artificial bee colony, Ziegler-Nichols tuning rule, etc. [18]–[21]. Even though different tuning methods have been proposed, some important challenges still hinder its wider applications, such as the stability of fractional controller in closed-loop systems, and the availability of a fast and reliable online optimization procedure.

In transport phenomena, although fractional calculus can describe in a natural way the anomalous transportation, yet the first introduction of fractional calculus in transportation phenomena came relatively late. In 1905, Einstein first discover an important property of Brownian motion governing the classical diffusion equation proposed by Fick in 1855, that the variance of the displacement of particles experiencing Brownian motion is proportional to time [22], [23]. Later, others found that this relationship was only valid for some diffusion cases whereas for many other cases this relationship did not hold. Some examples of the latter are the transportation in porous systems [24], the transportation on fractal geometries [25], the diffusion of long-chain alkanes in zeolite [26], and the transportation of protein in cells [27]. These diffusion phenomena are referred to as anomalous diffusion or non-Fickian diffusion. To model anomalous diffusion, different methods have been proposed, including the generalized diffusion equations [28] and the continuous time random walk models (CTRW) [29]. Based on the CTRW, different functions in the Fourier-Laplace domain have been suggested a replacement of the proportional relationship in time domain. One function characterizing the observed anomaly, long resting periods between two consecutive jumps in a random walk, was

suggested by Balakrishnan in 1985 to describe this asymptotic relationship. Its corresponding governing diffusion equation is equivalent to the fractional partial differential equations. However, the author was not aware that he was introducing a fractional derivative to express such long-rest diffusion [30]. At the same time, Wyss and Schneider also introduced fractional partial differential equations as a novel tool to express anomalous diffusion without knowing that fractional calculus can characterize the long rest property and the infinite jump property of anomalous diffusion [9], [31], [32].

1.2 Research Objectives

1.2.1 Performance of Fractional PI$^{\lambda}$D$^{\mu}$ Controllers

The major objective of this work is to evaluate the performance of fractional PI$^{\lambda}$D$^{\mu}$ controllers tuned using the third generation of the non-dominated sorting genetic algorithm III (NSGA-III) for different benchmark systems. Even though the performance of fractional PI$^{\lambda}$D$^{\mu}$ controllers has been found to be superior for a number of systems, few challenges impeding its wider use are its higher complexity, its significantly larger computation time and the availability of simple tuning methods for this nonlinear controller. Considering the simplicity, maturity, and performance of the classical PID controller for a wide range of processes, it is not clear for which processes the fractional PI$^{\lambda}$D$^{\mu}$ controller could advantageously replace classical PID controllers. The control performance of fractional PI$^{\lambda}$D$^{\mu}$ controllers was assessed and compared with classical PID controllers using various representative benchmark systems. For each benchmark system, the fractional PI$^{\lambda}$D$^{\mu}$ controller was tuned using NSGA-III as the multi-objective optimization algorithm. The control performance of these benchmark systems was evaluated for cases with and without measurement noise. It is desired to make clear recommendations with respect to the types of processes that would most benefit from fractional PI$^{\lambda}$D$^{\mu}$ controllers.

1.2.2 Proper Choice of NSGA-III Optimization Parameters

In the numerous attempts to tune fractional PI$^{\lambda}$D$^{\mu}$ controllers using NSGA-III, some major challenges were encountered. Depending on the values chosen to set the user-defined factors in the NSGA-III, the Pareto domain was often instable, inaccurate and/or incomplete. The lack of a reproducible and inaccurate Pareto domain strongly influenced the initial conclusion based on the performance of fractional PI$^{\lambda}$D$^{\mu}$ controllers. To properly optimize the tuning of fractional PI$^{\lambda}$D$^{\mu}$ controllers using NSGA-III and to obtain a reliable and stable Pareto domain, a series of tests were performed to determine the best values of the key parameters and operators of NSGA-III. This study allowed determining the most appropriate crossover and mutation

operators of NSGA-III. These operators have a tremendous influence on the determination of the Pareto domain. The values of other key factors for NSGA-III were also studied.

1.2.3 Modelling of Adsorption Breakthrough Curves

For the special nonlinearity often encountered in adsorption and its influence on the adsorption breakthrough, conventional models cannot always represent adequately some of the diffusion within an adsorbent. The potentially anomalous diffusion inside porous adsorbent particles has an impact on the rate of transfer from the mobile phase to the adsorbent sites and on the breakthrough curves. It was therefore desired to examine if the diffusion and adsorption of species within the porous adsorbent could be better represented using partial fractional differential diffusion equations instead of the Fick's second law of diffusion for simple binary solutions.

1.3 Thesis Structure

This thesis is comprised with four chapters: the introduction and three scientific papers. Chapter 1 provides the background information on fractional calculus, enumerates some first applications on fractional calculus, discusses the reason for introducing fractional calculus as a mathematical tool for modelling, introduces the applications of fractional calculus in control engineering and anomalous diffusion that are the main subject of this thesis, and provides the main objectives of each research project comprising this thesis. Chapter 2 presents the comparison of the performance of fractional and classical PID controllers tuned using NSGA-III for different benchmark systems with and without noise and recommends the best controller strategy suitable for these benchmark systems. Chapter 3 presents the results of the numerical study that was conducted to properly set the parameters and factors of the genetic algorithm NSGA-III in order to completely define the Pareto domain. The procedure described in Chapter 3 is not only useful to determine accurately the Pareto domain for the tuning of fractional $PI^{\lambda}D^{\mu}$ controllers but could be extended to other optimization problems. Chapter 4 examines the potential advantages of using fractional calculus to represent the diffusion and adsorption of chemical species in porous adsorbent particles where possible anomalous diffusion may occur. Results are analyzed and compared for single-component curves.

1.4 References

[1] A. Rosenthal, “The History of Calculus,” *The American Mathematical Monthly*, vol. 58, no. 2, pp. 75–86, 1951.

[2] B. Ross, “A brief history and exposition of the fundamental theory of fractional calculus,” in *Fractional Calculus and Its Applications*, Springer, Berlin, Heidelberg, 1975, pp. 1–36.

[3] R. Hilfer, *Applications of Fractional Calculus in Physics*. World Scientific, 2000.

[4] T. Machado, V. Kiryakova, and F. Mainardi, “A poster about the recent history of fractional calculus,” *Fractional Calculus and Applied Analysis*, vol. 13, no. 3, p. 329p–334p, 2010.

[5] J. T. Machado, V. Kiryakova, and F. Mainardi, “Recent history of fractional calculus,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 3, pp. 1140–1153, Mar. 2011.

[6] K. B. Oldham and J. Spanier, *The Fractional Calculus: Theory and Applications of Differentiation and Integration to Arbitrary Order*. Dover Publications, 2006.

[7] R. L. Magin, *Fractional Calculus in Bioengineering*. Begell House Publishers, 2006.

[8] V. V. Uchaikin, *Fractional Derivatives for Physicists and Engineers: Volume I Background and Theory Volume II Applications*. Springer Science & Business Media, 2013.

[9] R. Metzler and J. Klafter, “The random walk’s guide to anomalous diffusion: a fractional dynamics approach,” *Physics Reports*, vol. 339, no. 1, pp. 1–77, Dec. 2000.

[10] W. Chen, H. Sun, X. Zhang, and D. Korošak, “Anomalous diffusion modeling by fractal and fractional derivatives,” *Computers & Mathematics with Applications*, vol. 59, no. 5, pp. 1754–1758, Mar. 2010.

[11] R. Klages, G. Radons, and I. M. Sokolov, *Anomalous Transport: Foundations and Applications*. John Wiley & Sons, 2008.

[12] P. Shah and S. Agashe, “Review of fractional PID controller,” *Mechatronics*, vol. 38, no. Supplement C, pp. 29–41, Sep. 2016.

[13] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu, *Fractional-order Systems and Controls*. London: Springer London, 2010.

[14] I. Podlubny, L. Dorcak, and I. Kostial, “On fractional derivatives, fractional-order dynamic systems and PI lambda;D mu;-controllers,” in *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997, vol. 5, pp. 4985–4990 vol.5.

[15] G. Altintas and Y. Aydin, “A comparison on genetic algorithm based integer order and fractional order PID control of magnetic bearing system,” in *2017 IEEE International Conference on Mechatronics (ICM)*, 2017, pp. 20–24.

[16] M. E. S. M. Essa, M. A. S. Aboelela, and M. A. M. Hassan, “A comparative study between ordinary and fractional order pid controllers for modeling and control of an industrial system based on Genetic Algorithm,” in *2017 6th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 2017, pp. 1–4.

[17] R. K. Jatoth, V. K. K, N. Bhookya, and G. Ramesh, “A comparative study on design and tuning of integer and fractional order PID controller,” in *Proceedings of 2014 International Conference on Modelling, Identification Control*, 2014, pp. 160–165.

[18] B. Bourouba and S. Ladaci, “Comparative performance analysis of GA, PSO, CA and ABC algorithms for ractional PI #x03BB;D #x03BC; controller tuning,” in *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*, 2016, pp. 960–965.

[19] A. Rajasekhar, A. Abraham, and M. Pant, “A Hybrid Differential Artificial Bee Colony Algorithm based tuning of fractional order controller for Permanent Magnet Synchronous Motor drive,” *Int. J. Mach. Learn. & Cyber.*, vol. 5, no. 3, pp. 327–337, Jun. 2014.

[20] D. Valério and J. S. da Costa, “Ziegler-Nichols type tuning rules for fractional PID controllers,” *Proceedings of IDETC 2005 (ASME)*, pp. 1431–1440, 2005.

[21] D. Valerio and J. S. da Costa, “A review of tuning methods for fractional PIDs,” in *4th IFAC Workshop on Fractional Differentiation and its Applications, FDA*, 2010, vol. 10.

[22] A. Einstein, *Investigations on the Theory of the Brownian Movement*. Courier Corporation, 1956.

[23] A. Fick, “Ueber Diffusion,” *Ann. Phys.*, vol. 170, no. 1, pp. 59–86, Jan. 1855.

[24] D. L. Koch and J. F. Brady, “Anomalous diffusion in heterogeneous porous media,” *The Physics of Fluids*, vol. 31, no. 5, pp. 965–973, May 1988.

[25] S. Havlin, D. Movshovitz, B. Trus, and G. H. Weiss, “Probability densities for the displacement of random walks on percolation clusters,” *J. Phys. A: Math. Gen.*, vol. 18, no. 12, p. L719, 1985.

[26] H. Jobic, A. Méthivier, G. Ehlers, B. Farago, and W. Haeussler, “Accelerated Diffusion of Long-Chain Alkanes between Nanosized Cavities,” *Angewandte Chemie International Edition*, vol. 43, no. 3, pp. 364–366, Jan. 2004.

[27] D. S. Banks and C. Fradin, “Anomalous Diffusion of Proteins Due to Molecular Crowding,” *Biophysical Journal*, vol. 89, no. 5, pp. 2960–2971, Nov. 2005.

[28] B. O’Shaughnessy and I. Procaccia, “Analytical Solutions for Diffusion on Fractal Objects,” *Phys. Rev. Lett.*, vol. 54, no. 5, pp. 455–458, Feb. 1985.

[29] J. Klafter, M. F. Shlesinger, and G. Zumofen, “Beyond brownian motion,” *Physics today*, vol. 49, no. 2, pp. 33–39, 1996.

[30] V. Balakrishnan, “Anomalous diffusion in one dimension,” *Physica A Statistical Mechanics and its Applications*, vol. 132, pp. 569–580, Sep. 1985.

[31] W. Wyss, “The fractional diffusion equation,” *Journal of Mathematical Physics*, vol. 27, no. 11, pp. 2782–2785, Nov. 1986.

[32] W. R. Schneider and W. Wyss, “Fractional diffusion and wave equations,” *Journal of Mathematical Physics*, vol. 30, no. 1, pp. 134–144, Jan. 1989.

Chapter 2
Multi-Objective Tuning of Fractional $PI^{\lambda}D^{\mu}$ Controllers Based on Pareto Domain Deconvolution and Net Flow Algorithm

Xin Shen, Jules Thibault*
Department of Chemical and Biological Engineering
161 Louis Pasteur, University of Ottawa
Ottawa (ON) Canada K1N 6N5

Abstract – In the past decade, increased attention has been devoted to the application of fractional $PI^{\lambda}D^{\mu}$ controllers. To compare the performance of fractional $PI^{\lambda}D^{\mu}$ controllers with simpler controllers, for different representative systems, multi-objective optimization algorithm NSGA-III has been adopted in this investigation to tune fractional $PI^{\lambda}D^{\mu}$ controllers and obtain the Pareto domain with the five controller parameters as the decision variables and three dynamic response metrics as objective functions. Considering that the Pareto domain obtained for $PI^{\lambda}D^{\mu}$ controllers is a composite of different simpler controllers, such as PI^{λ} and PID controllers, the Net Flow Algorithm and the Pareto domain deconvolution were adopted in this investigation to evaluate the performance of fractional $PI^{\lambda}D^{\mu}$ controllers and other simpler controllers. The Net Flow Algorithm, used to find a suitable trade-off based on decision makers' expertise, is employed to evaluate the performance of different types of controllers present in the Pareto domain. The composite Pareto domain may be separated into a series of simpler controllers via the Pareto domain deconvolution. Results show that, in comparison with other simpler controllers, the added degrees of freedom in fractional $PI^{\lambda}D^{\mu}$ controllers have largely extended the Pareto domain and led to superior performance for the four representative benchmark systems with or without noise. For some systems, the performance of fractional PI^{λ} controllers was as good as fractional $PI^{\lambda}D^{\mu}$ controllers.

***Keywords*:** $PI^{\lambda}D^{\mu}$ controllers, benchmark systems, multi-objective optimization, Pareto domain deconvolution, Net Flow Algorithm, NSGA-III

2.1 Introduction

2.1.1 Background

Inspired by fractional calculus, Podlubny proposed using a fractional $PI^{\lambda}D^{\mu}$ controller in 1994 [1]. A fractional $PI^{\lambda}D^{\mu}$ controller is a generalization of the integer order integration and

derivation of the error of classical PID (Proportional-Integral-Derivative) controllers to the fractional order integration and derivation. Introducing fractional calculus gives rise to two extra degrees of freedom: the fractional integral order λ and derivative order μ. Performance enhancements obtained with fractional $PI^{\lambda}D^{\mu}$ controllers have spurred the interest in the development of these controllers.

Because the definition of fractional $PI^{\lambda}D^{\mu}$ controllers is the generalization of classical PID controllers, fractional $PI^{\lambda}D^{\mu}$ controllers can be reduced to simpler forms of controllers for specific controller parameters. These simpler controllers include controllers with only integer order components, namely P, PI, PD, and PID controllers, and controllers with fractional order components, including PI^{λ}, PD^{μ}, $PI^{\lambda}D$ and PID^{μ} controllers. Das and Pan mentioned that the different members of the fractional $PI^{\lambda}D^{\mu}$ controller family have different performance in terms of noise amplification and response time. They mentioned that a larger fractional integration can amplify low-frequency variations and a larger fractional derivative can amplify high-frequency variations [2]. Because of the different performance of the members of fractional $PI^{\lambda}D^{\mu}$ controller family, investigations of different forms of fractional $PI^{\lambda}D^{\mu}$ controllers have been performed on different systems. For conciseness, only few of the reported applications for simple processes are presented in this section. Maheswari et al. found that the fractional $PI^{\lambda}D^{\mu}$ controller outperformed a PID controller for an SO_2 emission system modelled as a first order plus dead time process [3]. Zhang et al. used a PD^{μ} controller for the path tracking control system of tractors, modelled as a second order system plus repeated integrator system [4]. $PI^{\lambda}D^{\mu}$ controllers were evaluated by Jatoth et al. for first to fourth order systems and exceptional performance compared to PID controllers was observed [5]. PI^{λ} controllers have been implemented for the control of a permanent magnet linear synchronous motor servo system modeled by a fractional order system [6].

One method to tune fractional $PI^{\lambda}D^{\mu}$ controllers is through multi-objective optimization (MOO) algorithms. For most cases, MOO is used to obtain the Pareto domain for multiple objective functions that need to be optimized simultaneously. In the optimization of fractional $PI^{\lambda}D^{\mu}$ controllers, most research implicitly assumed that Pareto-optimal solutions are all fractional $PI^{\lambda}D^{\mu}$ controllers. However, as one can easily deduce, Pareto-optimal solutions are comprised of a mixture of $PI^{\lambda}D^{\mu}$ controllers and their simpler controller forms as mentioned earlier. For example, over 90% of the Pareto-optimal solutions for the control of a first order plus dead time system in our experiments were fractional PI^{λ} controllers instead of $PI^{\lambda}D^{\mu}$ controllers.

The fact that the Pareto domain of the $PI^{\lambda}D^{\mu}$ controllers is a mixture of different types of controllers should be considered in the analysis of the Pareto domain. In some research papers, interpolation of Pareto-optimal solutions was performed in order to analyze and plot the Pareto

front of fractional $PI^{\lambda}D^{\mu}$ controllers [7]–[9]. However, by considering that Pareto domain may include other simpler forms of controllers belonging to the $PI^{\lambda}D^{\mu}$ controller family, interpolation of Pareto-optimal solutions can lead to other types of controllers instead of fractional $PI^{\lambda}D^{\mu}$ controllers. For the same reason, the method based on interpolation, called Nash solution, is not suitable for these cases [9]. In addition, according our experience, it is not guaranteed that the interpolated solution will satisfy the Pareto-optimality in some cases.

It is therefore required to develop a more robust method to analyze the Pareto domain for the tuning of fractional $PI^{\lambda}D^{\mu}$ controllers in order to select the optimal controller structure and its associated parameters. It is also important to consider which member of the $PI^{\lambda}D^{\mu}$ controller family will be the most appropriate for a specific system. In this investigation, the third generation of the non-dominated sorting genetic algorithm III (NSGA-III) based on reference lines has been used to circumscribe the Pareto domain. Four representative systems, with and without measurement noise, have been tested. In this investigation, after the Pareto domain has been circumscribed using NSGA-III, the Net Flow Algorithm is used to rank all Pareto-optimal solutions and the Pareto domain is deconvoluted into the different types of controllers belonging to the $PI^{\lambda}D^{\mu}$ controller family. The Net Flow Algorithm attributes fitness scores to all Pareto-optimal solutions that are then used to rank the entire Pareto domain, thus allowing to determine the types of controllers that were highly ranked and their frequency of occurrences.

The main highlights of this paper include:

- The deconvolution of the Pareto-optimal fractional $PI^{\lambda}D^{\mu}$ controllers into individual members of the fractional $PI^{\lambda}D^{\mu}$ controller family is performed for different dynamic systems with the objective of selecting the most appropriate structure and its associated parameters for each system.
- The Net Flow Algorithm and the deconvolution of the Pareto domain into individual controller structures are adopted in the analysis of the Pareto domain.
- The performance of $PI^{\lambda}D^{\mu}$ controllers and its constituting members for four different representative dynamic systems is assessed in the presence and the absence of measurement noise.
- Numerical experiments were based on a large population of 10000 and on a large generation of 1000 in order to clearly circumscribe a well-defined Pareto domain.

The rest paper is organized as follows: Section 2.2 introduces the definition and the numerical method of fractional $PI^{\lambda}D^{\mu}$ controllers, and the performance criteria used in this investigation. Section 2.3 presents four representative dynamic systems selected to assess the performance and benefits of $PI^{\lambda}D^{\mu}$ controllers. Section 2.4 briefly presents the fundamentals of multi-objective optimization, the NSGA-III algorithm, and the $PI^{\lambda}D^{\mu}$ controller reduction. Section 2.5

provides a brief description of the Net Flow Algorithm and the Pareto domain deconvolution used to analyze Pareto-optimal solutions. Section 2.6 presents a discussion and analysis of the performance of $PI^{\lambda}D^{\mu}$ controllers of different systems. Finally, Section 2.7 concludes by specifying the best type of controllers that is recommended for each benchmark system.

2.2 Fractional Controller [1]

2.2.1 Fractional Controller

Fractional integral and fractional derivative defined in Eqs. (1) and (2) respectively, are a generalization of the integral and the derivative from an integer order to an arbitrary order. For a fractional derivative operator, the negative derivative order is equivalent to the fractional integral, and vice versa for the fractional integral operator. As a limiting case, zero-order derivative or integral of a function equals to the function itself.

$$ {}_aD_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t-s)^{\alpha-1} f(s)\, ds, \quad t > a, \quad \alpha > 0 \tag{1} $$

$$ {}_aD_t^{\alpha} f(t) = \frac{d^n}{dt^n} {}_aD_t^{-(n-\alpha)} f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \left[\int_a^t (t-s)^{n-\alpha-1} f(s)\, ds \right], \quad t > a \tag{2} $$

where *n* is an integer satisfying $n-1 \le \alpha < n$.

A fractional $PI^{\lambda}D^{\mu}$ controller is a novel type of controller based on fractional integral and derivative that is currently attracting significant interest because of the good performances obtained in some applications. The $PI^{\lambda}D^{\mu}$ fractional controller is defined in the time domain by Eq. (3) and in the Laplace domain with Eq. (4).

$$ G_C = K_C \left(\varepsilon(t) + \frac{1}{\tau_I} {}_0D_t^{-\lambda} \varepsilon(t) + \tau_D {}_0D_t^{\mu} \varepsilon(t) \right) \tag{3} $$

$$ G_C(s) = K_C \left(1 + \frac{1}{\tau_I s^{\lambda}} + \tau_D s^{\mu} \right) \tag{4} $$

where ${}_0D_t^{-\lambda}$ and ${}_0D_t^{\mu}$ are the λ-order fractional integral and the μ-order fractional derivative, respectively. $\varepsilon(t)$ is the error function that is defined as the difference between the setpoint and the output of the measuring device. *Kc*, τ_I and τ_D are the gain, the integration constant and the derivative constant of the controller, respectively.

2.2.2 Numerical Method [10], [11]

To implement this fractional controller algorithm, a numerical method referred to as the Grünwald-Letnikov approximation and defined in Eqs. (5)-(6), is used for both the fractional integral and the fractional derivative.

$$ {}_aD_t^{\alpha} f(t) \approx \frac{1}{h^{\alpha}} \sum_{m=0}^{\left\lfloor \frac{t-a}{h} \right\rfloor} \omega_m^{(\alpha)} f(t-mh) \tag{5} $$

$$ \omega_m^{(\alpha)} = (-1)^n \binom{\alpha}{m} \tag{6} $$

where h is the time step size, $\omega_m^{(\alpha)}$ are the fractional weights of the Grünwald formula, $\left\lfloor \frac{t-a}{h} \right\rfloor$ is the floor function of the number of terms for which the summation is performed. As given in Eq. (5), the Grünwald-Letnikov approximation is defined for a fractional derivative. When α is a negative number, this approximation of the fractional derivative is equivalent to the corresponding approximation of the fractional integral.

2.2.3 Controller Performance Criteria

For the tuning of controllers and the evaluation of controller performance, three performance criteria are used simultaneously in this investigation: (1) the integral of the time-weighted absolute error (ITAE), (2) the time that the process controlled variable spent outside the zone (OZ) ±5% of the setpoint value, and (3) the integral of the squared differences in the controller output (ISDU). The three performance criteria are defined in Eqs. (7)-(10).

$$ ITAE = \int_0^t t \left| \varepsilon(t) \right| dt \tag{7} $$

$$ OZ = \int_0^t g\left(y(t)\right) dt \tag{8} $$

$$ g\left(y(t)\right) = \begin{cases} 1 & \left|y(t) - y(\infty)\right| \geq 0.05 \left|y(0) - y(\infty)\right| \\ 0 & \left|y(t) - y(\infty)\right| < 0.05 \left|y(0) - y(\infty)\right| \end{cases} \tag{9} $$

$$ ISDU = \int_0^t \Delta u_t^2 dt \tag{10} $$

where t is the time, $\varepsilon(t)$ the instantaneous error, $y(t)$ the process response, $y(0)$ and $y(\infty)$ the initial and steady-state outputs of the process, respectively, and Δu_t^2 represents the square of the change in the controller output at time t. The criterion ITAE punishes less severely errors immediately after a change is made but more severely as time progresses to ensure reaching diligently the setpoint value. The criterion OZ is akin to the settling time, that is often used in assessing the performance of controllers, but varies in a more continuous pattern. The criterion

ISDU punishes the magnitude of the changes in the controller output with the objective of achieving a smooth progression of the controller output.

2.3 Benchmark Systems

To determine which system could benefit the most from the two added degrees of freedom in the $PI^{\lambda}D^{\mu}$ controllers and to compare the performance of $PI^{\lambda}D^{\mu}$ controllers, the control of four representative benchmark systems have been evaluated. These four benchmark systems are a first order plus dead time system, a higher-order system, a nonlinear system and a first order plus integrator system. Figure 1 presents the control block diagram used in simulating the dynamic response of the process subjected to a unit step change of the setpoint. In Figure 1, $y^*(t)$ is the setpoint, $y(t)$ is the process response, $\varepsilon(t)$ is the instantaneous error, $u(t)$ is the output of the controller, $G_C(t)$ is the controller transfer function, $G_P(t)$ is the system transfer function that is comprised of the final control element, the process, and the measuring device. For each system, the performance of the control system is investigated in the absence and in the presence of measurement noise. For the case without noise, the output of the measuring device $y_m(t) = y(t)$, whereas for the case with measurement noise, $y_m(t)$ is defined in Eqs. (11) and (12).

$$y_m(t) = y(t) + a_{noise} \bullet \delta \tag{11}$$

$$\delta \sim N(\mu = 0, \sigma = 1) \tag{12}$$

where a_{noise} is the standard deviation of the Gaussian white noise level, which was set to 0.05 in this investigation for cases with noise. δ is the white noise normal distribution with zero mean μ and unit standard deviation σ.

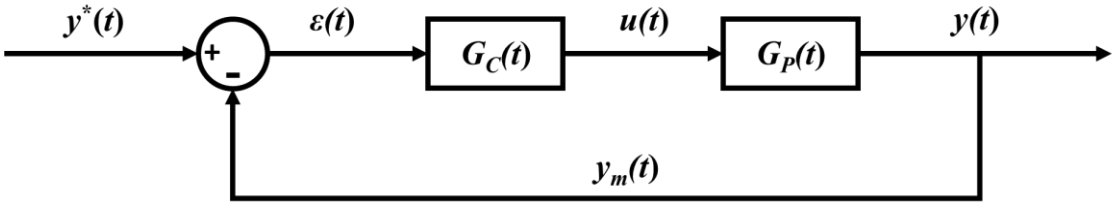

Figure 1 Control block diagram used to simulate the dynamic response of the closed-loop process.

2.3.1 First Order Plus Dead Time System (FOPDTS)

A first order plus dead time system (FOPDTS) is defined by three parameters: the process gain K_P, the process time constant τ_P and the process dead time θ. The dynamic equation for the FOPDTS, defined in the Laplace domain, is given in Eq. (13). The dynamics of many processes in the chemical industry can be adequately represented by a FOPDTS [12].

$$G_P = \frac{K_P e^{-\theta s}}{\tau_P s + 1} \tag{13}$$

2.3.2 Higher-Order System (HOS)

In this investigation, a fourth order system, defined in Eq. (14) as four first-order systems in series, is used to represent higher-order systems. This fourth order system is characterized by five parameters: the process gain K_P and four process time constants τ_{P1}, τ_{P2}, τ_{P3}, and τ_{P4}.

$$G_P(s) = \frac{K_P}{(\tau_{P1}s+1)(\tau_{P2}s+1)(\tau_{P3}s+1)(\tau_{P4}s+1)} \tag{14}$$

2.3.3 Nonlinear System (NLS)

Nonlinear systems are very common in many industries. In this investigation, a nonlinear system, whose form is similar to a FOPDTS, is defined in Eqs. (15)-(17), where both the gain and the time constant of the process are varying with the process output.

$$G_P(s) = \frac{Y(s)}{U(s)} = \frac{K_p e^{-\theta s}}{\tau_p s + 1} \tag{15}$$

$$K_P = K_{P0} + K_{NS} y(t) \tag{16}$$

$$\tau_P = \tau_{P0} + \tau_{NS} y(t) \tag{17}$$

where K_P is the system gain changing with the process output, K_{P0} is a nominal gain of the process at the original steady state, and K_{NS} is a constant coefficient. τ_P is the time constant changing with the process output, τ_{P0} is the nominal time constant of the process at the original steady state, and τ_{NS} is a constant coefficient.

2.3.4 First Order Plus Integrator System (FOPIS)

Similar to a FOPDTS, the first order plus integrator system is defined with three process parameters: the process gain K_P, the process time constant τ_P and the process dead time θ, as defined in Eq. (18). In addition, the FOPIS has an inherent integrator term.

$$G_P(s) = \frac{1}{s}\frac{K_P e^{-\theta s}}{(\tau_P s + 1)} \tag{18}$$

2.4 Controller Tuning

For each of the benchmark systems subjected to a setpoint unit step change, the optimal parameters of the fractional $PI^{\lambda}D^{\mu}$ controller need to be determined. In this investigation, the problem for the optimal tuning of the parameters of the fractional $PI^{\lambda}D^{\mu}$ controller is converted into a multi-objective optimization problem. The five controller parameters, namely Kc, τ_I, τ_D,

λ and μ, were determined by simultaneously minimizing three performance criteria, ITAE, OZ and ISDU, characterizing the closed-loop dynamic response of the system as shown in Figure 2. In this research, the Non-dominated Sorting Genetic Algorithm III (NSGA-III) has been used to circumscribe the Pareto domain for each benchmark system, and then solutions in the Pareto domain were ranked by the Net Flow Algorithm to determine the optimal controller parameters. The Net Flow Algorithm also allowed to determine which controllers in the family of the fractional $PI^{\lambda}D^{\mu}$ controller gave the best performance.

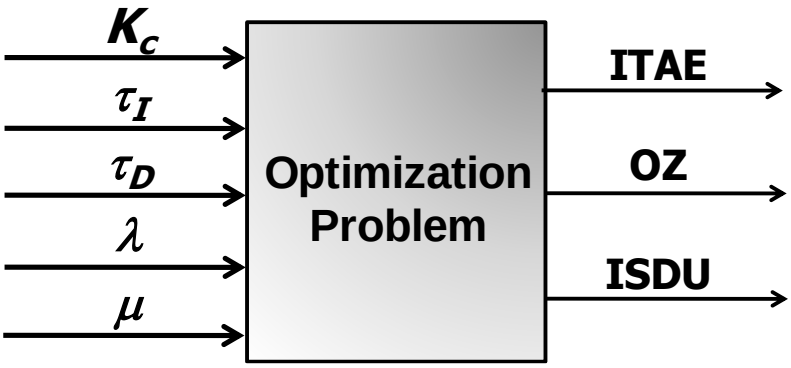

Figure 2 Multi-objective optimization problem for the tuning of the fractional $PI^{\lambda}D^{\mu}$ controllers.

2.4.1 Multi-Objective Optimization Problem

Unlike single-objective optimization problems, multi-objective optimization problems are trade-off problems where the objective vector $\boldsymbol{f}$ composed of multiple potentially conflicting objectives $f_1, \ldots, f_n$, which need to be minimized simultaneously by changing a decision vector $\boldsymbol{x}$ composed of m elements, $x_1, \ldots, x_m$ ($x_i \in \mathbb{R}$), subjected to constraints S as expressed in Eq. (19). Most often, in multi-objective optimization problems, the improvement of one objective leads to the deterioration of some other objectives such that a judicious compromise that best satisfies the overall evaluation and preference of a decision maker must be sought amongst all the objectives. An analytical solution is usually unavailable or very difficult to obtain such that these types of problems are usually solved numerically.

$$\min_{\boldsymbol{x}\in S} \boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} \min\limits_{\boldsymbol{x}\in S} f_1(\boldsymbol{x}) \\ \vdots \\ \min\limits_{\boldsymbol{x}\in S} f_n(\boldsymbol{x}) \end{bmatrix} \tag{19}$$

The decision space is the collection of all decision vectors satisfying constraints S; the objective space is the collection of the corresponding objective vectors.

In this investigation, the concept of domination is used to circumscribe the Pareto domain. A decision vector $\boldsymbol{x_1}$ is said to dominate a decision vector $\boldsymbol{x_2}$ if and only if $f(\boldsymbol{x_1}) \neq f(\boldsymbol{x_2})$ and $f(\boldsymbol{x_1}) < f(\boldsymbol{x_2})$. With respect to the decision vector $\boldsymbol{x_1}$, the decision vector $\boldsymbol{x_2}$ is a dominated solution or

dominated decision vector. Non-domination prevails when two different decision vectors $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ do not dominate each other. The Pareto domain is a subset containing all solutions (both decision vector and objective vector) that are non-dominated with each other. The objective space corresponding to the Pareto domain is called the Pareto front.

2.4.2 NSGA-III [13], [14]

Non-dominated Sorting Genetic Algorithm II (NSGA-II) has been very frequently used to circumscribe the Pareto domain for numerous optimization problems. Non-dominated Sorting Genetic Algorithm III (NSGA-III), which improves the preservation of non-dominated and least dominated elite individuals in the optimization procedure, is a third-generation multi-objective optimization algorithm developed from NSGA-II. The main difference of NSGA-II and NSGA-III is concerned with the offspring selection. The offspring selection of NSGA-III ingeniously adopts normalized structured reference lines to select representative offspring. Based on the shortest distance rule, all candidate solutions for selection are associated with only one reference line. Offspring solutions are chosen from different and least associated reference lines to keep population diversity and representativeness. For details about the exact procedure for using NSGA-III, please refer the paper by Deb [13]. The algorithm of NSGA-III used in this investigation is through open sourced library Distributed Evolutionary Algorithms in Python (DEAP) [14].

2.4.3 Reduction of Fractional $PI^{\lambda}D^{\mu}$ Controllers

In the search of Pareto-optimal solutions by NSGA-III for the tuning of fractional $PI^{\lambda}D^{\mu}$ controllers, different types of controllers in the Pareto domain were identified. Because the special definition of fractional $PI^{\lambda}D^{\mu}$ controllers given in Eqs. (3) and (4), the fractional $PI^{\lambda}D^{\mu}$ controllers can be simplified to other types of simpler controllers belonging to the $PI^{\lambda}D^{\mu}$ controller family depending on the values of the five controller parameters. In this paper, this phenomenon is referred to as controller reduction during the multi-objective optimization. The fractional $PI^{\lambda}D^{\mu}$ controllers that are part of the Pareto domain can be reduced in some cases to any of the eight types of simpler controllers: (1) integer order P, PI, PD and PID controllers, and (2) controllers with fractional order components, namely PI^{λ}, PD^{μ}, $PI^{\lambda}D$ and PID^{μ} controllers. Mathematically, this reduction phenomenon can be explained by examining the decision space of fractional $PI^{\lambda}D^{\mu}$ controllers as shown in Figure 3. These simpler controllers correspond to a line section, a plane, or other specific subset region of the decision space as illustrated in Figure 3. In other words, simpler controllers are special cases of $PI^{\lambda}D^{\mu}$ controllers.

In some investigations, where both the Pareto domains of PID and fractional $PI^{\lambda}D^{\mu}$ controllers were determined, suggested that PID controllers outperformed fractional $PI^{\lambda}D^{\mu}$ controllers for some cases [7]. Mathematically speaking, this is obviously not possible since PID and other

reduced controllers are subset of fractional $PI^{\lambda}D^{\mu}$ controllers in the decision space. The performance of fractional $PI^{\lambda}D^{\mu}$ controllers should be at least as good as the simpler controllers. The most probable reason for PID controllers to perform better than fractional $PI^{\lambda}D^{\mu}$ controllers is that the true Pareto front of fractional $PI^{\lambda}D^{\mu}$ controllers has not been circumscribed. This clearly points to the difficulty of circumscribing a representative Pareto domain for fractional $PI^{\lambda}D^{\mu}$ controllers, especially with their five decision variables. The authors have experienced similar difficulty in determining a representative and complete Pareto domain for fractional $PI^{\lambda}D^{\mu}$ controllers. Only with a large population size, numerous generations and appropriate crossover and mutation operators, it was possible to circumscribe a representative Pareto domain. In this research, all experiments are based on a population of 10000 individuals and 1000 generations, and a well-defined Pareto domain was obtained.

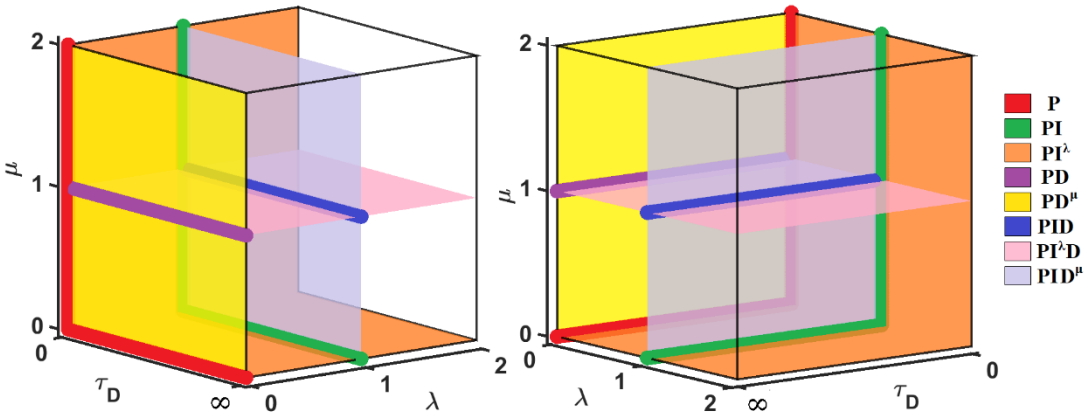

Figure 3 Different types of controllers in the decision space of fractional $PI^{\lambda}D^{\mu}$ controllers.

Quantitatively, the reduction can be compartmentalized into two categories: coefficient reduction and power reduction. The coefficient reduction occurs when a coefficient becomes zero such as $\tau_D = 0$. For illustration purposes, let us consider a solution where the five decision variables are $Kc = 5$, $\tau_I = 2, \tau_D = 0$, $\lambda = 0.5$, and $\mu = 0.5$. Because the coefficient of the derivative component τ_D is 0, the derivative component disappears independently of the value of the fractional derivative order. This illustration corresponds to a coefficient reduction from the fractional $PI^{\lambda}D^{\mu}$ controller to a PI^{λ} controller. Similarly, the reduction to a simpler controller could also occur through a power reduction when λ or μ is either equal to zero or one. For instance, when λ=1 and μ=0, the fractional controller reduces to an effective PI controller with parameters that are modified as shown in Eqs. (20)–(22):

$$G_C = K_C\left(\varepsilon(t) + \frac{1}{\tau_I}\,{}_0D_t^{-1}\varepsilon(t) + \tau_D\varepsilon(t)\right) = K_C(1+\tau_D)\left(\varepsilon(t) + \frac{1}{\tau_I(1+\tau_D)}\,{}_0D_t^{-1}\varepsilon(t)\right) \qquad (20)$$

$$K_C' = K_C\left(1+\tau_D\right) \tag{21}$$

$$\tau_I' = \tau_I\left(1+\tau_D\right) \tag{22}$$

where K_c' is the effective gain of the PI controller whereas τ_I' is the effective integration constant.

Table 1 presents the list of all potentially reduced controllers and the reduction conditions for which fractional $PI^{\lambda}D^{\mu}$ controllers lead to these simplified controllers. To differentiate coefficient reduction and power reduction, a star sign is used to differentiate different type controllers. For example, to differentiate a PI controller from an effective PI controller: (1) if $\lambda = 1$ and $\tau_D = 0$, this primitive controller, resulting from coefficient reduction, is classified as a PI controller, and (2) if $\lambda = 1$, $\tau_D \neq 0$, $\mu = 0$, this effective PI controller, resulting from power reduction, is referred to as a PI* controller. Mathematically, they are the same type of controllers with different controller parameters. Two cases for P and PI^{λ} controllers are defined in a similar fashion.

Table 1 Reduction conditions for which a fractional $PI^{\lambda}D^{\mu}$ controller is reduced to a simplified controller. Note that in this investigation K_c and τ_I cannot be zero or infinite.

Controllers	Reduction Conditions
P	$\lambda=0$, $\tau_D=0$
P*	$\lambda=0$, $\tau_D\neq0$, $\mu=0$
PI	$\lambda=1$, $\tau_D=0$
PI*	$\lambda=1$, $\tau_D\neq0$, $\mu=0$
PI^{λ}	$\lambda\neq0$, $\lambda\neq1$, $\tau_D=0$
PI^{λ}*	$\lambda\neq0$, $\lambda\neq1$, $\tau_D\neq0$, $\mu=0$
PD	$\lambda=0$, $\tau_D\neq0$, $\mu=1$
PD^{μ}	$\lambda=0$, $\tau_D\neq0$, $\mu\neq0$, $\mu\neq1$
PID	$\lambda=1$, $\tau_D\neq0$, $\mu=1$
$PI^{\lambda}D$	$\lambda\neq0$, $\lambda\neq1$, $\tau_D\neq0$, $\mu=1$
PID^{μ}	$\lambda=1$, $\tau_D\neq0$, $\mu\neq0$, $\mu\neq1$

In some comparative research of fractional $PI^{\lambda}D^{\mu}$ controllers with PID controllers, fractional $PI^{\lambda}D^{\mu}$ controllers and PID controllers were independently optimized using multi-objective

optimization [7], [8], [15], [16]. However, considering the fractional $PI^{\lambda}D^{\mu}$ controller reduction in multi-objective optimization, the independent optimization of each type controllers to compare their Pareto domain is not necessary. Mathematically speaking, the decision space for multi-objective optimization includes all members of the fractional $PI^{\lambda}D^{\mu}$ controller family. Therefore, if only some types of controllers exist in the final Pareto domain, it implies that these Pareto-optimal controllers are dominating those controllers that do not appear in the remaining decision space. Therefore, through the optimization of fractional $PI^{\lambda}D^{\mu}$ controllers, the performance of PID controllers can be deduced.

2.5. Analysis of the Pareto Domain

In this research, it was observed that different types of controllers are Pareto-optimal and their solutions coexist in the Pareto domain of fractional $PI^{\lambda}D^{\mu}$ controllers. By considering the different simpler controllers within the fractional $PI^{\lambda}D^{\mu}$ controller family in the Pareto domain, the Net flow Algorithm and the deconvolution of the Pareto domain into different types of controllers are used for the final selection of the optimal operating region and to assess the substitutability of fractional $PI^{\lambda}D^{\mu}$ controllers. First, Net Flow Algorithm is used to give each Pareto-optimal solution a score, which serves to rank the Pareto domain based on the preferences of a decision maker. Second, controllers ranked in the top 5% are deconvoluted into the different types of controllers. Third, the final selection is made based on the average scores and the frequency occurrences of the solutions ranked in the top 5% for each benchmark process.

2.5.1 Net Flow Algorithm [17]

The Net Flow method is an algorithm for ranking all Pareto-optimal solutions based on the preferences of experts. Each Pareto-optimal solution in the Pareto front is given a score. High scores are attributed to solutions with robustness and good trade-off features. To rank the Pareto domain using the Net Flow Algorithm, four parameters for each objective need to be specified by experienced users. For an objective t, these four parameters are denoted as W_t, Q_t, P_t, and V_t, respectively. W_t is the relative weight for objective t, which represents the importance of objective t in comparison to other objectives. The summation of the relative weights for all objectives is equal to unity. Q_t, P_t, and V_t, are three thresholds subjected to the following relationship: $0 \leq Q_t \leq P_t \leq V_t$. The three thresholds are used for comparing all solutions in the Pareto domain for objective t. Q_t is the indifference threshold, defined as the difference of two solutions in the Pareto domain, below which the expert cannot favor one solution over another for objective t. Therefore, if the difference of solutions is below or equivalent to Q_t, their difference is regarded as indiscernible. The preference threshold P_t is the difference of objective

t of two solutions, over which an expert prefers one solution over the other. If objective *t* needs to be maximized, the solution with larger objective *t* is preferred, and vice versa for an objective minimization. The Veto threshold V_t is used to penalize one solution over another if their difference is too high to be tolerated for objective *t*.

The pseudocode to rank the Pareto domain is presented in Algorithm 1. The procedure of the Net Flow Algorithm can be divided into seven steps as follow:

1. The pairwise differences of all solutions in the Pareto domain are calculated. If *s* is the number of solutions in the Pareto domain and *n* the number of objectives, then the difference of solutions *i* and *j* for objective f_k, if it is to be minimized, is calculated using Eq. (23) whereas Eq. (24) is used when objective *k* needs to be maximized.

$$\Delta_k[i,j] = f_{k,i} - f_{k,j} \quad i,\ j,\ k,s \in \mathbb{Z},\ \ i,\ j \in [1,s]\ \ k \in [1,n] \tag{23}$$

$$\Delta_k[i,j] = f_{k,j} - f_{k,i} \quad i,\ j,\ k,s \in \mathbb{Z},\ \ i,\ j \in [1,s]\ \ k \in [1,n] \tag{24}$$

2. Using the values of the difference $\Delta_k[i,j]$ between the objective *k* of two solutions *i* and *j* in the Pareto domain, the individual concordance index $c_k[i,j]$ is calculated using the indifference and preference thresholds to assess the proximity of the two solutions. The individual concordance index is equal to unity within the range of the indifference threshold and progressively decreases to zero between the indifference and preference thresholds as given by Eq. (25).

$$c_k[i,j] = \begin{cases} 1 & if\ \Delta_k[i,j] \le Q_k \\ \dfrac{P_k - \Delta_k[i,j]}{P_k - Q_k} & if\ Q_k \le \Delta_k[i,j] \le P_k \\ 0 & if\ \Delta_k[i,j] > P_k \end{cases} \tag{25}$$

3. The global concordance index, which is the weighted sum of the individual concordance indices is calculated using Eq. (26).

$$C[i,j] = \sum_{k=1}^{n} W_k c_k[i,j] \tag{26}$$

4. The discordance index $D_k[i,j]$ evaluates over the range between the preference and the veto thresholds the dissimilarity of two solutions *i* and *j* for objective *k*, calculated using Eq. (27). When the difference $\Delta_k[i,j]$ becomes equal or exceeds the veto threshold, the discordance index is equal to unity whereas it was zero until reaching the preference threshold.

$$D_k[i,j] = \begin{cases} 0 & if\ \Delta_k[i,j] \le P_k \\ \dfrac{\Delta_k[i,j] - P_k}{V_k - P_k} & if\ P_k \le \Delta_k[i,j] \le V_k \\ 1 & if\ \Delta_k[i,j] > V_k \end{cases} \quad (27)$$

5. Using the global concordance and discordance indices, the relative performance for each pair of Pareto-optimal solutions is finally evaluated by calculating the outranking matrix $\sigma[i,j]$ as defined using Eq. (28).

$$\sigma[i,j] = C[i,j]\left(\prod_{k=1}^{n}\left[1-\left(D_k[i,j]\right)^3\right]\right) \quad (28)$$

6. The final score for ranking solution *i* is calculated using Eq. (29).

$$\sigma_i = \sum_{j\in S^*}\sigma[i,j] - \sum_{j\in S^*}\sigma[j,i] \quad (29)$$

where the first term represents the performance of solution *i* in comparison to all the other solutions in the Pareto domain whereas the second term evaluates the performance of all the other solutions when compared with solution *i*.

7. The last step is to rank all solutions in Pareto domain according to their final score σ, and then to categorize all solutions in the Pareto domain into high-ranked, medium-ranked, and low-ranked solutions for the quantitative and graphical analysis of the entire Pareto domain.

Ranking parameters (W_t, Q_t, P_t, and V_t) used in the Net Flow Algorithm for each objective and for each benchmark system are shown in Table 2. These parameters were the same regardless whether the system was subjected to noise or not. Based on experience gained in this investigation, the controller tuning always led to a relatively small value of the ISDU criterion such that the relative weight of this objective was set at a lower value than for objectives ITAE and OZ. The three thresholds (*Q*, *P*, and *V*) were determined based on the distribution of each objective of the Pareto domain. Although the selection of these parameters bears a certain degree of subjectivity, it was shown that the ranking of the solutions in the Pareto domain is relatively robust for the same Pareto domain distribution.

Algorithm 1: Net Flow Algorithm Pseudocode

Input: number of solutions in Pareto domain *s*, and *W*, *Q*, *P*, *V* for each objective

Output: Ranked Pareto domain with scores

1: **For** each solution *i* in Pareto domain

2: $\sigma[i, j] = 0$, $\sigma[j, i] = 0$

3: **For** each solution *j* in Pareto domain

4: $C[i, j] = 0$, $temp_1 = 0$

5: **For** each objective *k*

6: Calculate $\Delta_k[i, j]$, $c_k[i, j]$, $D_k[i, j]$

7: $C[i, j] = C[i, j] + W_k c_k[i, j]$

8: $temp_1 = temp_1 \bullet \left[1 - \left(D_k[i, j]\right)^3\right]$

9: **End**

10: $\sigma[i, j] = \sigma[i, j] + temp_1 \bullet C[i, j]$

11: $C[j, i] = 0$, $temp_2 = 0$

12: **For** each objective *k*

13: Calculate $\Delta_k[j, i]$, $c_k[j, i]$, $D_k[j, i]$

14: $C[j, i] = C[j, i] + W_k c_k[j, i]$

15: $temp_2 = temp_2 \bullet \left[1 - \left(D_k[j, i]\right)^3\right]$

16: **End**

17: $\sigma[j, i] = \sigma[j, i] + temp_2 \bullet C[j, i]$

18: **End**

19: $\sigma_i = \sigma[i, j] - \sigma[j, i]$

20: **End**

2.5.2 Pareto Domain Deconvolution

Fractional $PI^{\lambda}D^{\mu}$ controllers and other simpler controllers coexist in the Pareto domain. It is therefore interesting to separate these different types of controllers based on the reduction conditions shown in Table 1. The process of separating the Pareto domain into several sub-Pareto domains associated with individual types of controllers is referred to as Pareto domain deconvolution. Since real numerical values are used to search for Pareto-optimal solutions, it is

possible to generate a value of 0.9999 which should reasonably be considered as 1. A tolerance of 0.001 was therefore used for the Pareto domain deconvolution. For instance, a controller with $Kc = 5$, $\tau_I = 2, \tau_D = 0$, $\lambda = 0.9992$, and $\mu = 0.0005$, is considered as PI controller in the deconvolution process. Even though it could still be regarded as a fractional PI^{λ} controller, yet the computation time for a PI^{λ} controller is much larger than the one for a PI controller and the difference in their responses is obviously indiscernible. The Pareto domain deconvolution and investigating the performance of fractional $PI^{\lambda}D^{\mu}$ controllers can assist users in selecting a suitable type of controller with potentially less computation expense. Examples of Pareto domain deconvolution are presented in the following sections.

Table 2 Parameters of the Net Flow Algorithm for benchmark dynamic systems used in this investigation.

System	Objective	*W*	*Q*	*P*	*V*
FOPDS	ITAE/τ_P^2	0.4	0.5	1.0	3.0
	OZ/τ_P	0.4	0.5	1.0	2.0
	ISDU/τ_P	0.2	1.0	2.0	5.0
HOS	ITAE/τ_P^2	0.4	1.5	3.0	9.0
	OZ/τ_P	0.4	0.5	1.0	2.0
	ISDU/τ_P	0.2	1.0	2.0	5.0
NLS	ITAE/τ_P^2	0.4	1	2	4
	OZ/τ_P	0.4	0.5	1	2
	ISDU/τ_P	0.2	0.5	1	2
FOPIS	ITAE/τ_P^2	0.4	1	2	4
	OZ/τ_P	0.4	0.5	1	2
	ISDU/τ_P	0.2	50	100	500

2.6. Results and Discussion

2.6.1 First Order Plus Dead Time System (FOPDTS)

2.6.1.1 Mapping the Decision Space into the Objective Space of Pareto-Optimal Solutions

Results obtained for the FOPDTS (Eq. (13) with $K_P = 1$, $\tau_P = 2$ and $\theta = 0.5$) is first examined and analyzed in greater details than other benchmark systems. The graphs of the normalized Pareto domain of the FOPDTS without noise, obtained by NSGA-III, are presented in Figures 4 and 5. To identify the mapping between the normalized decision space in Figure 4 and the objective space in Figure 5, every solution in the Pareto domain is assigned one color based on the position in $\tau_{D/}\tau_P$, λ and μ coordinates (shown in Figure 4b), based on the Red-Green-Blue

(RGB) color model. Therefore, Pareto-optimal points at different positions in Figure 4b are represented with different colors. Because τ_D, λ and μ also determine the type of controllers as expressed by the reduction conditions given in Table 1, RGB colors of the different points can be associated to different types of controllers akin to Figure 3.

For example, pink points are in the region close to PI controllers whereas green, blue, and orange points are in a region where PI^{λ} controllers prevail rather than in a region where clearly $PI^{\lambda}D^{\mu}$ controllers are prevalent. By considering controller types (shown by different colors in Figures 4 and 5) and their mapping relationship, the fact that points with different colors in both Figures 4 and 5 indicates different types of controllers coexist because fractional $PI^{\lambda}D^{\mu}$ controllers inherently include all members of the $PI^{\lambda}D^{\mu}$ controller family. The observed overlap in Figures 4 and 5 also suggests that these Pareto-optimal solutions may have similar responses.

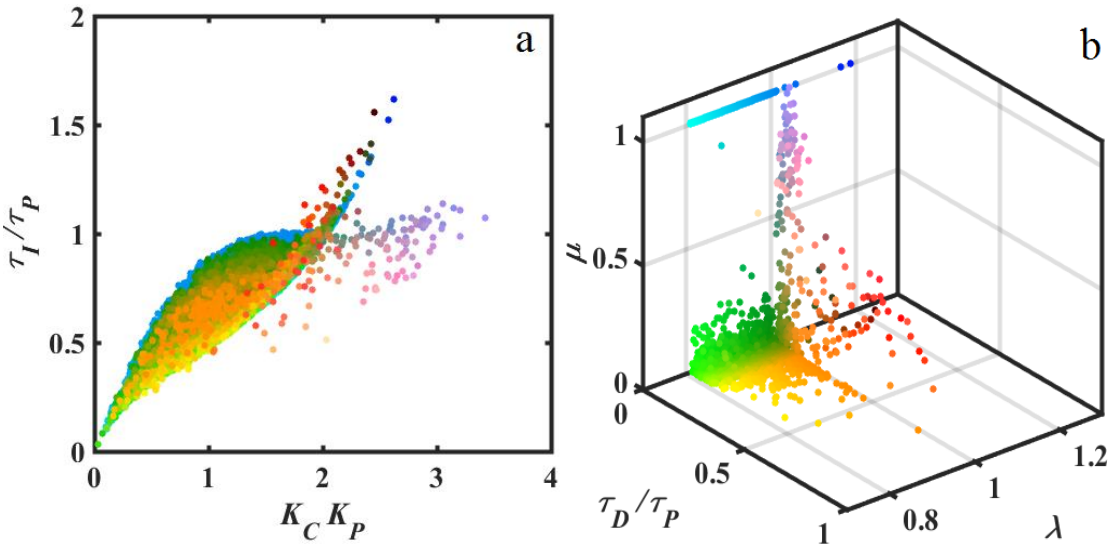

Figure 4 Decision variable space of the normalized Pareto domain for the fractional $PI^{\lambda}D^{\mu}$ controller used to control the FOPDTS without noise. The color scheme is based on the RGB color model using the three normalized decision variables of Figure 4b.

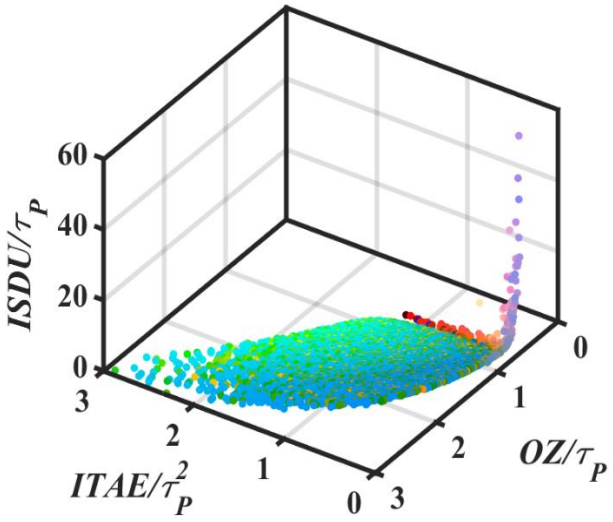

Figure 5 Objective space of the normalized Pareto domain for the fractional PI$^{\lambda}$D$^{\mu}$ controller used to control the FOPDTS without noise. The color scheme is based on the RGB color model using the three normalized decision variables of Figure 4b.

2.6.1.2 Component of the Pareto Domain

Since different types of controllers coexist in the Pareto domain, it was desired to examine if the same Pareto domain could be obtained by combining the Pareto domains circumscribed independently for the series of simpler types of controllers. To perform this validation, a two-step procedure was used. Five controllers, namely PI, PI$^{\lambda}$, PI$^{\lambda}$D, PID and PID$^{\mu}$ controllers were first optimized independently using NSGA-III. Then, the five individual Pareto domains were combined and only the non-dominated solutions of the combined data sets were retained to form a new Pareto domain. The resulting Pareto domain of the combined set is plotted in Figure 6. For simplicity, only the graph of τ_I/τ_P versus $K_C K_P$ is shown here. Results show that the Pareto domain obtained for the fractional PI$^{\lambda}$D$^{\mu}$ controller is identical to the Pareto domain obtained by combining individual Pareto domains of individual controllers. The boundaries of the Pareto domain of Figure 6 are more clearly defined than those of Figure 4a because the Pareto domain of Figure 6 contains a much larger number of solutions as it is the combination of a large fraction of the Pareto domains of five simpler controllers. Pareto domain of the fractional PI$^{\lambda}$D$^{\mu}$ controller can be obtained on its own or by combining the Pareto domains of different simpler controllers, both approaches leading to identical optimal controllers. A complex optimization problem could advantageously be divided into several sub-problems solved independently and then combined to obtain the final optimization results. If all these sub-problems can be solved using parallel computing, the optimization problem could be solved more efficiently and accurately.

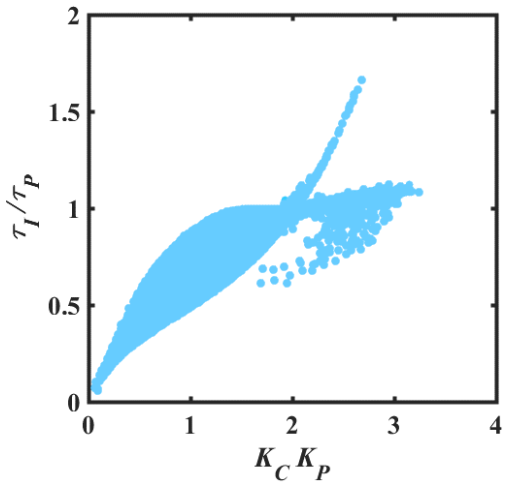

Figure 6 Non-dominated solutions for the collection of independent optimization results from five different types of controllers.

2.6.1.3 Deconvolution of the Pareto Domain

To gain a deeper understanding as to the composition of the Pareto domain, it was separated to show the regions corresponding to different types of controllers. With a tolerance of 0.001, the Pareto domain of fractional $PI^{\lambda}D^{\mu}$ controllers shown in Figures 4 and 5 are deconvoluted into different simpler controllers as presented in Figure 7.

Figure 7 reveals some important information. Firstly, five different types of controllers can be identified in the Pareto domain for the FOPDTS, namely $PI^{\lambda}D$, PI, PID^{μ}, PI^{λ}, and $PI^{\lambda}D^{\mu}$ controllers. Interestingly, the pure linear PID controller is not part of the Pareto domain; the $PI^{\lambda}D$ controller is associated with only one non-dominated solution in the Pareto domain, which suggests that the PID and $PI^{\lambda}D$ controllers are not suitable controllers for FOPDTS without noise.

Secondly, the regions of the Pareto domain associated with the different controllers are partly overlapping both in the decision space and in the objective space. The overlapping distribution in the objective space indicates that their performance criteria (ITAE, OZ and ISDU) are very similar and can generate similar and equally good responses in the time domain. Because of the partial overlap of these controllers with the fractional $PI^{\lambda}D^{\mu}$ controllers, it is paramount to determine quantitatively if the implementation of the more complex fractional $PI^{\lambda}D^{\mu}$ controller, instead of simpler controllers such as PI or PI^{λ} controllers, would show significant improvement in performances.

Thirdly, the non-dominated solutions belonging to the fractional $PI^{\lambda}D^{\mu}$ controller have obviously the widest distribution as it includes all potential controllers present in the Pareto domain. Similarly, the non-dominated solutions associated with the PI^{λ} and PI^{λ}* controllers have the widest distribution among the simpler controllers, almost equivalent to the fractional

$PI^{\lambda}D^{\mu}$ controller. In comparison, the span of the non-dominated solutions belonging to the PI, PI* and PID^{μ} controllers is quite limited. This conspicuous difference demonstrates that the introduction of a fractional integral component in the controller structure has been more beneficial than the introduction of a fractional derivative component. The fractional components, especially the integral part of the controller, have led to a significantly wider Pareto-optimal operating range.

Fourthly, although the fractional derivative component may not bring significantly more benefit, the derivative component shows nevertheless a weak effect in optimizing the ITAE objective. In Figure 7c, the non-dominated solutions associated to $PI^{\lambda}D^{\mu}$, PID^{μ} and $PI^{\lambda}D$ controllers have relatively smaller values of ITEA, with values below 0.3. However, all the other controllers without a derivative component have ITAE values larger than 0.5.

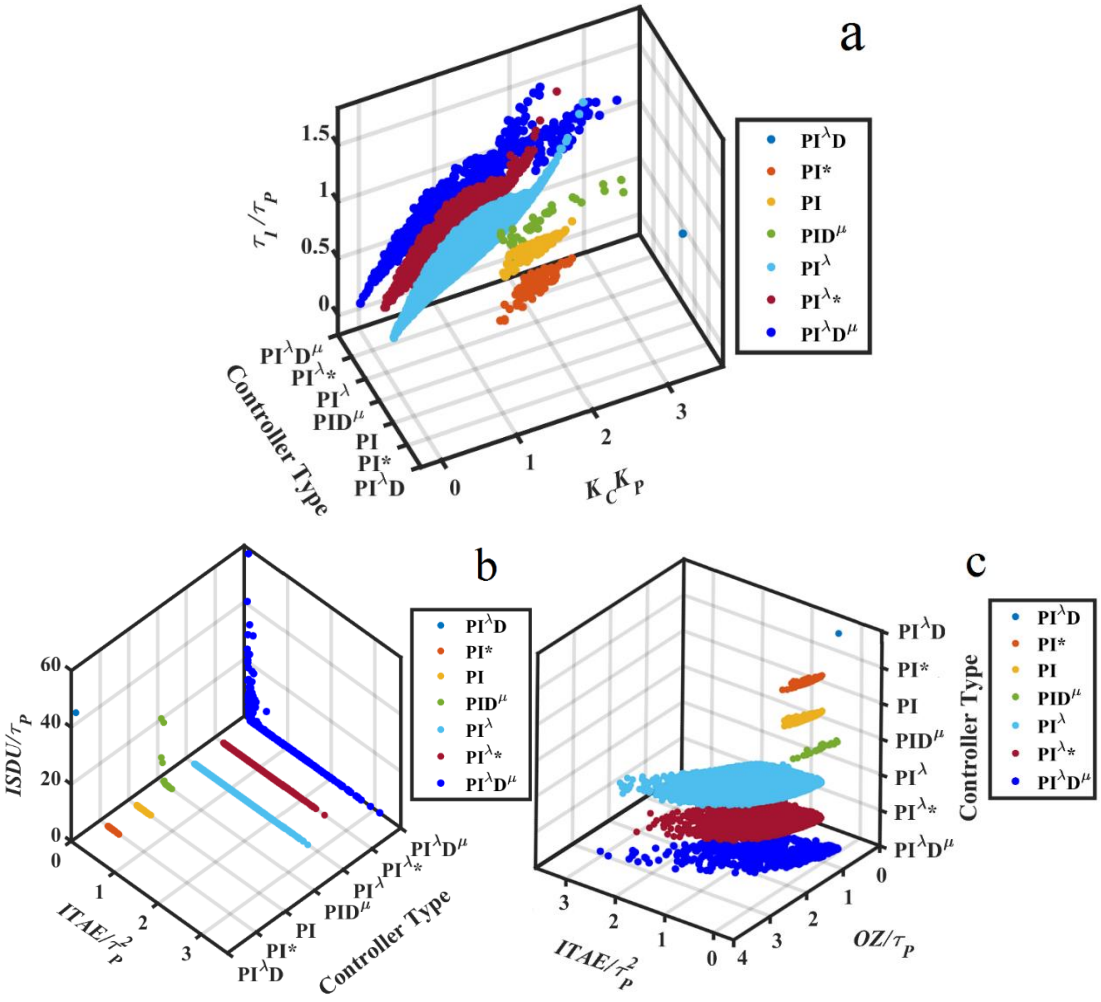

Figure 7 Non-dominated solutions deconvoluted from the collection of independent optimization for circumscribing the Pareto domains of different types of controllers.

2.6.1.4 Analysis of Pareto Domain Using the Net Flow Algorithm

The Net Flow Algorithm, using the parameters given in Table 2 for ranking all Pareto-optimal fractional PI$^{\lambda}$D$^{\mu}$ controllers for the FOPDTS without noise, assigns to every solution in the Pareto domain a score based on their pairwise comparison of the three objectives using the relative weights and the three thresholds specified for each objective. The solutions are then ranked from the highest to the lowest scores and sorted into three categories: solutions ranked in the top 5%, the subsequent 15%, and the 80% remaining solutions. Figure 8 presents the plots of the Pareto domain using these three categories: solutions ranked in top 5% colored in yellow, the subsequent 15% in red, and the remaining 80% in blue.

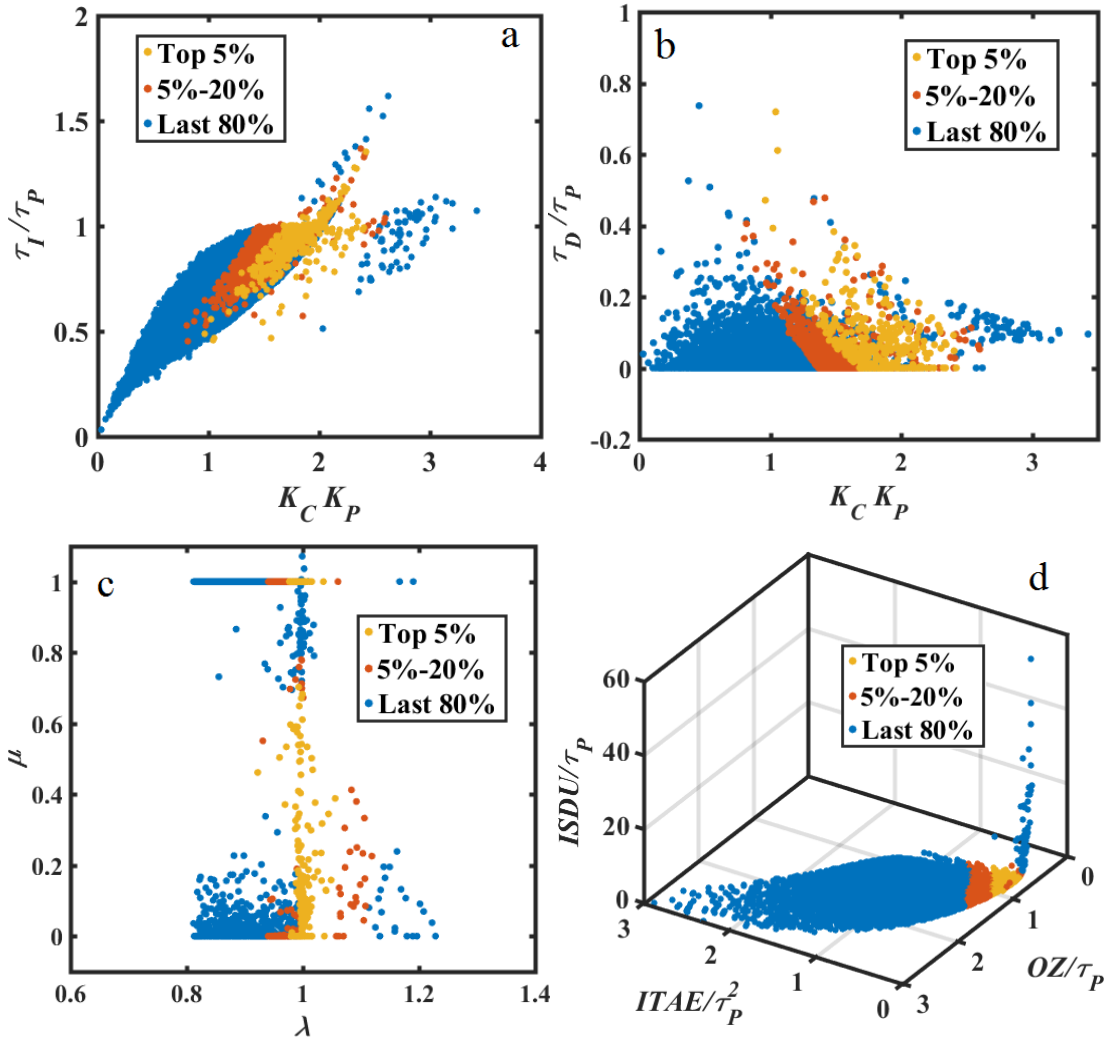

Figure 8 Ranking of the Pareto domain of the fractional PI$^{\lambda}$D$^{\mu}$ controller using the Net Flow Algorithm for the FOPDTS without noise.

The Pareto-optimal solutions ranked in top 5% are those that offer the best compromise considering the three objectives. In the objective space (Figure 8d), the solutions ranked in the top 5% are aggregated in the vicinity of a turning point, where a dramatic increase in ISDU/τ_P

is observed. In the decision variable space (Figure 8a), the solutions ranked in the top 5% are surrounding the bifurcation point of the Pareto domain that is located at a $K_C K_P$ value of 2 and a τ_I/τ_P value of 1. At the bifurcation point in Figure 8a, the Pareto domain separates into two elongating tails. Based on the color scheme of the solutions in Figure 4b, the upward tail is mainly comprised of solutions whose values of λ are increasing from one to higher values. The horizontal tail contains solutions whose values of the derivative order μ are 0.5 to higher values. In other words, solutions ranked in top 5% are at the transition state from PI controllers to PI^{λ} and PID^{μ} controllers. It is important to note that the distribution of top 5% is also the overlapping region in Figure 7.

To determine what type of controllers would be best for the control of the FOPDTS, the frequency of occurrence of these different controllers ranked in the top 5% is examined. From the frequency of occurrence presented in Table 3, five types of controllers coexist in the Pareto domain and 90% of the Pareto-optimal solutions are fractional PI^{λ} controllers. For the top 5% of the Pareto domain, only four types of controllers coexist. Based on the relative average scores (RAS) given by NFM, these four controllers only show a moderate difference. However, the proportions of these four controllers show a significant difference. Considering the proportion and the relative average score, the $PI^{\lambda}D^{\mu}$ controller is recommended for the control of FOPDTS. If a simpler controller algorithm is preferred, then a PI^{λ} controller could be selected.

Table 3 Different types of controllers identified in the whole Pareto domain and the top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the FOPDTS without and with noise.

Controller	Without Noise				With Noise			
	Pareto Domain		Top 5%		Pareto Domain		Top 5%	
	RAS	Proportion	RAS	Proportion	RAS	Proportion	RAS	Proportion
PI, PI*	0.881	3.0%	0.934	23.6%	0.882	0.1%	-	-
PI^{λ}, PI^{λ}*	0.539	90.3%	0.929	47.8%	0.571	77.9%	0.978	53.1%
PID	-	-	-	-	-	-	-	-
$PI^{\lambda}D$	0.484	0.0%	-	-	-	-	-	-
PID^{μ}	0.889	0.4%	0.962	5.0%	0.436	0.1%	-	-
$PI^{\lambda}D^{\mu}$	0.563	6.3%	0.964	23.6%	0.567	21.9%	0.983	46.9%

It is believed the reasons why the fractional $PI^{\lambda}D^{\mu}$ controller has some advantages over the other types of controllers is its versatility imparted by the fractional order components, especially the fractional order integral. As shown above, the region of the top 5%, corresponding to the significantly overlapped area in Figure 7 and the transition region of different controllers in

Figure 8, contains more than one type of controllers. For example, the average value of λ for the Pareto-optimal solutions ranked in the top 5% is 0.998, which implies that the order of integral component of the controller is close to unity and/or uniformly distributed around unity for most controllers in the top 5%. However, given the sensitivity of the integral order λ, it cannot be regarded as unity as the results of Figure 9 suggest. A representative solution randomly picked in the top 5% of the Pareto domain is presented, whose nominal controller parameter values are K_C 2.135, τ_I 1.564, τ_D 0.364, λ 0.971, and μ 0.534. The variation of the three objectives for this controller is plotted as a function of the variation of the integration order λ. Although λ is almost unity, the minimum ITEA objective is exactly the one obtained by NSGA-III. A value of λ=1 is not the nadir for ITAE where ITAE increases by 20% if a PID$^{\mu}$ controller were to be used. A solution for which a value of λ close to unity but not unity possesses the characteristics of a controller between PID$^{\mu}$ and PI$^{\lambda}$D$^{\mu}$.

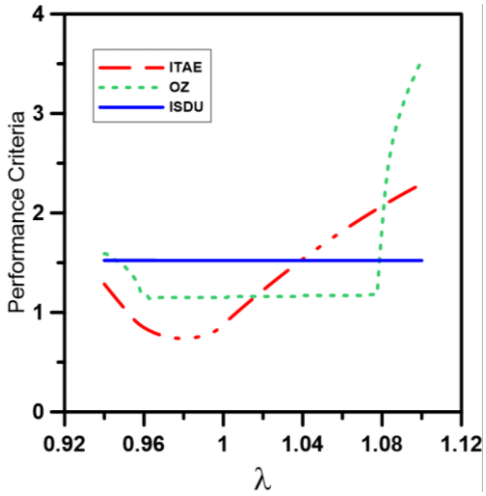

Figure 9 Variation of the three objectives as a function of the order of the integral component for a representative Pareto-optimal solution ranked in the top 5%.

2.6.1.5 First Order Plus Dead Time System with Noise

The identical optimization procedure was used to circumscribe the Pareto domain for the FOPDTS in the presence of measurement noise. Results are presented in Figure 10. Comparing the results of Figures 8 and 10, major differences in the Pareto domain for the FOPDTS in the presence of measurement noise, suggest that a strong integral component and a weak derivative component are preferred for the control of the FOPDTS with noise.

Firstly, the wide distribution of Pareto-optimal solutions observed for values of $K_C K_P$ below 2 in Figure 8a has shrunk to a much thinner region overlapping with the lower boundary of the same region of the noise-free Pareto domain. The shrinkage of the Pareto domain to the lower boundary indicates that smaller values of τ_I, a stronger contribution of the integral component, is necessary to mitigate the effect of white measurement noise. Secondly, for the FOPDTS with

noise, the distribution of the Pareto-optimal solutions ranked in the top 5% in Figure 10c shows a much wider spread with a large proportion of the values of λ greater than unity in comparison to Figure 8c. It appears that a fractional integral component seems to be more important for the efficient control of the FOPDTS with noise. Thirdly, the horizontal tail containing solutions with the derivative order μ above 0.5 in Figure 8a disappeared in Figure 10a; and solutions with μ close to unit in Figure 10c, are also solutions with τ_D at about zero. This result clearly suggests that controllers with a strong derivative component are not preferred for the control of the FOPDTS without noise.

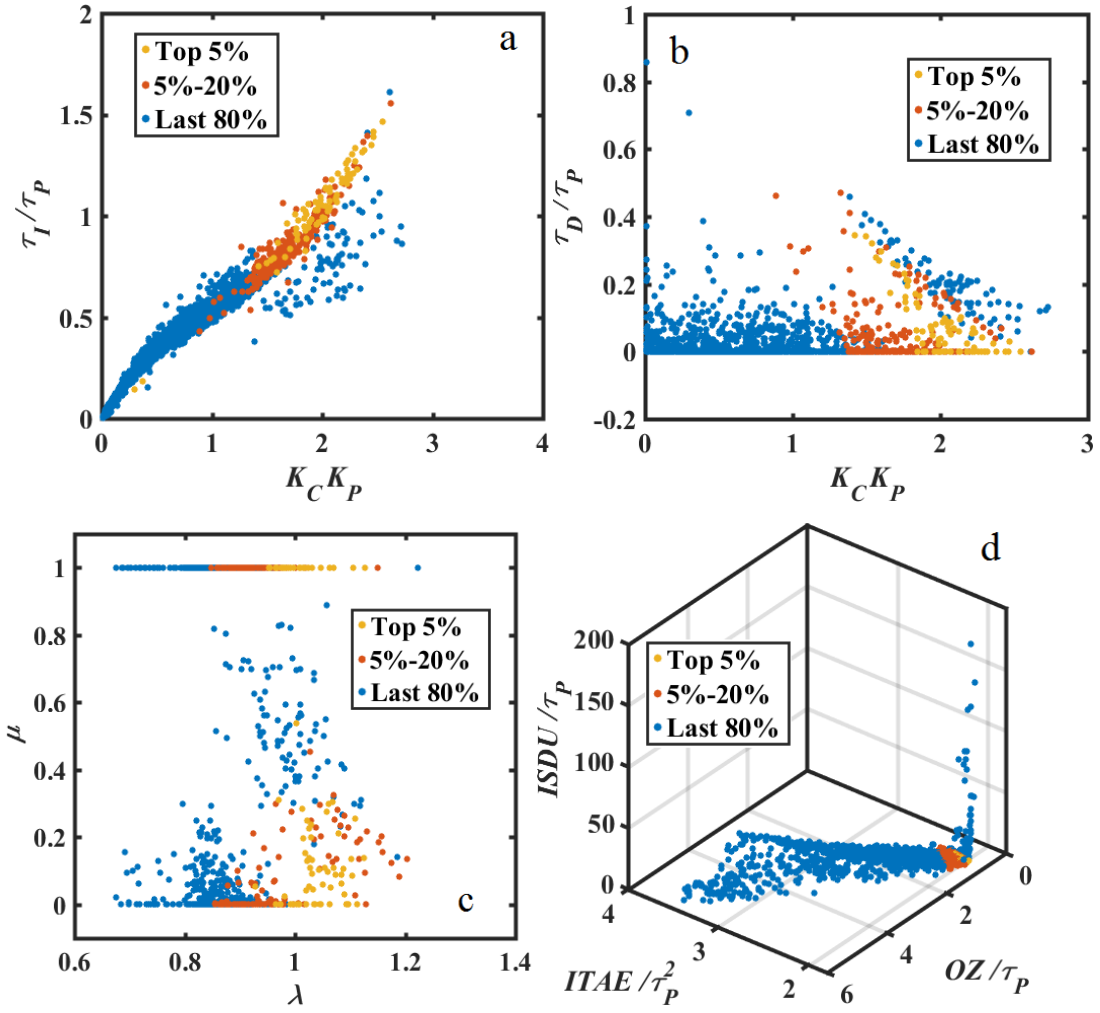

Figure 10 Ranked Pareto domain for the control of the FOPDTS with noise: (a) normalized integral time as a function of the normalized controller gain; (b) normalized derivative time as a function of the normalized controller gain; (c) fractional integral order λ as a function of derivative order μ; (d) 3-D plot of the three objective functions.

As shown in Table 3, four types of controllers are present in the entire Pareto domain but only two types of controllers are predominant with a frequency of occurrence of 99.8%: PI^{λ} and $PI^{\lambda}D^{\mu}$. These two controllers have in general a fractional integral order λ greater than 1, a small

or nil derivative action τ_D and a small fractional derivative order μ. These results imply that a relatively strong fractional integral component is more advantageous than an integer order integral component to help mitigating noise. These two controllers have nearly identical relative scores and close probability of occurrence. Thus, for the FOPDTS with noise, both PI^{λ} and $PI^{\lambda}D^{\mu}$ controllers are recommended.

2.6.2 Higher Order System (HOS)

2.6.2.1 Higher Order System without Noise

An identical procedure was used to circumscribe the Pareto domain for the tuning of the fractional $PI^{\lambda}D^{\mu}$ controller for the fourth order system of Eq. (14) with a gain K_P of unity and with τ_{P1}, τ_{P2}, τ_{P3} and τ_{P4} all equal to 1.0. Because a higher order system can be modeled relatively well using a FOPDTS, their Pareto domains share some commonalities in shape, distribution and overlapping. In this case, the equivalent FOPDTS has a gain K_P of 1.00, τ_P of 2.37 and θ of 1.82. In Figure 11, the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller for the HOS without noise assumes the shape of an elongated Pareto domain of a FOPDTS without noise. Compared with the Pareto domain of Figure 8a, the left branch of the Pareto domain in Figure 11 is narrower whereas the right branch is elongated and wider. For the limitation of space, graphs of the other three decision variables and the three objectives space of the $PI^{\lambda}D^{\mu}$ controller Pareto domain for the HOS without noise are not provided [18].

Table 4 presents the different types of controllers that were present in the complete Pareto domain and the set of Pareto-optimal solutions that were in the top 5%. Six types of controllers are present in the Pareto domain with a predominance of PI^{λ} and $PI^{\lambda}D^{\mu}$ controllers with a proportion of more than 90%. Even though six controllers were identified in the complete Pareto domain, yet only the $PI^{\lambda}D^{\mu}$ controller remains in the Pareto-optimal solutions ranked in the top 5% with a proportion of 99.7%. Because of its overwhelming proportion in top 5%, the $PI^{\lambda}D^{\mu}$ controller is the only controller recommended for the higher order system without noise.

2.6.2.2 Higher Order System with Noise

The Pareto domain for the tuning of $PI^{\lambda}D^{\mu}$ controllers for the HOS with noise was obtained and ranked with Net Flow as in all the other cases. Table 4 shows that five types of controllers were identified in the complete Pareto domain with a clear predominance of PI^{λ} and $PI^{\lambda}D^{\mu}$ controllers with a proportion of more than 99%. It is interesting to note that the linear PID controller was completely dominated by the other types of controllers. For the Pareto-optimal solutions ranked in the top 5%, only the $PI^{\lambda}D^{\mu}$ controller occurs. Therefore, for the control of the HOS with and without noise, the best controller recommended for this application is the $PI^{\lambda}D^{\mu}$ controller.

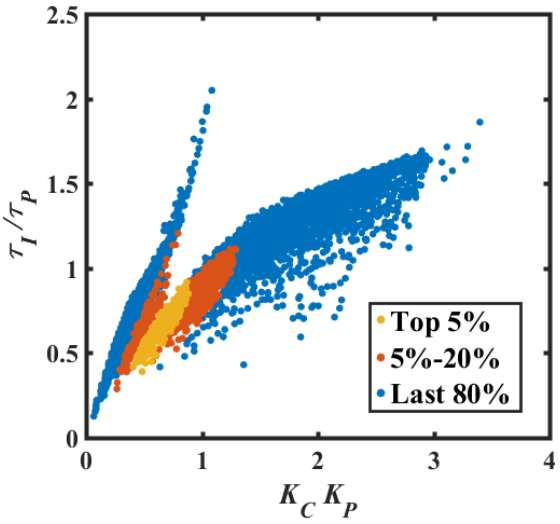

Figure 11 Normalized integral time as a function of the normalized controller gain in the ranked Pareto domain for the higher order system without noise.

Table 4 Different types of controllers identified in the whole Pareto domain and top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the HOS without and with noise.

Controller	Without Noise				With Noise			
	Pareto Domain		Top 5%		Pareto Domain		Top 5%	
	RAS	Proportion	RAS	Proportion	RAS	Proportion	RAS	Proportion
PI, PI*	0.329	2.7%	-	-	0.566	0.2%	-	-
PI^{λ}, PI^{λ}*	0.278	22.6%	-	-	0.470	48.2%	-	-
PID	0.322	0.1%	-	-	-	-	-	-
$PI^{\lambda}D$	0.305	0.2%	-	-	0.501	0.2%	-	-
PID^{μ}	0.310	5.1%	0.900	0.3%	0.425	0.3%	-	-
$PI^{\lambda}D^{\mu}$	0.471	69.3%	0.908	99.7%	0.573	51.1%	0.952	100.0%

2.6.3 Nonlinear System (NLS)

2.6.3.1 Nonlinear System without Noise

The nonlinear system (NLS) is defined in Eqs. (15)-(17) and the constants used in this investigation are 1.0, 1.0, 2.0, and 2.0, respectively for K_{P0}, K_{NS}, τ_{P0} and τ_{NS}. In this section, only the two-dimensional projection of the first two decision variables is plotted in Figure 12 for this case study. The Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller for the NLS without noise has also some common features as the one for the FOPDTS. First, the position of the Pareto domain with respect to $K_C K_P$ axis is nearly the same whereas an upward shift is observed with respect to the τ_I/τ_{P0} axis. Second, the size of the Pareto domain below the bifurcation point is much wider, especially the observed inflation of the lower section. Third, the Pareto-optimal

solutions ranked in the top 5% are also located at the bifurcation point in the upper part of the Pareto domain.

Table 5 presents the different types of controllers that are present in the complete Pareto domain and the set of Pareto-optimal solutions that are ranked in the top 5%. Five types of controllers can be found in the Pareto domain with a net preponderance for the PI^{λ} controller with a proportion of more than 92% followed by the $PI^{\lambda}D^{\mu}$ controller with a proportion of approximately 6%. In the top 5% of the ranked solutions, four types of controllers remain. By considering the relative ranking scores and the relative proportions, the $PI^{\lambda}D^{\mu}$ and PI^{λ} controllers are suggested for controlling the nonlinear system. If computational complexity is a factor, the PI^{λ} controller should be used.

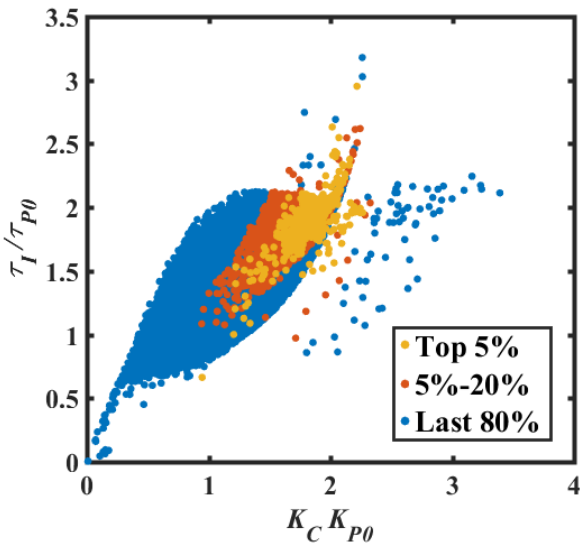

Figure 12 Normalized integral time as a function of the normalized controller gain in ranked Pareto domain for the nonlinear system with noise.

Table 5 Different types of controllers identified in the whole Pareto domain and top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the NLS without and with noise.

Controller	Without Noise				With Noise			
	Pareto Domain		Top 5%		Pareto Domain		Top 5%	
	RAS	Proportion	RAS	Proportion	RAS	Proportion	RAS	Proportion
PI, PI*	0.909	1.2%	0.940	10.6%	0.909	0.1%	-	-
PI^{λ}, PI^{λ}*	0.546	92.4%	0.934	57.0%	0.595	77.7%	0.980	71.0%
PID	-	-	-	-	-	-	-	-
$PI^{\lambda}D$	0.473	0.01%	-	-	-	-	-	-
PID^{μ}	0.729	0.1%	0.958	1.4%	0.731	0.2%	-	-
$PI^{\lambda}D^{\mu}$	0.618	6.3%	0.963	31.0%	0.515	22.1%	0.982	29.0%

2.6.3.2 Nonlinear System with Noise

The Pareto domain for the tuning of the fractional $PI^{\lambda}D^{\mu}$ controller for the nonlinear system with noise was not significantly different than the one without noise [18]. However, only the $PI^{\lambda}D^{\mu}$ and PI^{λ} controllers have a significant predominance in the complete Pareto domain with a combined proportion of 99.7%. For the Pareto-optimal solutions ranked in the top 5%, only these two types of controllers appear. For the NLS with noise, it is recommended to use a PI^{λ} because the relative average score is the same as the $PI^{\lambda}D^{\mu}$ controller and its proportion is much higher. In addition, a $PI^{\lambda}D^{\mu}$ controller is more computationally intensive than a PI^{λ} controller.

2.6.4 First Order Plus Integrator System (FOPIS)

2.6.4.1 First Order Plus Integrator System without Noise

The FOPIS is defined in Eq. (18) and the same values for K_C, τ_P and θ as for the FOPDTS were used (see Section 6.1.1). Since the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller for the FOPIS is significantly different than for all the previous systems, it will be covered in greater details. Figure 13 presents the plots of the five decision variables and the three objectives using the RGB color scheme that was described in Figure 4b. Unlike the control of the other benchmark systems discussed in previous sections, the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller for the FOPIS is split into two distinct regions: one region colored in blue and another region in red and yellow. To assist in the analysis of these results, the ranked Pareto domain, presented in Figure 14, will be used in combination with the color-coded Pareto domain of Figure 13.

One region of the Pareto domain, associated with the color going from red to yellow, is located at the non-zero lower bound value of K_C. As shown in Figure 13c, this region is associated to the lowest values of the objective ISDU/τ_P, which is the reason these Pareto-optimal solutions were retained despite their very poor performance for the other two objectives. For very small values of the controller gain, the dynamic response of the process is dictated by the integration term of the process transfer function and the rise to the set point will be relatively slow and smooth, thereby explaining the very low values of ISDU/τ_P. However, these solutions were ranked very poorly by the Net Flow Algorithm and do not represent a significant interest for the selection of the most suitable controller for the FOPIS.

The second region of the Pareto domain is the most interesting because it contains the Pareto-optimal solutions that are ranked in the top 5%. Intuitively, because of the presence of an integration term in the transfer function of the FOPIS, it would be expected that the integration part of the controller may not be critical. This is the reason for the large values of τ_I/τ_P observed for the higher values of K_C (Figure 14a). Since the upper bound value of τ_I/τ_P was reached, a new optimization was conducted with an upper bound of 50. The obtained Pareto domain was

identical to the one of Figure 14a with the exception that the upward band at higher values of K_C was simply extended to the new upper bound. These newly created Pareto-optimal solutions have lower ISDU/τ_P but were not as good for the other two objectives. None of the solutions of this upward band were ranked within the top 5%.

The solutions in the top 5% occupies a very small area located just prior to the large increase in ISDU/τ_P with relatively low values of the other two objectives. This represents a very judicious compromise amongst all objectives. The five decision variables at the center of the top 5% region are approximately 0.5, 4.0, 2.0, 0.4 and 0.8, respectively of K_PK_C, τ_I/τ_P, τ_D/τ_P, λ and μ.

Table 6 presents the different types of controllers that were present in the complete Pareto domain and the set of Pareto-optimal solutions that were retained in the top 5%. A total of seven types of controllers were identified for the control of the FOPIS with a net predominance of the fractional $PI^{\lambda}D^{\mu}$ controller with a proportion of 95%. The Pareto-optimal solutions ranked in the top 5% come only from the fractional $PI^{\lambda}D^{\mu}$ controller. This is also evident when examining Figure 14c where the values of λ and μ are in the vicinity of 0.4 and 0.8 for the top 5%.

Table 6 Different types of controllers identified in the whole Pareto domain and top 5% of the Pareto Domain and their relative average scores (RAS) and proportions for the FOPIS without and with noise.

Controller	Without Noise				With Noise			
	Pareto Domain		Top 5%		Pareto Domain		Top 5%	
	RAS	Proportion	RAS	Proportion	RAS	Proportion	RAS	Proportion
P, P*	0.003	0.01%	-	-	0.242	0.65%	-	-
PI, PI*	-	-	-	-	-	-	-	-
PI^{λ}, PI^{λ}*	0.002	0.2%	-	-	0.121	15.1%	-	-
PD	0.355	0.8%	-	-	-	-	-	-
PD^{μ}	0.186	4.0%	-	-	0.386	5.4%	-	-
PID	-	-	-	-	-	-	-	-
$PI^{\lambda}D$	0.272	0.4%	-	-	0.435	0.2%	-	-
PID^{μ}	0.335	0.01%	-	-	0.903	0.2%	-	-
$PI^{\lambda}D^{\mu}$	0.532	94.6%	0.972	100.0%	0.604	78.5%	0.966	100.0%

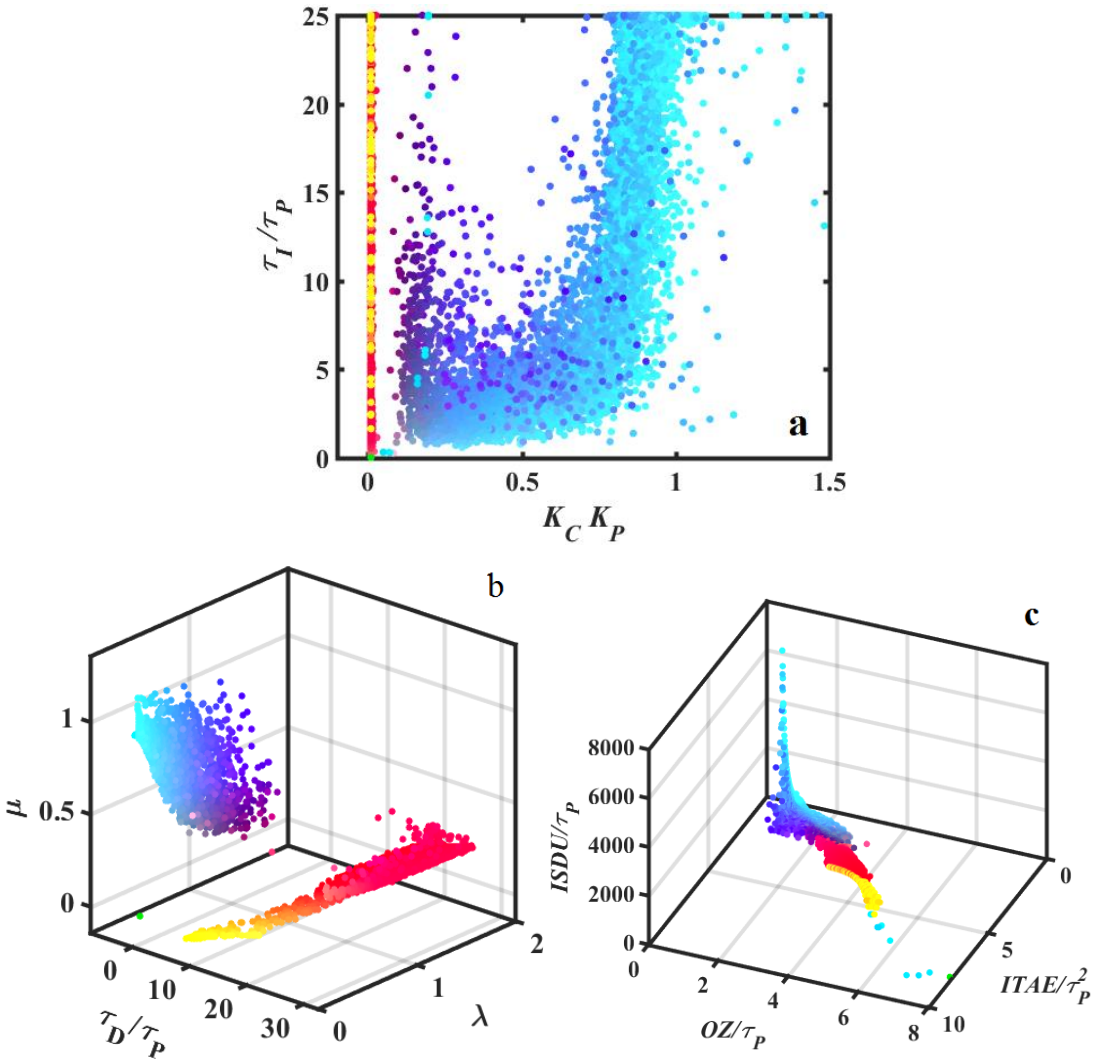

Figure 13 Decision variable space and objective space of the normalized Pareto domain for the fractional $PI^{\lambda}D^{\mu}$ controller used to control the FOPIS without noise. The color scheme is based on the RGB color model using the three normalized decision variables of Figure 4b.

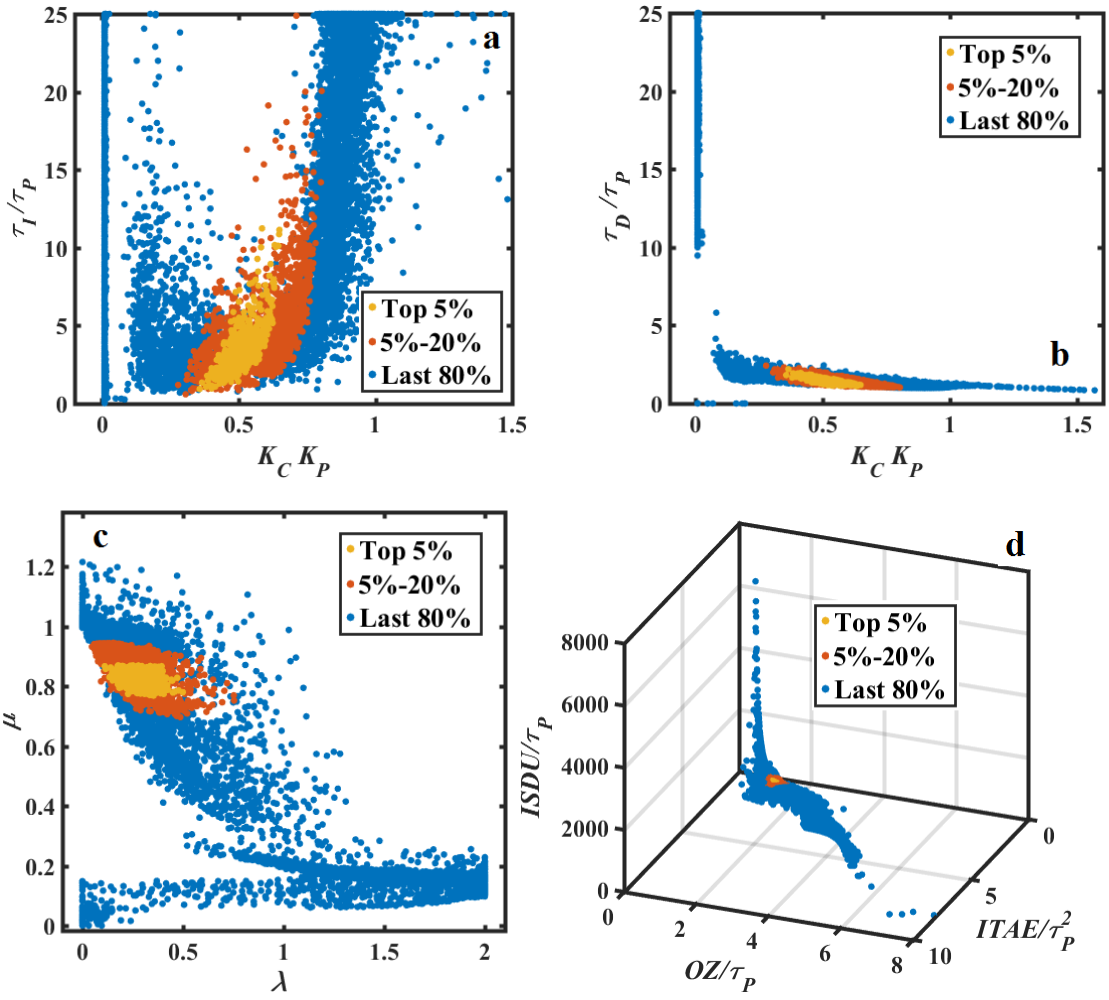

Figure 14 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPDTS without noise.

2.6.4.2 First Order Plus Integrator System with Noise

Table 6 shows that, in the presence of measurement noise, the proportion of PI^{λ} controllers in the complete Pareto domain increases significantly compared to the results for the FOPIS without noise. However, the PI^{λ} controller is not part of the Pareto-optimal solutions ranked in the top 5%. With its high relative score and 100% proportion in the solutions ranked in the top 5%, the fractional $PI^{\lambda}D^{\mu}$ controller is the most suitable controller for the FOPIS with noise.

The derivative action considers the rate of change of the instantaneous error with time to modify the controller output. However, the derivative action loses its benefit in the presence of measurement noise as the derivative action tends to amplify the noise and deteriorate the controller performance. This is undoubtedly the reason that the proportion of the PD controller, ranked in third place in the absence of noise, decreases from 0.8% to 0% for the control of the FOPIS with noise. It is therefore interesting to examine the difference between the fractional D^{μ} component and integer D component since the two types of controllers present in the

solutions ranked in the top 5% have a D^{μ} term. To gain a deeper understanding on the treatment of noise with fractional D^{μ} component and, in particular, the impact of the derivative order μ on the attenuation or amplification of noise, a series of simulations were conducted. Results of these simulations are presented in Figure 15. These results clearly show that noise is increasingly amplified as the derivative order μ increases to unity. The fractional D^{μ} component is therefore less sensitive to measurement noise, especially at lower values of the fractional order μ.

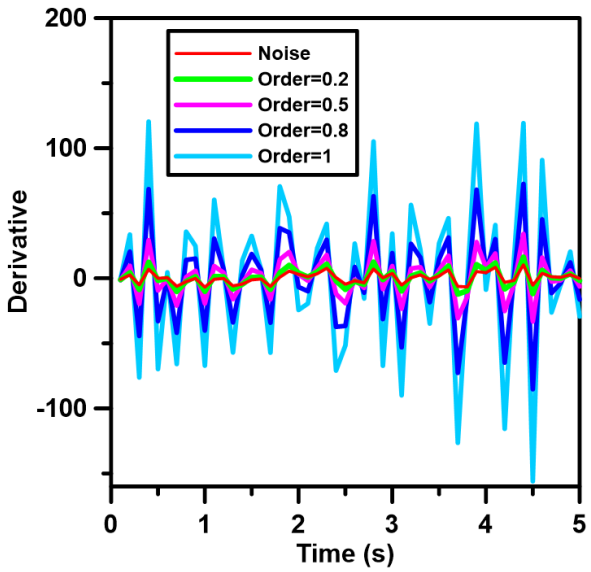

Figure 15 Different fractional order derivative of white noise.

2.7 Response of the Best Pareto-optimal Controllers for Different Systems

To summarize the results obtained in this study, Table 7 presents parameters of the best Pareto-optimal controller for both fractional controllers and linear controllers. These optimal controller parameters correspond to the solutions in the Pareto domain with the highest scores, ranked by the Net Flow Algorithm, for the four benchmark systems. Results of Table 7 clearly show that, for all four benchmark systems without noise, the fractional $PI^{\lambda}D^{\mu}$ controller was ranked as the best controller by Net Flow. On the other hand, the linear controller in the Pareto domain that was ranked the best by the Net Flow Algorithm was the PI controller for all benchmark systems, except the FOPIS. For benchmark systems subjected to noise, fractional $PI^{\lambda}D^{\mu}$ controllers were ranked the best Pareto-optimal controllers for HOS and FOPIS whereas fractional PI^{λ} controllers were favored for FOPDTS and NLS. For the best Pareto-optimal linear controllers, PI controllers were favored for FOPDTS, HOS, and NLS. On the other hand, for FOPIS without noise, the PD controller was ranked the best Pareto-optimal linear controller where a P controller was ranked the best linear controller for FOPIS in the presence of noise.

Table 7 Parameters of the best Pareto-optimal fractional and integer order controllers, ranked by Net Flow, for the four benchmark systems without and with noise.

System	Noise	Controller Type	Decision Variables					Objectives		
			K_C	τ_I	τ_D	λ	μ	ITAE	OZ	ISDU
FOPDTS	No	$PI^{\lambda}D^{\mu}$	2.254	1.947	0.130	0.991	0.590	0.613	1.35	0.520
		PI	2.020	2.034	0.000	1.000	1.000	0.761	1.64	0.041
	Yes	PI^{λ}	2.274	2.609	0.000	1.032	1.000	7.875	1.54	0.523
		PI	1.431	1.673	0.000	1.000	1.000	8.311	2.29	0.195
HOS	No	$PI^{\lambda}D^{\mu}$	0.498	1.058	2.546	0.953	0.442	6.760	4.11	1.244
		PI	0.740	2.780	0.000	1.000	1.000	11.44	5.94	0.006
	Yes	$PI^{\lambda}D^{\mu}$	0.688	2.361	0.738	1.051	0.230	23.31	4.91	0.831
		PI	0.706	2.760	0.000	1.000	1.000	25.00	6.21	0.076
NLS	No	$PI^{\lambda}D^{\mu}$	2.092	3.434	0.249	0.975	0.452	0.689	1.31	0.436
		PI	2.018	4.133	0.000	1.000	1.000	0.800	1.71	0.041
	Yes	PI^{λ}	1.797	4.399	0.247	1.047	0.000	8.077	1.56	0.494
		PI	1.246	3.164	0.400	1.000	0.000	8.166	2.30	0.310
FOPIS	No	$PI^{\lambda}D^{\mu}$	0.401	2.331	3.839	0.179	0.820	2.067	2.22	76.48
		PD	0.828	34.20	2.058	0.000	1.000	0.836	2.07	585.2
	Yes	$PI^{\lambda}D^{\mu}$	0.308	34.38	3.355	1.004	0.603	9.888	3.04	54.72
		P	0.010	49.76	19.66	0.000	0.000	23.48	9.48	0.005

Figure 16 presents the dynamic response of the FOPDTS and FOPIS subjected to a unit set point step change when the best Pareto-optimal fractional and linear controllers were used. The dynamic responses obtained with controllers having fractional components have a similar behavior, characterized with a short response time, mild overshoot and short settling time. The dynamic responses obtained with linear controllers are in general more sluggish, especially in the presence of noise. Similar results were obtained for the HOS and NLS dynamic responses [18]. On the other hand, for FOPIS without noise, Figure 16c visually shows that response obtained with a PD controller is better than the response obtained with a fractional $PI^{\lambda}D^{\mu}$ controller. The ITAE and OZ objectives of Table 7 corroborate this observation. The reason for the fractional $PI^{\lambda}D^{\mu}$ controller being ranked as the best controller is dues to its much lower ISDU value. It is important to stress that the best Pareto-optimal controller is selected as a compromise between the three objectives.

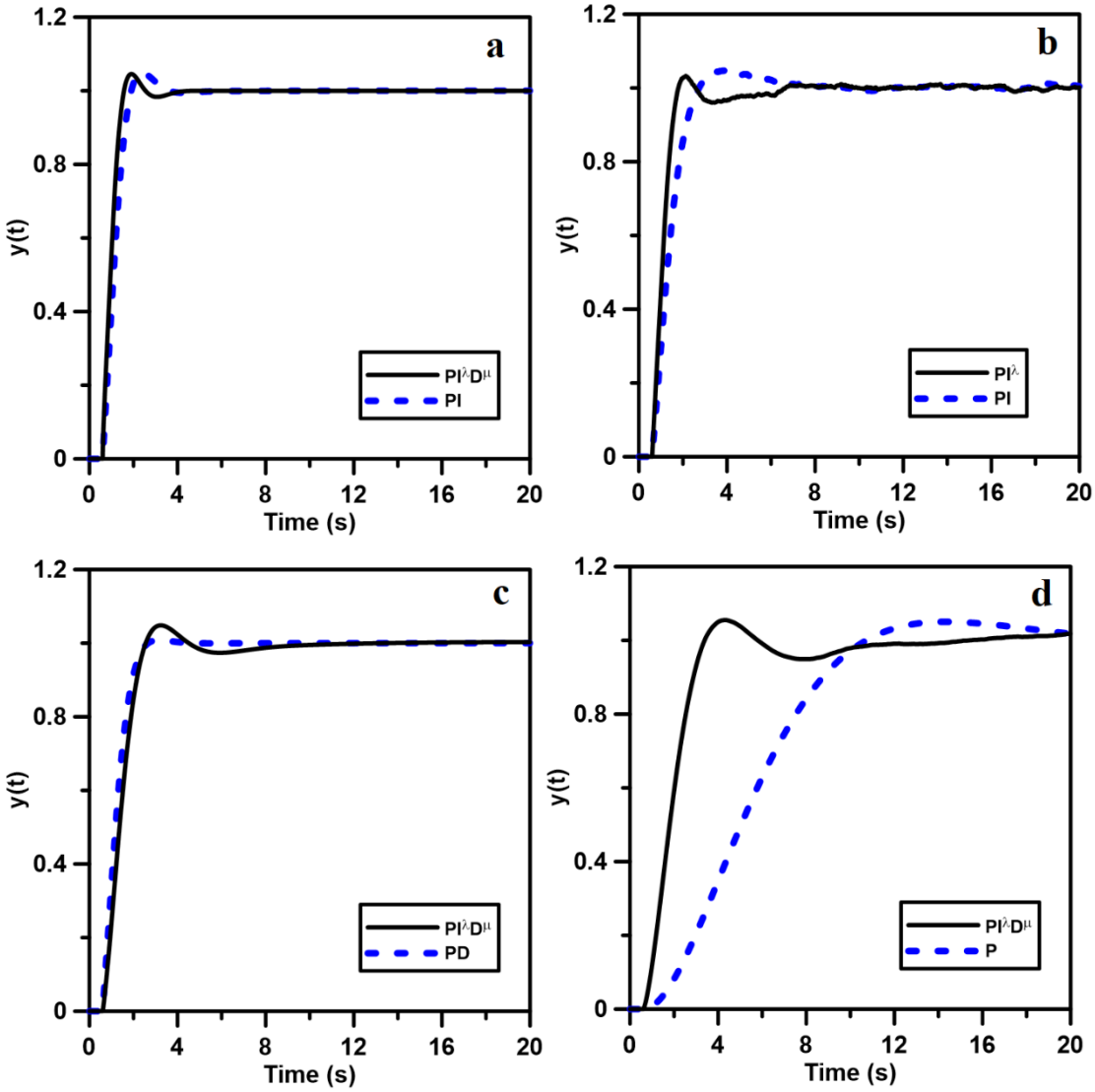

Figure 16 Dynamic response following a unit step change in the set point and when the best Pareto-optimal fractional and integer order controllers were used: (a) FOPDTS without noise; (b) FOPDTS with noise; (c) FOPIS without noise; (d) FOPIS with noise.

2.8 Conclusion

In this investigation, the multi-objective optimization algorithm NSGA-III was used to tune fractional $PI^{\lambda}D^{\mu}$ controllers for the control of four benchmark systems. Fractional $PI^{\lambda}D^{\mu}$ controller was always present among the Pareto-optimal solutions ranked in the top 5% for the four benchmark systems. Results clearly show that the fractional $PI^{\lambda}D^{\mu}$ controller naturally inherit the advantages of different types of controllers that are members of the fractional $PI^{\lambda}D^{\mu}$ controller family. This was especially true for the higher order system and the first order plus integrator system, where the proportion of the fractional $PI^{\lambda}D^{\mu}$ controller is in the vicinity of 100%. For the other two benchmark systems (FOPDTS and NLS), the proportion of the

fractional PI^{λ} controllers was larger than for the fractional $PI^{\lambda}D^{\mu}$ controller. For the latter two systems, it may preferable to use PI^{λ} controllers to reduce the computational effort. It was also shown that adding a fractional integral component had a greater impact on the controller performance than a fractional derivative component. The fractional derivative component is less impacted by the presence of noise and this impact is decreasing with a decrease in the derivative order μ.

2.9 Nomenclature

a_{noise}	Noise level	-
$c_k[i, j]$	Individual concordance index	-
$C[i, j]$	Global concordance index	-
$d(\boldsymbol{s})$	Distance with individual $\boldsymbol{s}$	-
$D_k[i, j]$	Discordance index	-
${}_a D_t^{-\alpha}$	Riemann-Liouville fractional derivative operator with fractional order α with respect time from a to t	[time]$^{-1}$
$\boldsymbol{f}$	Objective function	-
G_C	Transfer function of fractional $PI^{\lambda}D^{\mu}$ controller	-
G_P	Transfer function of process	-
ISDU	Integral of the squared differences in the controller output	[time]
ITAE	Integral of the time-weighted absolute error	[time]
K_C	Gain of fractional $PI^{\lambda}D^{\mu}$ controller	-
K_c'	Effective gain	[time]
K_{NS}	Constant coefficient for nonlinear system	-
K_P	Gain of process	-
K_{P0}	Nominal gain of the process at the original steady state	-
L_0	Non-dominated level	-
L_{last}	Last domination level selected	-
N	Size of population	-
$NC(R)$	Niche-count of reference line R	-
$N(\bar{\mu}, \sigma)$	Standard Normal distribution with mean value $\bar{\mu}$ and standard deviation σ	-
OZ	Time of the process controlled variable spent outside the zone ±5% of the setpoint value	[time]
P_t	Parent population at generation t	-
P_t	Preference threshold for objective t	-

Q_t	Population after crossover and mutation operations at generation t	-
Q_t	Indifference threshold for objective t	-
Q_t	Offspring population	-
$R(\boldsymbol{s})$	Reference line R of individual $\boldsymbol{s}$	-
s	Parameter of Laplace transform	-
S	Decision space	-
S_t	Offspring population	-
S^*	Subset of decision space	-
t	time	[time]
T_t	Total population combined primitive population with population after operations at generation t	-
V_t	Veto threshold for objective t	-
W_t	Relative weight for objective t	-
$\boldsymbol{x}$	Decision variables	-
$y(t)$	Process response	-
$y_m(t)$	Measurement device response	-
δ	Gaussian white noise	-
$\Delta_k[i, j]$	Difference between the objective k of two solutions i and j in the Pareto domain	-
Δu_t^2	Square of the change in the controller output at time t	-
$\varepsilon(t)$	Error function	-
θ	Dead time	[time]
λ	Fractional integral order of fractional $PI^{\lambda}D^{\mu}$ controller	-
μ	Fractional derivative order of fractional $PI^{\lambda}D^{\mu}$ controller	-
σ_i	Score of solution i in Pareto domain	
$\sigma[i, j]$	Outranking matrix	-
τ_D	Fractional derivative constant of fractional $PI^{\lambda}D^{\mu}$ controller	[time]
τ_I	Fractional integration constant of fractional $PI^{\lambda}D^{\mu}$ controller	[time]
τ_I'	Effective integration constant	$[\text{time}]^2$
τ_{NS}	Nominal time constant of the process at the original steady state	[time]
τ_P	Time constant of process	[time]
τ_{P0}	Constant coefficient for nonlinear system	[time]
$\omega_m^{(\alpha)}$	Fractional weights of the Grünwald formula	-

2.10 References

[1] I. Podlubny, L. Dorcak, and I. Kostial, "On fractional derivatives, fractional-order dynamic systems and $PI^{\lambda}D^{\mu}$-controllers," in *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997, vol. 5, pp. 4985–4990.

[2] I. Pan and S. Das, "Fractional-order load-frequency control of interconnected power systems using chaotic multi-objective optimization," *Applied Soft Computing*, vol. 29, pp. 328–344, Apr. 2015.

[3] C. Maheswari, E. Priyanka, and B. Meenakshipriya, "Fractional-order PIλDμ controller tuned by coefficient diagram method and particle swarm optimization algorithms for SO2 emission control process," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 231, no. 8, pp. 587–599, Sep. 2017.

[4] M. Zhang, X. Lin, and W. Yin, "An improved tuning method of fractional order proportional differentiation (FOPD) controller for the path tracking control of tractors," *Biosystems Engineering*, vol. 116, no. 4, pp. 478–486, Dec. 2013.

[5] R. K. Jatoth, V. K. K, N. Bhookya, and G. Ramesh, "A comparative study on design and tuning of integer and fractional order PID controller," in *Proceedings of 2014 International Conference on Modelling, Identification Control*, 2014, pp. 160–165.

[6] Song Bao, Zheng Shiqi, Tang Xiaoqi, and Qiao Wenjun, "Fractional Order Modeling And Nonlinear Fractional Order Pi-Type Control For PMLSM System," *Asian Journal of Control*, vol. 19, no. 2, pp. 521–531, Jul. 2016.

[7] I. Pan and S. Das, "Chaotic multi-objective optimization based design of fractional order PIλDμ controller in AVR system," *International Journal of Electrical Power & Energy Systems*, vol. 43, no. 1, pp. 393–407, Dec. 2012.

[8] G.-Q. Zeng, J. Chen, Y.-X. Dai, L.-M. Li, C.-W. Zheng, and M.-R. Chen, "Design of fractional order PID controller for automatic regulator voltage system based on multi-objective extremal optimization," *Neurocomputing*, vol. 160, pp. 173–184, Jul. 2015.

[9] H. S. Sánchez, F. Padula, A. Visioli, and R. Vilanova, "Tuning rules for robust FOPID controllers based on multi-objective optimization with FOPDT models," *ISA Transactions*, vol. 66, pp. 344–361, Jan. 2017.

[10] S. Das, *Functional Fractional Calculus*. Springer Science & Business Media, 2011.

[11] C. Li and F. Zeng, *Numerical Methods for Fractional Calculus*. CRC Press, 2015.

[12] D. E. Seborg, D. A. Mellichamp, T. F. Edgar, and F. J. Doyle, *Process Dynamics and Control*. John Wiley & Sons, 2010.

[13] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[14] F.-A. Fortin, F.-M. D. Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary Algorithms Made Easy," *Journal of Machine Learning Research*, vol. 13, no. Jul, pp. 2171–2175, 2012.

[15] Z. Chen, X. Yuan, B. Ji, P. Wang, and H. Tian, "Design of a fractional order PID controller for hydraulic turbine regulating system using chaotic non-dominated sorting genetic algorithm II," *Energy Conversion and Management*, vol. 84, no. Supplement C, pp. 390–404, Aug. 2014.

[16] I. Pan and S. Das, "Frequency domain design of fractional order PID controller for AVR system using chaotic multi-objective optimization," *International Journal of Electrical Power & Energy Systems*, vol. 51, no. Supplement C, pp. 106–118, Oct. 2013.

[17] J. Thibault, "Net Flow and Rough Sets: Two Methods for Ranking the Pareto Domain," in *Multi-Objective Optimization*, vol. Volume 5, 0 vols., World Scientific, 2016, pp. 199–246.

[18] X. Shen, "Applications of Fractional Calculus in Chemical Engineering," MASc Thesis, University of Ottawa, 2018.

Chapter 3
Practical Methodology to Initialize Multi-Objective Genetic Algorithm through Evolution History Analysis: A Case Study for the Tuning Fractional Controllers by NSGA-III

Xin Shen and Jules Thibault*

Department of Chemical and Biological Engineering

University of Ottawa, Ottawa, Ontario, Canada K1N 6N5

Abstract – Parameter-dependent multi-objective genetic algorithms have been widely used in optimization problems. These user-defined parameters in multi-objective optimization algorithms, to specify crossover operator, mutation operator, population selection, population size and others, determine the optimization process and the resulting Pareto domain. Similarly, using different operators in multi-objective genetic algorithms may result in different Pareto domains. Therefore, suitable parameters and operators are crucial to efficiently obtain a complete Pareto domain with good diversity. A practical and efficient method, through analysis of the evolution history and Pareto domains obtained to adjust parameters, was proposed to more easily initialize a multi-objective optimization algorithm for specific applications. To introduce this method, the tuning of fractional $PI^{\lambda}D^{\mu}$ controllers by Non-Dominated Sorting Genetic Algorithm III (NSGA-III) was used as a good case study because of its complex and parameter-sensitive Pareto domain.

***Keywords*:** Parameter-dependent, multi-objective genetic algorithm, evolution history analysis, NSGA-III

3.1 Introduction

In the past decades, multi-objective optimization, which considers problems involving more than one objective function to be optimized simultaneously, has gained significant attention. These multi-objective optimization problems are encountered very often in the real world. Because of their complex multi-objective features, analytical solutions are rarely available, and numerical multi-objective genetic algorithms are commonly used in the search of a compromised solution considering all objective functions. Currently, multi-objective optimization genetic algorithms (MOGA) can be classified into three categories: Pareto-ranking based, aggregation-based, and indicator-based [1]. Pareto-ranking based MOGAs are algorithms generating offspring with Pareto optimality, such as NSGA-II [2], SPEA2 [3], and NSGA-III [4].

In these Pareto-ranking based MOGAs, a number of user-defined parameters characterizing offspring need to be specified based on the experience of the user in order to keep variety in the population and rapid convergence to Pareto-optimal solutions. For example, the number and positions of reference points in NSGA-III algorithm are important parameters that require to be specified carefully. Besides these user-defined parameters, suitable crossover and mutation operators are additional parameters to be selected by users. In the past decades, a variety of significant real-coded crossover operators have been proposed by researchers: parental central crossover (PCX) [5], blend crossover (BLX-α) [6], differential evolution (DE) [7], and adaptive hybrid crossover operator (AHX) [8] amongst others. However, unlike the wide diversity of crossover operators, commonly used mutation operators are comprised of some basic types: Gaussian mutation, Cauchy mutation, and exponential mutation [9]. Nevertheless, other complex mutation operators were also proposed, like the Polynomial Mutation (PLM) operator and its elitist highly disruptive version [10], [11] and the Shrink-Mutation operator [12].

Given the large number of operators to be selected and their parameters, to obtain a good Pareto domain, users not only need to select a crossover operator and a mutation operator for the multi-objective optimization algorithm, but also specify their parameters to reap the full potential of the algorithm. In many cases, it is not trivial to select these user-defined parameters characterizing multi-objective optimization algorithms. One way to select suitable multi-objective optimization operators and their associated parameters is to use another optimization algorithm to optimize the current optimization algorithm until robust and well-defined Pareto domain can be obtained. However, this method is computationally expensive, and it is highly probable that additional user-defined parameters would be introduced. Ideally, users of multi-objective optimization algorithms should be able to easily and quickly initialize a multi-objective algorithm and obtain representative and repeatable results. To achieve this objective, a practical and amenable method to adequately initialize the application of multi-objective

optimization, called evolution history analysis, is proposed in this investigation. Through the analysis of the evolution history of the Pareto domain, users can relate the progression in circumscribing the Pareto domain with specified parameters to guide in the selection of suitable operators and their most appropriate parameters.

The motivation to propose the evolution history analysis stems from the difficulty encountered in the selection of operators and their associated parameters to obtain a properly defined and consistent Pareto domain. In our experimentation using NSGA-III for the fractional $PI^{\lambda}D^{\mu}$ controller optimal tuning, it was observed that the resulting Pareto domains were unstable, incomplete and very sensitive to the parameters and types of operators. These problems are not specific to this particular algorithm and users truly need to modulate the optimization algorithm to maximize the probability to find the true optimal solutions of the system under consideration. To more clearly explain how the evolution history analysis can be used for these problems, the optimal tuning of fractional $PI^{\lambda}D^{\mu}$ controllers using NSGA-III is used as the case study in this investigation.

The objective of the paper is to show that the evolution history analysis can be used as a practical method to determine parameters and select operators to obtain a good and reliable Pareto domain. The paper is organized as follows: Section 3.2 presents the fundamental concepts of multi-objective optimization, the fractional controller tuning problem and challenges that were encountered in circumscribing a complete Pareto domain. Section 3.3 explores the important factors that were investigated in this case study. Section 3.4 analyzes the optimization results based on the evolution history analysis. Finally, Section 3.5 provides general guidelines how to set algorithmic parameters efficiently for an optimization problem.

3.2 Fractional Controller Multi-Objective Optimization Problem

3.2.1 Fundamental Concept of Multi-Objective Optimization

A typical multi-objective optimization problem consists in the minimization of objective vector $\boldsymbol{f}$ composed of n usually conflicting objectives $f_1, \ldots, f_n$, whose independent variables are elements of a decision vector $\boldsymbol{x}$ composed of m elements, $x_1, \ldots, x_m$ ($x_i \in \mathbb{R}$), subjected to constraints S as expressed in Eq. (1):

$$\min_{\boldsymbol{x}\in S} \boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} \min\limits_{\boldsymbol{x}\in S} f_1(\boldsymbol{x}) \\ \vdots \\ \min\limits_{\boldsymbol{x}\in S} f_n(\boldsymbol{x}) \end{bmatrix} \tag{1}$$

The decision space is comprised of the collection of all decision vectors satisfying constraints S; the objective space is the set of the corresponding objective vectors. A decision vector $\boldsymbol{x_1}$

dominates a decision vector $\boldsymbol{x}_2$ if and only if $f(\boldsymbol{x}_1) \neq f(\boldsymbol{x}_2)$ and $f(\boldsymbol{x}_1) < f(\boldsymbol{x}_2)$. With respect to decision vector $\boldsymbol{x}_1$, the decision vector $\boldsymbol{x}_2$ is a dominated point or dominated decision vector. Non-domination prevails when two different decision vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ do not dominate each other. The Pareto domain is a subset of the decision space, containing all solutions that are non-dominated with each other.

3.2.2 Fractional PI$^{\lambda}$D$^{\mu}$ Fractional Controller and Its Design [13]

To introduce the concept of the evolution history analysis for setting up a multi-objective optimization algorithm, the tuning of fractional PI$^{\lambda}$D$^{\mu}$ controllers was used as a good illustrative example because of the difficulties encountered for obtaining a complete Pareto domain. More details are presented in the following section. The fractional PI$^{\lambda}$D$^{\mu}$ controller defined by Eq. (2), is a novel controller that is currently being seriously considered in process control for the good performances that were achieved in some applications.

$$G_C = K_C\left(\varepsilon(t) + \frac{1}{\tau_I}\,{}_0D_t^{-\lambda}\varepsilon(t) + \tau_D\,{}_0D_t^{\mu}\varepsilon(t)\right) \tag{2}$$

where ${}_0D_t^{-\lambda}$ and ${}_0D_t^{\mu}$ are the λ-order fractional integral and μ-order fractional derivative, respectively. K_c, τ_I and τ_D are the gain, the integration constant and the derivative constant of the controller, respectively. Process output error $\varepsilon(t)$ is the difference between the desired setpoint and the measured process variable.

The general definition of the fractional integral and fractional derivative are given in Eqs. (3) and (4).

$${}_aD_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)}\int_a^t (t-s)^{\alpha-1} f(s)\,ds, \quad t > a, \quad \alpha > 0 \tag{3}$$

$${}_aD_t^{\alpha} f(t) = \frac{d^n}{dt^n}\,{}_aD_t^{-(n-\alpha)} f(t) = \frac{1}{\Gamma(n-\alpha)}\frac{d^n}{dt^n}\left[\int_a^t (t-s)^{n-\alpha-1} f(s)\,ds\right], \quad t > a \tag{4}$$

where n is an integer satisfying $n-1 \le \alpha < n$.

In this case study, the fractional controller was designed to control a known linear first order plus dead time (FOPDT) process. At time $t = 0$, the setpoint of the feedback system was subjected to a unit step change. The FOPDT model, which can adequately represent many high order processes, is characterized completely using three parameters: the process gain K_P, the process time constant τ_P and the process dead time θ. The equation for the FOPDT model, defined in the Laplace domain, is given in Eq. (5).

$$G_P = \frac{K_P e^{-\theta s}}{\tau_P s + 1} \tag{5}$$

The controller design problem was formulated as a multi-objective optimization problem to demonstrate how the evolution history analysis was used to initialize the optimization algorithm. The optimization algorithm NSGA-III selects the five decision variables of the fractional controller to minimize simultaneously three performance criteria as shown in Figure 1. In this case study, the decision variables are the five parameters of the fractional controller: K_C, τ_I, τ_D, λ and μ. The three performance criteria defined in Eqs. (6) - (9), all to be minimized, are: (1) the integral of the time weighted absolute error (ITAE), (2) the time that the controlled variable spent outside the zone ±5% of the set point value (OZ), and (3) the integral of the squared differences in the controller output (ISDU).

$$ITAE = \int_0^t t\left|\varepsilon(t)\right| dt \tag{6}$$

$$OZ = \int_0^t g\left(f(t)\right) dt \tag{7}$$

$$g\left(f(t)\right) = \begin{cases} 1 & \left|f(t) - f(\infty)\right| \geq 0.05\left|f(0) - f(\infty)\right| \\ 0 & \left|f(t) - f(\infty)\right| < 0.05\left|f(0) - f(\infty)\right| \end{cases} \tag{8}$$

$$ISDU = \int_0^t \Delta u_t^2 dt \tag{9}$$

where t is the time, $\varepsilon(t)$ the instantaneous error, $f(t)$ the process response, $f(0)$ and $f(\infty)$ the initial and final steady-state outputs of the process, respectively. Δu_t^2 represents the square of the change in the controller output at time t. Criterion OZ is equivalent to the settling time, very often used in assessing controller performance, but is a more continuous function.

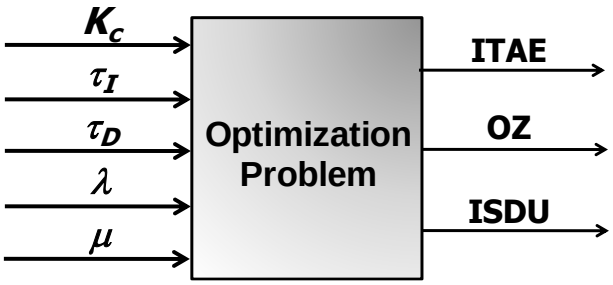

Figure 1 Optimization problem for the tuning of a fractional $PI^{\lambda}D^{\mu}$ controller.

3.2.3 Challenges to Circumscribe the True Pareto Domain

For the tuning of fractional controllers using NSGA-III, the types of operators and user-defined parameters in the optimization algorithm were initially set according to experience. However, the obtained Pareto domains were very sensitive to these user-defined parameters and types of

operators. It was observed that Pareto domains with different shapes were obtained even for the same parameters and the same types of operators. An investigation has therefore been undertaken to develop a practicable procedure to initialize optimization algorithm for systematically finding a repeatable well-defined Pareto domain.

To analyze and determine the reasons for the difficulty to achieve a complete and reliable Pareto domain, a large number of experiments were performed with different multi-objective algorithms and with a wide range of settings of these algorithms. Figure 2 presents the plot of the first two decision variables of the Pareto domain from the combined results of all these experiments. Because of the huge data bank and the wide exploration of the decision variable domain, it is assumed that the Pareto domain associated with Figure 2 is an excellent approximation of the true Pareto domain. As it is difficult to represent multi-dimensional information using graphs, only the first two controller parameters, τ_I as a function of K_C, were used as indicators in this case study for the determination of the Pareto domain. As shown in Figure 2, this Pareto domain can be divided into several regions. Here, different colors and indices, are used to quantify the probability for finding these different regions. In the following sections, these index values are used to refer to these regions of the Pareto domain. The ranking of probability for finding these regions from highest to lowest are Regions 2 & 5, Region 6, Regions 1 & 7, Region 4, and Regions 3 & 8.

When compared to commonly used benchmark multi-objective optimization problems, including ZDT [14], DTLZ [15], and CEC [16], the Pareto domain for the fractional controller tuning problem possesses some unique features hindering circumscribing a complete Pareto domain. First, the shape of the Pareto domain is very irregular and uneven. Second, the density of solutions of the Pareto domain is highly heterogeneous. Thirdly, the Pareto domain is discontinuous between different regions. Because of these special features, this case study can be very useful to gain a deeper understanding on the relationship existing between the optimization performance with user-defined parameters, types of operators, evolution history of Pareto domains and the development of a procedure to more easily initialize optimization algorithms.

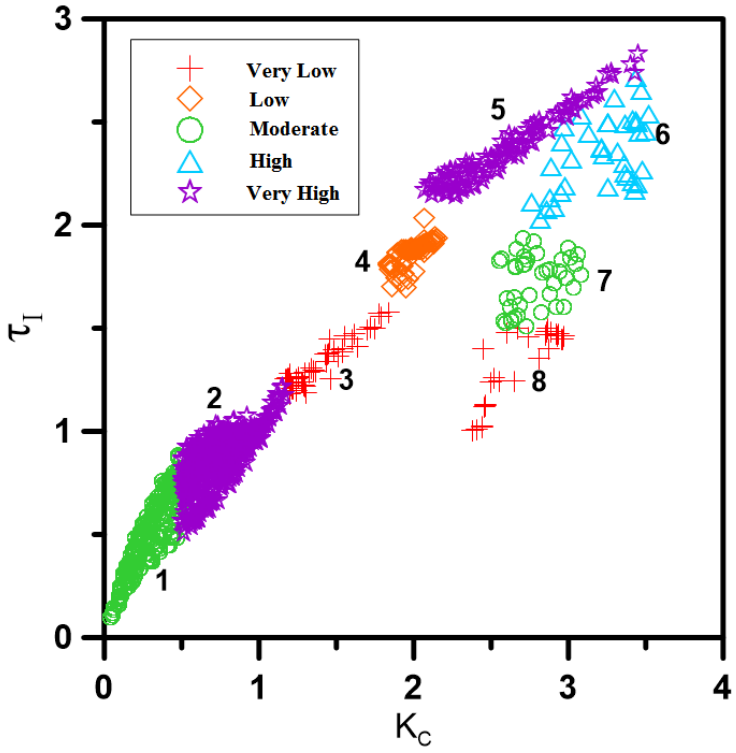

Figure 2 Plot of the first two decision variables of the Pareto. The color of numbered regions corresponds to their probability of occurrence.

3.3 Important User-defined Factors to Initialize NSGA-III

In this case study, the tuning of fractional controllers was used as an optimization problem and NSGA-III was used as the optimization algorithm. A large number of tests were performed by varying the numerous user-defined factors and types of operators, known to influence the optimization performance of NSGA-III and the resulting Pareto domain. This section mainly introduces the definitions of these user-defined factors, including parameters characterizing crossover operator, mutation operator, population selection, and types of operators used.

3.3.1 Crossover Operator [6]

At each generation, when two parents are selected to bring about two new offspring, the crossover operator mixes the genes of the two parents, namely genes $\boldsymbol{x}_{P1}$ and $\boldsymbol{x}_{P2}$, to generate offspring individuals $\boldsymbol{x}_{S1}$ and $\boldsymbol{x}_{S2}$ inheriting genes partially from their parents for each decision variable. In this investigation, a modified linear crossover operator, equivalent to the Blend crossover operator (BLX-α) [6], is tested. This operator is defined in Eqs. (10) - (12)

$$\gamma = (1+2\alpha)\chi - \alpha \boldsymbol{I} \tag{10}$$

$$\boldsymbol{x}_{S1} = (\boldsymbol{I} - \gamma)^T \boldsymbol{x}_{P1} + \gamma^T \boldsymbol{x}_{P2} \tag{11}$$

$$\boldsymbol{x}_{S2} = \boldsymbol{\gamma}^T \boldsymbol{x}_{P1} + \left(\boldsymbol{I} - \boldsymbol{\gamma}\right)^T \boldsymbol{x}_{P2} \tag{12}$$

where $\alpha \in \mathbb{R}$ is a key parameter of the blend crossover operator that is specified by the user; $\boldsymbol{\chi} \in \mathbb{R}^m$ is a vector whose elements are independently subjected to the same uniform distribution $\chi_n \sim U[0,1)$; $\boldsymbol{I} \in \mathbb{R}^m$ is a vector $[1, \ldots, 1]^T$ of dimension m.

As shown by Eqs. (10) - (12), when α is 0, values of γ lie in the range from 0 to 1; when α is greater than 0, values of γ range from values less than 0 to values greater than 1 as shown in Figure 3. In other words, user-defined parameter α determines the fraction of gene γ inherited from parents. Geometrically speaking, when α is 0, two offspring are generated through interpolation between two parents; when α is greater than 0, two offspring are generated through both interpolation and extrapolation. Offspring generated by extrapolation can be interpreted as offspring strengthening some features and removing other features from their parents; offspring generated only by interpolation, do not bring extra diversity in the population than the one inherited from the parents. Considering both diversity and inheritance of optimality, the best value to set α for the crossover operator is unknown.

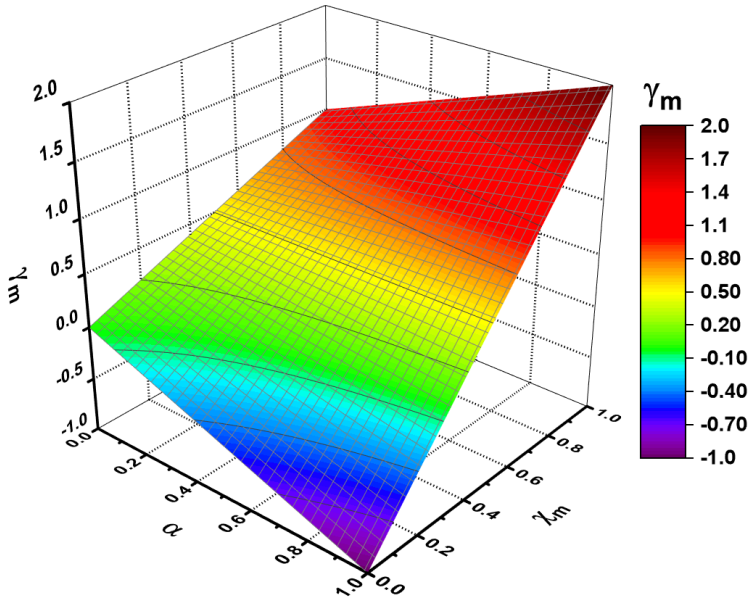

Figure 3 Offspring Distribution of Blend-α Crossover Operator.

3.3.2 Mutation Operator

To bring extra variety to the population, the mutation operator is used following the crossover operation where generated genes can change locally. One important parameter of the mutation operator is the mutation rate, characterizing the probability to bring diversity into the population. The mutation rate ξ_m is defined here as the probability of a decision variable x_m in the decision vector $\boldsymbol{x}$ to mutate independently from the current gene x_m to a new gene x'_m. Another important factor to consider is the type of mutation operator that needs to be selected for an optimization

problem. For simplicity, two basic mutation operators, namely the uniform distribution operator and the normal distribution operator, have been examined in this investigation.

The uniform distribution operator is limited by a series of constraints. Consider an arbitrary decision variable x_m that lies within the range D_m of the constraints of the decision variable. This new gene x'_m is a random value between the upper bound $sup\ D_m$ and the lower bound $inf\ D_m$, as expressed in Eq. (13).

$$x_m' \sim U\left[inf\ \ D_m, sup\, D_m\right) \tag{13}$$

Similarly, the normal distribution operator will add a normally distributed perturbation δx_m to x_m and ensure that the new gene x'_m still remains within the imposed decision variable constraints as shown in Eq. (14). The perturbation δx_m obeys a normal distribution whose expectation μ is equal to 0 and the standard deviation is σ_m. The zero expectation implies that the exploration revolves around x_m and, mathematically, this operation increases or decreases x_m only in its vicinity.

$$x_m' = \begin{cases} inf\ \ D_m & x_m + \delta x_m \le inf\ \ D_m \\ x_m + \delta x_m & inf\ \ D_m < x_m + \delta x_m < sup\, D_m \\ sup\, D_m & x_m + \delta x_m \ge sup\, D_m \end{cases} \qquad \delta x_m \sim N\left(\mu_m = 0, \sigma_m^2\right) \tag{14}$$

3.3.3 Population Size and Decision Space Size

The population size and the decision space size are two important user-defined factors that were investigated in this case study. The population size N is fixed and corresponds to the number of parents and children that are kept at each generation. Intuitively, a larger population should lead to a deeper exploration of all potential solutions and can delineate more clearly the boundary of the Pareto domain. However, the optimization with a higher population size could be very time-consuming as the time of computation increases exponentially with the size of the population.

In addition, the size of the decision variable space is a factor intimately coupled with the population size. For instance, if the decision variable constraints for a decision variable x_m were $0 \le x_m \le 20$, the optimization problem would be relatively simpler than for the same problem with decision variable constraints defined in the range $0 \le x_m \le 50$. Indeed, the distribution density for the same population size will be much greater for the smaller range, at least for the initial generations. To increase the distribution density, it is possible to use a larger population size but at the expense of larger computation time. More details about the population size and the size of the decision space are provided in subsequent sections.

3.3.4 Division of the Objective Space

Non-dominated sorting genetic algorithm III (NSGA-III) is a third-generation multi-objective optimization algorithm used to circumscribe the Pareto domain, whose prominent innovation is that the offspring selection is based on normalized structured reference lines to keep population diversity and representativeness. For details about the exact procedure for using NSGA-III, please refer the paper by Deb [4]. This improved offspring selection with reference points and reference lines relies on the number of divisions of the objective space. For instance, in Figure 4, red dots represent different reference points on a normalized 3-dimensional hyperplane (3 objectives) circumscribed by a blue dash line. For every objective dimension, the range is divided into 3 sections, namely 4 reference dots. Every red point determines a reference line. All individuals, here typified by a single green dot, are associated to the closest reference line for offspring selection. In this approach, the diversity preservation in offspring selection depends on the distribution and number of reference lines. In turn, these structured reference points and reference lines rely on the number of divisions of the objective space. Thus, the number of divisions of the objective space is a key factor to be investigated.

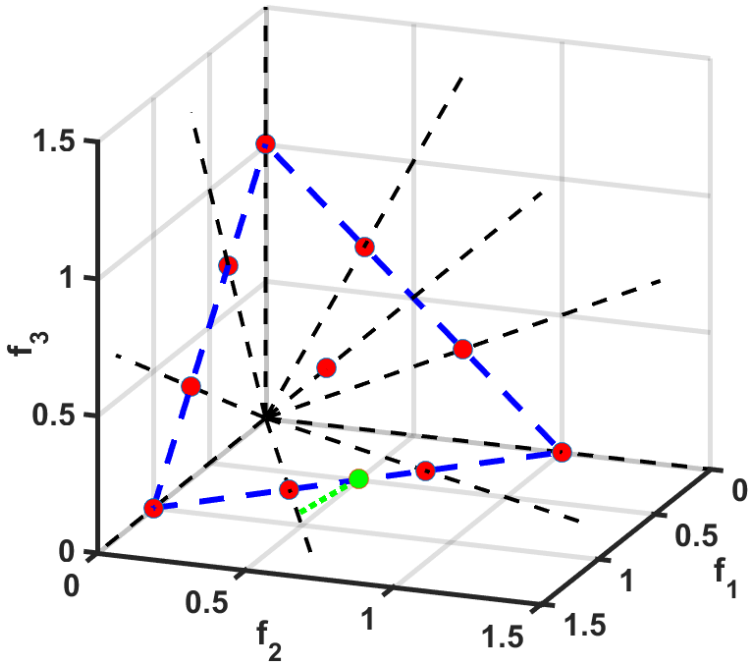

Figure 4 Schematic diagram to illustrate the creation of reference points and reference lines using the association operator.

3.4 Evolution History Analysis

In this investigation, a new procedure, referred to as the evolution history analysis, is proposed as a practical approach to properly initialize multi-objective optimization, namely determining user-defined parameters and selecting suitable operators. The evolution history of the population as a function of the generation contains a large amount of information about the evolution path and the population distribution. This information can reveal and help understanding how key factors influence the evolution process of the population progressively defining the Pareto domain, especially highlighting some significant transformations during the evolution of the population from one generation to the next. These important user-defined factors characterizing the evolution procedure eventually determine the final Pareto domain obtained. Figure 5 presents a schematic overview of the intimate relationship that exists between the key factors characterizing the optimization algorithm, the evolution history of the population and the obtained Pareto domain. In other words, users can specify these key factors in order to design and stabilize the evolution path of the population through many generations to obtain a stable and repeatable Pareto domain. In the current case study, the completeness of the Pareto domain for the optimal tuning of fractional controllers using NSGA-III and the multi-objective optimization performance have been improved prominently based on the evolution history analysis.

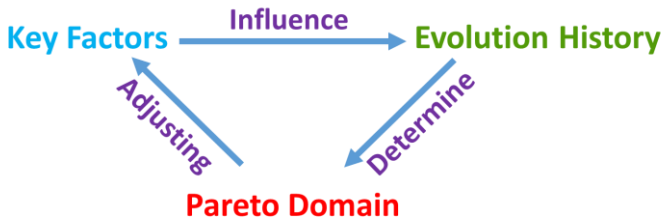

Figure 5 Evolution History Analysis

3.5 Computational Experiments: Results and Discussion

A series of computational experiments have been conducted to analyze the impact of different factors on the evolution history and how the evolution history of the population from one generation to the next determines the resulting Pareto domain. The factors associated to each experiment and the corresponding figures are given in Table 1. Through the analysis and comparison of nuances of the results presented in these figures, a deeper insight of the application of the evolution history analysis to initialize multi-objective optimization algorithm can be achieved.

Table 1 Different optimization parameters used to obtain Pareto-optimal solutions presented in each figure of Section 6.

Figures	Population Size	Crossover	Mutation	Mutation Rate	Decision Variable Ranges				
	P_t	α	-	ξ_m	D_1	D_2	D_3	D_4	D_5
6a, 8a	10000	0	Uniform	20%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
6b, 8b	10000	0.5	Uniform	20%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
7	10000	0, 0.5	Uniform	20%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
9	10000	0.8	Uniform	20%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
10, 11a	10000	0.5	Uniform	60%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
11b	10000	0.5	Uniform	100%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
12a	10000	0.5	Normal	60%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
12b	10000	0.5	Normal	100%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
13a	10000	0.5	Normal	100%	[0.01,50]	[0.01,50]	[0,50]	[0,1]	[0,1]
13b	1000	0.5	Normal	100%	[0.01,5]	[0.01,5]	[0,5]	[0,1]	[0,1]

3.5.1 Crossover

To evaluate the influence of the parameter α of the crossover operator on the evolution history, computational experiments based on the blend crossover operator and uniform distribution mutation operator have been conducted. Results of these experiments are presented in Figures 6 to 9. Figure 6 shows the first two decision variables, K_C and τ_I, of the Pareto domain for experiments performed with different values of the crossover parameter α (see Eqs. (10)-(12)) and a constant mutation rate ξ_m = 20%.

The Pareto domain in Figure 6a was obtained with a value of α = 0, which implies that only interpolation of parents' genes occurred in the entire crossover process. As can be observed in Figure 6a, a crossover parameter α of zero results in a Pareto domain, which is predominantly located in Region 2 of Figure 2. The Pareto domain with values of K_C below 0.4 and above 1.25 are missing. On the other hand, the Pareto domain in Fig. 6b was obtained with a value of the crossover parameter α = 0.5 for which solutions with values of K_C greater than 1.25 (in Regions 5 and 6 of Figure 2) were obtained. The resulting Pareto domain shown in Figure 6b corresponds to an experiment performed with a crossover operator that includes both interpolation and extrapolation. This example clearly illustrates the strong impact of the crossover parameter α on the resulting Pareto domain and the importance of the crossover operator to include extrapolation to achieve better optimization results. Despite the significant improvement, some parts of Region 1 and a large area between values of K_c of 1.25 and 2.2 are missing.

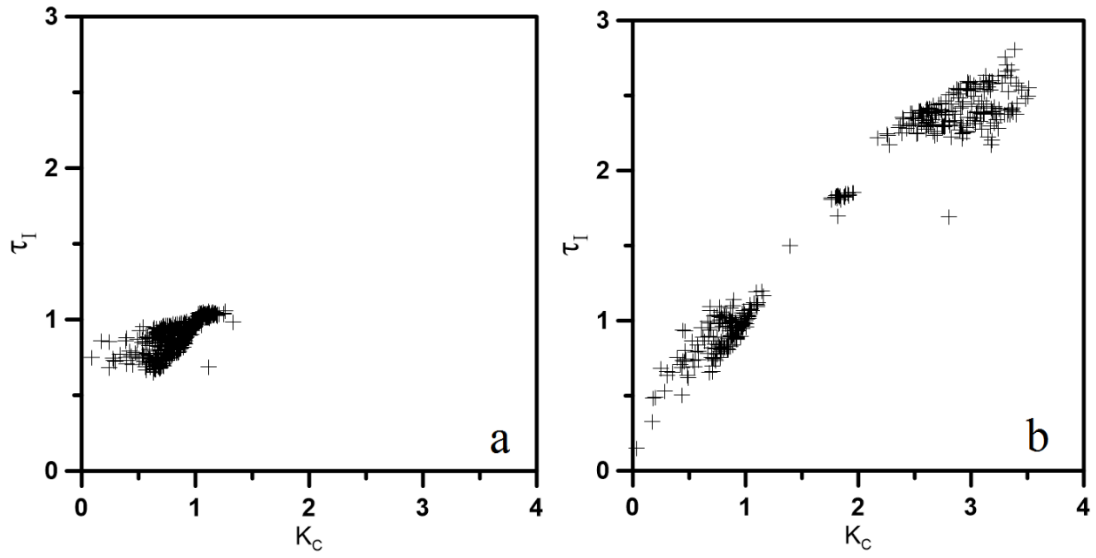

Figure 6 Plot of the two decision variables K_C-τ_I of the Pareto Domain obtained when the crossover parameter α = 0 (a) and α = 0.5 (b) with a constant mutation parameter (ξ_m = 20%).

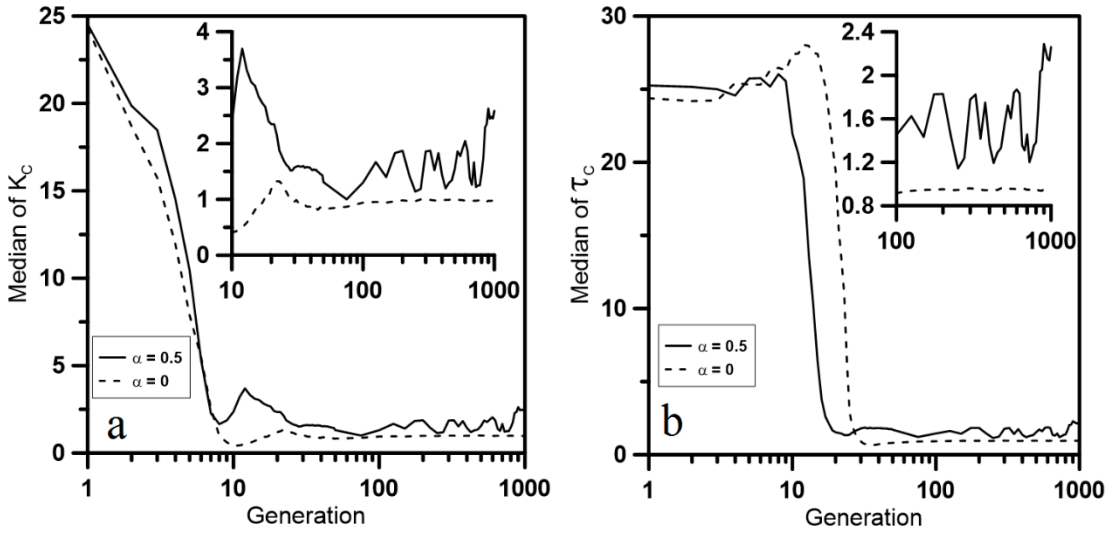

Figure 7 Plots of the history of the median values of K_C (a) and τ_I (b) through the population evolution process.

To explore the underlying reasons for the very strong influence of the crossover parameter α, the evolution history analysis of the decision variables K_C and τ_I was performed. The median of the first two decision variables K_C and τ_I of the entire population through the evolution in optimization process from generation to generation is plotted in Figures 7a and 7b, respectively. As shown in Figure 7, the initial median values of K_C and τ_I are approximately 25 at the beginning because the population was initialized using uniform random values with upper limits for K_C = 50 and τ_I = 50 and the lower limit was close to zero. The median value of K_C drops rapidly within the first 8 generations. A similar dramatic drop is also observed for the

value of the median of τ_I between generations 15 and 30. This sharp decrease indicates the occurrence of a significant transmutation of the population from one status to a new status where the density, the distribution and the performance all changed following the identification of an increasing number of Pareto-optimal solutions. Following the observed transmutation (from generation 7 in Figure 7a), the curves of the median value of K_C for the crossover parameter $\alpha = 0$ and $\alpha = 0.5$ show totally different evolution paths. Indeed, the median value of K_C for $\alpha = 0.5$ is always greater than the median of K_C for $\alpha = 0$ after the transmutation.

Generation 9, given its special location at the end of the transformation phase and beginning of increasing K_C for $\alpha = 0.5$, was used as a good sample. Figure 8 shows the dense regions of the population distribution at generation 9 for $\alpha = 0$ and $\alpha = 0.5$. Figure 8a clearly shows that the population distribution for $\alpha = 0$ has a narrow and very dense distribution with the great majority of K_C values converging well within the region of less than 2. On the other hand, the population distribution for $\alpha = 0.5$ has a much wider distribution with K_C over 2. These population distributions obtained in the transformation do lead to two distinct Pareto domains. These results clearly show that the generation of offspring individuals resorting only to interpolation ($\alpha = 0$) leads to a rapid population distribution shrinkage into a denser zone of solutions with narrower ranges (K_C below 2), which prevents identifying well-circumscribed Pareto domains. Crossover operator with both interpolation and extrapolation can alleviate population shrinkage into a denser zone. Thus, the crossover parameter α should be larger than zero; a crossover parameter $\alpha = 0.5$ appears to be a good balance with both extrapolation and interpolation having a 50% probability occurrence in the optimization process. Nevertheless, an additional experiment with a crossover parameter $\alpha = 0.8$ was performed to observe if further improvement was possible. Results of this experiment are presented in Figure 9. These results suggest that a higher value of the crossover parameter is not sufficient on its own to ensure the identification of solutions located in low density zones and to ensure the entire Pareto domain is completely circumscribed. To improve optimization results, the evolution history is used to analyze the synergistic effect of other factors.

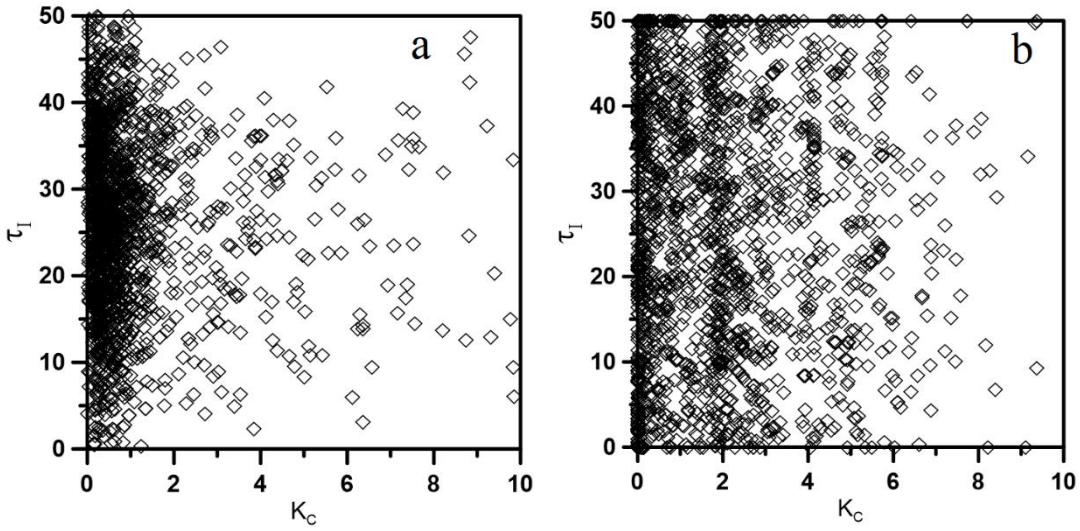

Figure 8 Population distribution for $\alpha = 0$ (a) and $\alpha = 0.5$ (b) at the 9th generation of the optimization process for a population size of 10000.

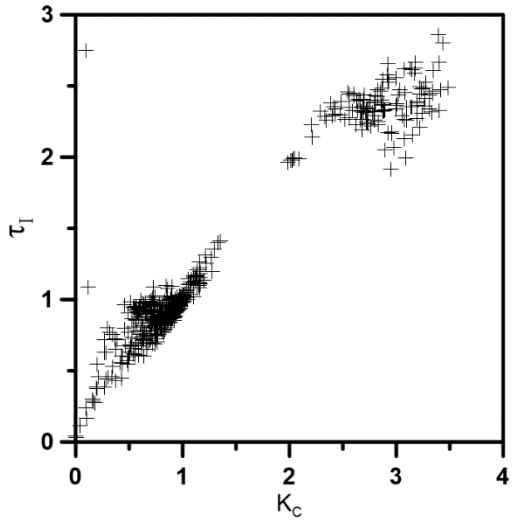

Figure 9 Plot of two decision variables K_C-τ_1 of the Pareto Domain obtained when the crossover parameter α = 0.8 with a constant mutation parameter (ξ_m = 20%).

3.5.2 Mutation Rate and Mutation Operator

Mutation is an additional method to bring more variety within the population in addition to the crossover operation. To investigate the combination of a high mutation rate ξ_m with a high crossover parameter α, experiments were performed with uniform mutation with a mutation rate ξ_m = 60% and a crossover parameter $\alpha = 0.5$. The results of two identical optimization experiments with a different initial random seed, which led to a different initial population, are presented in Figure 10. Comparing with previous Pareto domains in Figure 6b, the combination of a high mutation rate and a high crossover parameter leads to complex results. Firstly, Regions 1 and 4 are found with a high mutation rate ξ_m = 60% and a crossover parameter $\alpha = 0.5$.

Secondly, improvement in finding Region 3 is unstable in Figures 10a and 10b. Thirdly, Regions 5 and 6 become sparse and almost absent in comparison to the Pareto domain of Figure 6b with $\alpha = 0.5$ and a low mutation rate $\xi_m = 20\%$. These results suggest that a high mutation rate alone is not sufficient to provide a stable, well-circumscribed and repeatable Pareto domain. Low mutation rate has also been tested but no important improvements were observed compared with the results of Figure 6b.

To analyze the reasons a high uniform mutation rate does not lead to important improvements in circumscribing a more complete Pareto domain, the evolution history of the population was used to investigate the influence of the mutation rate on the population distribution. Figure 11 shows the difference in the population distribution at the 500^{th} generation based on a mutation with a uniform distribution with mutation rates $\xi_m = 60\%$ and $\xi_m = 100\%$. In Figure 11, population distribution can be divided into two areas. One is the area colored in dark, the dense zone, where the Pareto domain will be situated at the end of the optimization process. The large area outside the dense zone is the area occupied by the random population. Comparing Figures 11a and 11b, a high mutation parameter generates a considerable number of solutions located outside and often very far from the dense zone, i.e. the zone of interest. Through the evolution history, population distributions observed in Figure 11 will be sustained after the transmutation and until the stopping criterion is reached. In other words, no important improvement in the diversity of the population has been achieved using a uniform distribution mutation operator with a high mutation rate. The expectation that non-dominated solutions may be generated through the crossover using points located far from the dense zone which does not improve Pareto domain. If the initial feasible ranges of the decision variables are large, then a large number of dominated solutions will also be created. In addition, even if some solutions were not dominated, solutions in an excessively sparse region would not be worth retaining because they would never be selected for application for the fractional controller due their very low robustness. Thus, it would be worth devising an alternative mutation operator, which could generate offspring individuals surrounding the potential Pareto-optimal regions when the region of interest becomes trivial. A progressive reduction or shrinking of the feasible ranges could represent an alternative method if the uniform mutation operator is used in order to avoid generating solutions that are excessively far from the solution zone of interest.

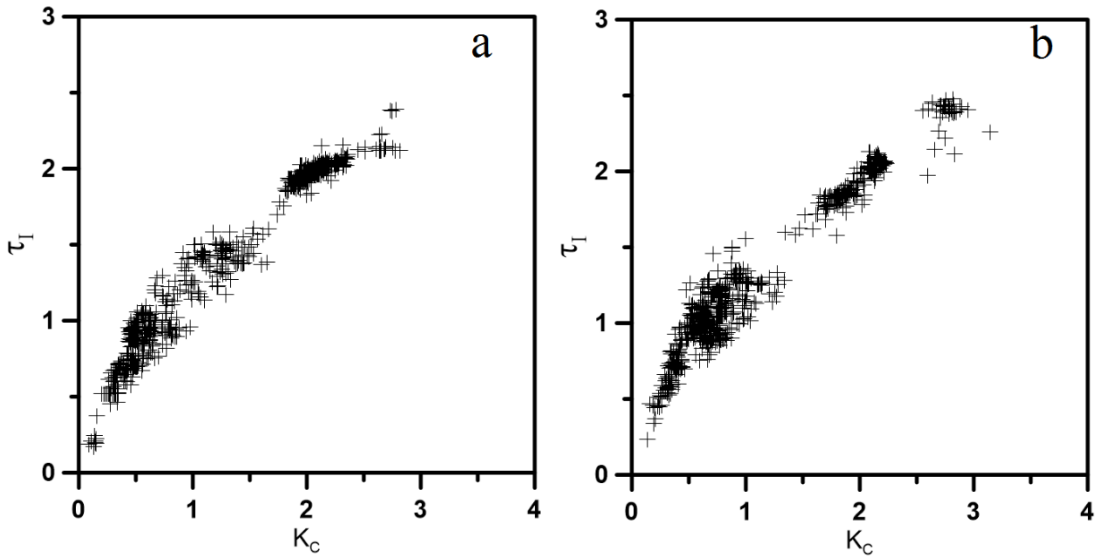

Figure 10 Pareto domain for two separate runs with a mutation rate parameter ξ_m = 60% and a crossover parameter α = 0.5.

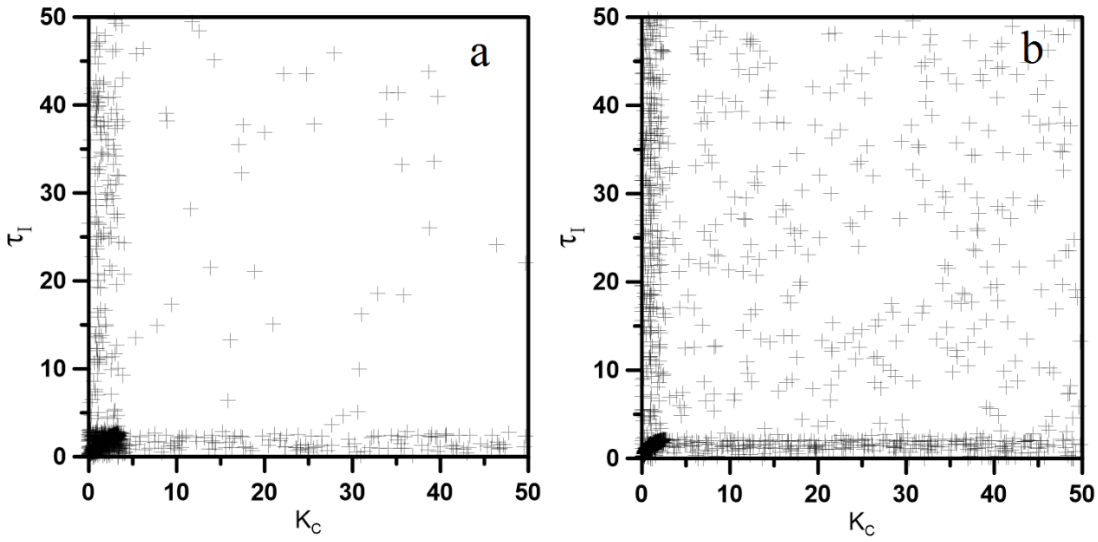

Figure 11 The first two decision variables associated to the population at the 500th generation in search of the Pareto domain for a uniform distribution mutation for ξ_m = 60% (a) and ξ_m = 100% (b).

The previous results suggest that a better mutation operator should generate offspring individuals near the dense Pareto-optimal zone without extending too far from the dense zone. This objective can be fulfilled with a normal distribution mutation operator instead of a uniform distribution. The normal distribution mutation operator has a higher probability to perturb parental individuals near the dense zone. To control the perturbation in the vicinity of the dense zone, the values of the mutation standard deviation σ for the five decision variables are set as small values. For all experiments, σ was set to 0.3, 0.3, 0.1, 0.1, 0.1 for the five decision

variables, respectively. Results are presented in Figure 12 for a mutation rate ξ_m = 60% and ξ_m = 100% while the crossover operator remained constant at α = 0.5. With the normal distribution mutation operator, Regions 5 and 6 (of Figure 2) have now been identified for both mutation rates (Figures 12a and 12b) for the first time with a well-defined boundary. These results indicate that a normal distribution mutation operator improves the ability of the optimization process to find the region(s) characterized with high values of K_C in comparison to the results obtained with a uniform distribution mutation operator.

Considering the magnitude of the mutation rate, the approximation of the Pareto domain with a high mutation rate still shows better performance. The distribution of individuals in the population is more homogeneous when ξ_m = 100%. Unlike for the Pareto domain obtained with ξ_m = 60%, its population did not generate a large number of solutions in a dense zone around the region of K_C = 1. Moreover, the optimization procedure based on a higher mutation rate has found most Regions of the Pareto domain. In fact, only Regions 3 and 8 (see Figure 2) are missing; these regions are the hardest regions of the Pareto domain to find because they are inherently low-density regions.

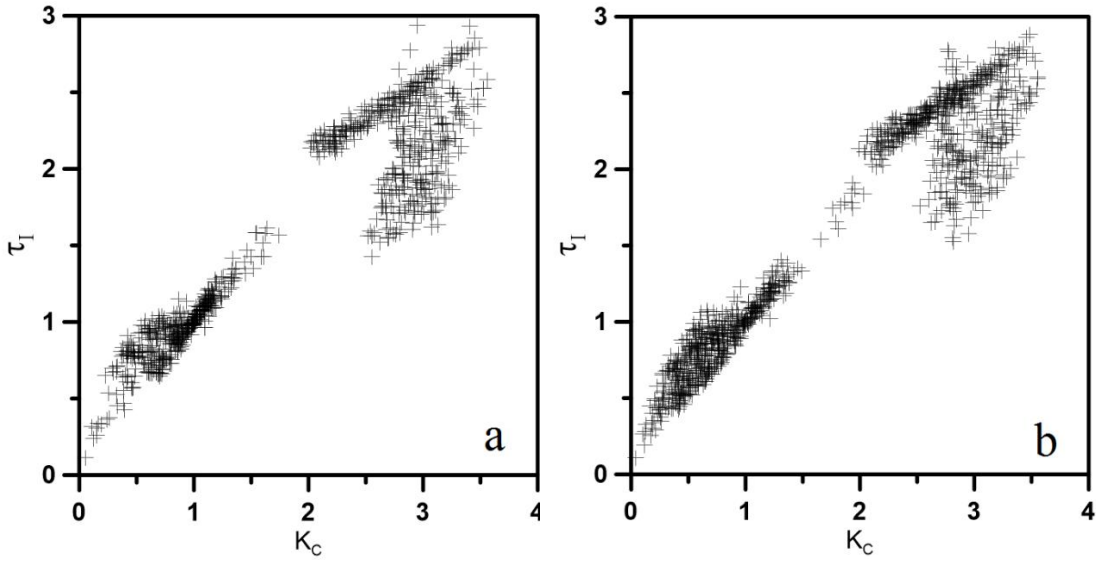

Figure 12 Plot of the first two decision variables K_C-τ_I of the Pareto domain obtained with a normal distribution mutation parameter ξ_m = 60% (a) and ξ_m = 100% (b) while the crossover parameter remained constant at α = 0.5.

3.5.3 Population Size and Decision Space Size

One possible avenue to identify Pareto-optimal solutions with the lowest probability of occurrence is to perform the optimization procedure using a larger population. Figure 13a shows the Pareto domain based on population 10000 individuals, large decision variable range [0.01,50] for K_C and τ_I, [0,50] for τ_D, [0,1] for λ and μ. Other experimental conditions for Figures

13a and 13b, including crossover operator and mutation operator, can be found in Table 1. Even though Regions 3 and 4 have a slightly different shape from the benchmark Pareto domain (Figure 2) and Region 8 is still missing, this Pareto domain based on a high population size still clearly delimits the boundary, especially for Regions 3 and 4, which are naturally sparse regions. Thus, for this case study, it is beneficial to perform the optimization with a larger population. With respect to Region 8, which is naturally very sparse, it can only be obtained by chance.

Instead of optimizing the system using a larger population (i.e. 10000 here), which involves larger computing time, a possible alternative method is to constrain the space of the decision variables to smaller ranges. For example, results of Figure 13b are based on a population of 1000, ranges of K_c and τ_I were decreased from [0.01,50] to [0.01,5], similarly for τ_D [0,5]. Intuitively, it is relatively easier to search Pareto-optimal solutions in a smaller decision space than over a larger range provided the initial ranges contain the Pareto domain. However, it was found that the Pareto domain obtained with smaller ranges and a reduced population size was not sufficiently stable, especially for the low occurrence regions like Regions 3 and 4. A good illustrative example of this behavior is shown in Figure 13b, highly similar to Figure 12a. A good approximation of the Pareto domain is still obtained but without Region 4. However, this method works more efficiently because it converges more rapidly due to the smaller population.

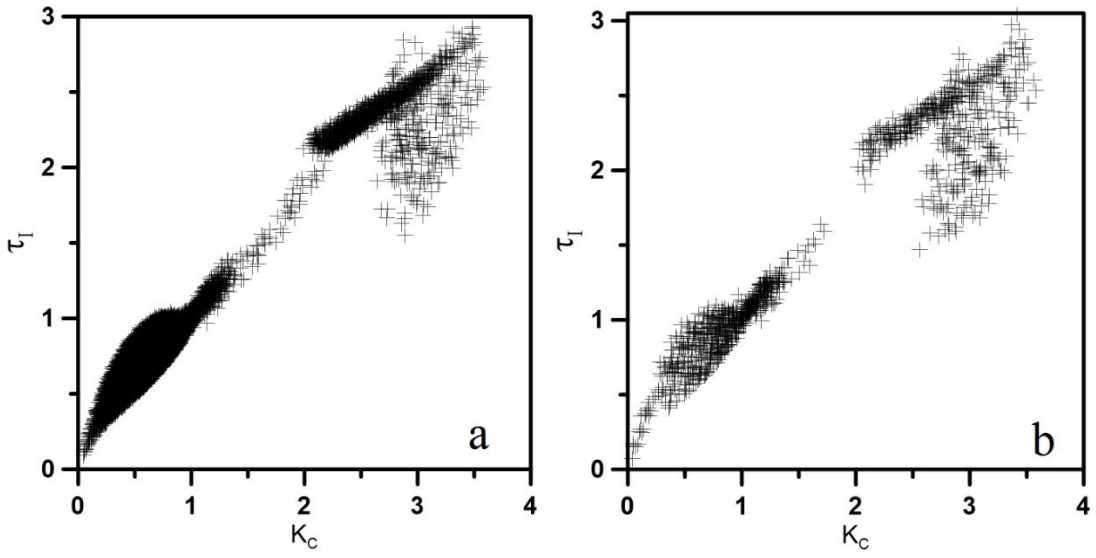

Figure 13 Pareto domain based on a population of 10000 individuals with a larger decision variable space size (a) and a population of 1000 individuals with a smaller decision variable space size (b).

3.5.4 Division of the Objective Space

Niching is a procedure incorporated into a multi-objective optimization to ensure of obtaining and preserving multiple favorable regions of the solution space as well as preventing convergence to a limited number of dense regions. In NSGA-III, reference points and reference

lines are important in the selection of offspring in the niching algorithm. The division of every objective function determines the number of reference lines and their distribution. To examine the influence of the divisions of the objective space, a series of optimization runs were performed with a wide range of divisions and a population of 10000. All optimization runs with different numbers of divisions were compared by determining the percentages of non-dominated solutions based on the total population made of all Pareto-optimal solutions for all separate runs. Results of the different runs that were performed using 2 to 16 divisions, each having a population of 10000, were combined to form a total population of 40000 solutions. All solutions in this combined population were analyzed to determine the percentages of non-dominated solutions associated with each of the different numbers of divisions of the objective functions. Results presented in Table 2 show that the percentages of non-dominated solutions for different divisions, based on the combined population, are essentially the same with percentages ranging from 47.1% to 48.4%. This means there is no prominent advantages to exploit a larger or a smaller number of divisions for every objective function in this case study. On the other hand, increasing the number of divisions leads to a larger number of reference lines and the computational time spent at every generation is much longer as Table 2 reveals. For the tuning of the fractional controller based on the three performance criteria, 4 divisions of the objective, which corresponds to the default value, is suggested.

Table 2 Non-Dominated solutions for different numbers of divisions for the combined population of 40000 individuals.

Number of Divisions for Each Objective	2	4	8	16
Number of Non-Dominated Solutions	4843	4710	4792	4785
Computation Time for 1000 Generations (h)	18.4	46.8	101.1	297.7
Percentage of Non-Dominated Solutions	48.4%	47.1%	47.9%	47.8%

3.6 Conclusion

In this case study, the evolution history analysis was proposed as a practical method to determine user-defined factors for multi-objective optimization. Through the example of fractional controllers tuned by NSGA-III, the evolution history was used to analyze how user-defined factors determine the obtained Pareto domain and to improve the performance of the optimization by adjusting these parameters. Eventually, a repeatable, stable and more complete Pareto domain can be obtained.

In the case study considering the tuning of fractional controllers by NSGA-III, the blend-α crossover with $\alpha = 0.5$ and the normal distribution mutation operator with $\xi_m = 100\%$ mutation rate are recommended. To obtain a complete and well-defined Pareto domain, it is necessary to

resort to a large initial population. For computational efficiency, it is preferable using a smaller number of divisions of the objective function space (2 or 4), resulting in a smaller number of reference lines, in addition to a smaller decision variable range to accelerate convergence.

3.7 Acknowledgement

The authors gratefully acknowledge the financial support for this project provided by the Natural Science and Engineering Research Council of Canada (NSERC).

3.8 Nomenclature

$d(\boldsymbol{s})$	Distance with individual $\boldsymbol{s}$	-
D_m	Decision variable range	-
${}_a D_t^{-\alpha}$	Riemann-Liouville fractional derivative operator with fractional order α with respect time from a to t	[time]$^{-1}$
$\boldsymbol{f}$	Objective function vector	-
G_C	Transfer function of fractional PI$^{\lambda}$D$^{\mu}$ controller	-
G_P	Transfer function of process	-
ISDU	Integral of the squared differences in the controller output	[time]
ITAE	Integral of the time-weighted absolute error	[time]
K_C	Gain of fractional PI$^{\lambda}$D$^{\mu}$ controller	-
K_P	Gain of process	-
L_0	Non-dominated level	-
L_{last}	Last domination level selected	-
N	Size of population	-
$NC(R)$	Niche-count of reference line R	-
OZ	Time of the process controlled variable spent outside the zone ±5% of the setpoint value	[time]
P_t	Parent population at generation t	-
Q_t	Offspring population	-
$R(\boldsymbol{s})$	Reference line R of individual $\boldsymbol{s}$	-
S	Decision space	-
S_t	Offspring population	-
S^*	Decision space subset	-

t	Time	[time]
T_t	Total population combined primitive population with population after operations at generation t	-
Δu_t^2	Square of the change in the controller output at time t	-
$\boldsymbol{x}$	Decision variable vector	-
$\boldsymbol{x}_P$	Parent vector	-
$\boldsymbol{x}_S$	Son vector	-
x'_m	m-th variable in decision vector after mutation	-
α	Parameter of the blend crossover operator	-
$\boldsymbol{\gamma}$	Crossover weight vector	-
γ_m	Crossover weight for the m-th variable in decision vector	-
δx_m	Normally distributed perturbation	-
Δu_t^2	Square of the change in the controller output at time t	-
$\varepsilon(t)$	process response	-
θ	Dead time	[time]
λ	Fractional integral order of fractional $PI^{\lambda}D^{\mu}$ controller	-
μ	Fractional derivative order of fractional $PI^{\lambda}D^{\mu}$ controller	-
μ_m	Expectation of	
ξ_m	Mutation rate for the m-th variable in decision vector	-
τ_D	Fractional derivative constant of fractional $PI^{\lambda}D^{\mu}$ controller	[time]
τ_I	Fractional integration constant of fractional $PI^{\lambda}D^{\mu}$ controller	[time]
τ_P	Time constant of process	[time]
$\boldsymbol{\chi}$	Vector whose elements subjected to uniform distribution	-

3.9 References

[1] T. Wagner, N. Beume, and B. Naujoks, "Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization," in *Evolutionary Multi-Criterion Optimization*, 2007, pp. 742–756.

[2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[3] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the Strength Pareto Evolutionary Algorithm,” 2001.

[4] K. Deb and H. Jain, “An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints,” *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[5] K. Deb, A. Anand, and D. Joshi, “A Computationally Efficient Evolutionary Algorithm for Real-Parameter Optimization,” *Evolutionary Computation*, vol. 10, no. 4, pp. 371–395, Dec. 2002.

[6] L. J. Eshelman and J. D. Schaffer, “Real-coded genetic algorithms and interval schemata,” in *Foundations of genetic algorithms 2*, vol. 2, San Mateo: Morgan Kaufmann, 1993, pp. 187–202.

[7] R. Storn and K. Price, “Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[8] Q. Zhu *et al.*, “A novel adaptive hybrid crossover operator for multiobjective evolutionary algorithm,” *Information Sciences*, vol. 345, no. Supplement C, pp. 177–198, Jun. 2016.

[9] A. Ahrari, “A requirement for the mutation operator in continuous optimization,” *Optim Lett*, vol. 7, no. 8, pp. 1681–1690, Dec. 2013.

[10] K. Deb and M. Goyal, “A combined genetic adaptive search (GeneAS) for engineering design,” *Computer Science and informatics*, vol. 26, pp. 30–45, 1996.

[11] K. Deb and S. Tiwari, “Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization,” *European Journal of Operational Research*, vol. 185, no. 3, pp. 1062–1087, Mar. 2008.

[12] C. C. Da Ronco and E. Benini, “A Simplex Crossover based evolutionary algorithm including the genetic diversity as objective,” *Applied Soft Computing*, vol. 13, no. 4, pp. 2104–2123, Apr. 2013.

[13] I. Podlubny, L. Dorcak, and I. Kostial, “On fractional derivatives, fractional-order dynamic systems and PI lambda;D mu;-controllers,” in *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997, vol. 5, pp. 4985–4990 vol.5.

[14] E. Zitzler, K. Deb, and L. Thiele, “Comparison of Multiobjective Evolutionary Algorithms: Empirical Results,” *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, Jun. 2000.

[15] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable multi-objective optimization test problems,” in *Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC ’02*, 2002, vol. 1, pp. 825–830.

[16] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, “Multiobjective optimization test instances for the CEC 2009 special session and competition,” *University of Essex, Colchester, UK and Nanyang technological University, Singapore, special session on performance assessment of multi-objective optimization algorithms, technical report*, vol. 264, 2008.

Chapter 4
Modelling Adsorption Breakthrough Curves

Xin Shen, Jules Thibault*
Department of Chemical and Biological Engineering
University of Ottawa, Ottawa, Ontario, Canada K1N 6N5

Abstract - Adsorption is an important separation process widely used for its low energy consumption for a myriad of applications such as air separation, water purification, chromatographic analysis, wastewater treatment and protein adsorption on biomaterials. Adsorption and desorption are usually performed in a number of parallel packed bed adsorption columns. To design and optimize adsorption columns, experimental breakthrough curves are usually obtained experimentally to characterize the adsorption capacity and the kinetics of adsorption. In this investigation, five different models based on instantaneous adsorption, non-instantaneous adsorption, Fickian diffusion and anomalous diffusion have been used to predict breakthrough curves for the adsorption of butanol on activated carbons. The first four models are expressed using traditional partial differential equations of different complexity whereas the fifth model was derived based on anomalous diffusion assumption expressed using fractional order differential equations. Results showed that some models better predicted the breakthrough curves for the adsorption of butanol for one type of adsorbent and other models were better for the other adsorbent.

***Keywords*:** Adsorption, modelling, breakthrough curve, fractional partial differential equations

4.1 Introduction

The design and optimization of a packed bed adsorption column for a particular application rely on the determination of two main models: the isotherm equation of the adsorbed species for the selected adsorbent and the dynamic adsorption model for the simulation of the breakthrough curves. The selection of the best adsorbent is made on the basis of numerous criteria such as the adsorption capacity, the kinetics of adsorption, the ease of desorption, the reusability of the adsorbent for a large number of adsorption-desorption cycles, the selectivity of the targeted molecules, and the cost.

For the first model, in spite of recent theoretical development to determine adsorption isotherms such as with molecular dynamics, experimental evaluation still remains the main method to obtain the isotherm equation, i.e. the equilibrium adsorption capacity of the adsorbent for a given species as a function of the concentration of this species in the gaseous or liquid solution. When equilibrium experimental adsorption data are available, the isotherm equation can be obtained by a simple least-squares nonlinear regression of the experimental data to determine the isotherm model such as Langmuir, Freundlich and Sips, which best represents the relationship between the equilibrium adsorption capacity as a function of the concentration of the species in solution. More complex models can be used to express the isotherm such as neural networks [1] or even more complex methods can be used to select the most appropriate isotherm equation such as resorting to nonlinear frequency response [2]. For the design and optimization of the adsorption packed bed, the form of the isotherm expression is not critical as long as it provides an accurate representation of the equilibrium adsorbed amount in contact with the solution.

To simulate the breakthrough curve of a species through a packed bed adsorber, it is necessary to solve the set of differential equations that include the convective flow of the solution through the packed column, the diffusion of each species within the adsorbent particles and the adsorption of species on the surface of the adsorbent. Good correlations exist to estimate the parameters of the set of differential equations such as the mass transfer coefficient, the column axial dispersion, the diffusion coefficient of each species in the particle, and the column void fraction. Some other parameters such as the particle void fraction and the kinetics of adsorption must be determined experimentally. However, most of the time, key model parameters are estimated by fitting the simulated breakthrough curves with the experimental breakthrough curves under different conditions. When the simulated breakthrough curves represent adequately the experimental breakthrough curves, the model can be used with confidence to design and optimize the larger scale packed bed adsorber.

To properly design and optimize an adsorption packed bed system, it is important to use a representative model to simulate the breakthrough curves. To quote Wiechert [3]: "*As a rule, a model should always be as simple as possible and as complex as necessary*". It is therefore

important to modulate the complexity of the model to maximize the representativity of the packed bed adsorber and, at the same time, to minimize the development and computation time. In this investigation, models of different complexity will be developed, tested and compared for the adsorption of butanol from aqueous binary solutions on activated carbon. Experimental adsorption data obtained in our laboratory and data from the literature were used for this comparison. In this investigation, in particular, we are interested to determine if the use of a fractional derivative of the diffusion equation would more adequately represent the breakthrough curves observed in a packed bed system.

The paper is divided as follows. A brief description of the experimental system, materials and methods is first presented followed by the presentation of the various models that were compared in this investigation. The results are then presented and discussed before concluding.

4.2 Materials and Methods

4.2.1 Materials

For the experiments performed in our laboratories, butanol-water solutions were prepared with n-butanol (99 % pure, Acros) and deionized distilled water. The adsorbent used for this study was F-400 activated carbon (AC F-400) purchased from Calgon Corporation (Mississauga, ON, Canada). Some of the AC F-400 properties are given in Table 1.

Table 1 Some properties of the two activated carbon adsorbents used for butanol adsorption

Properties	**AC F-400**	**AC Norit ROW 0.8** [4]
Shape	Granule	Extruded particles
Surface area (m^2/g)	1090	1159
Effective diameter (mm)	0.55-0.75	0.8
Micropore volume (cm^3/g)	0.43	1.04

For the data extracted from the literature, the isotherm and breakthrough data at 37ºC were obtained from the paper of Xue et al. [5] who used activated carbon Norit ROW 0.8 to adsorb butanol. The main properties of AC Norit ROW 0.8 are presented in Table 1.

4.2.2 Methods

For the experiments performed in our laboratory, the butanol isotherm at 25ºC was determined using a packed bed adsorber (0.175 m high and 0.015 m in diameter) containing 20 g of AC F-400 adsorbent using the experimental system of Figure 1. A butanol solution of known initial concentration and volume was continuously circulated through the adsorbent packed bed until equilibrium was achieved. This procedure was repeated with butanol solutions of different initial concentrations. For each experiment, one point on the isotherm curve was determined by

performing a mass balance between the initial and final concentrations whereby the amount of butanol lost from the solution was deemed to be adsorbed by the solids adsorbent [6]. Concentrations were measured by HPLC (Waters, Canada). The detector, pump, and autosampler were Refractive Index Detector (Waters 2414), Isocratic HPLC pump (Waters 1515) and Autosampler (Waters 717 plus), respectively. The experimental data was fitted to a Freundlich isotherm. For the literature data [5], the butanol isotherm at 37°C were obtained in flask experiments using various amounts of adsorbents, ranging from 0.25 g to 6 g, added to 15 mL of a 40g/L aqueous butanol solution and allowed to equilibrate for 24 h with agitation at 150 rpm.

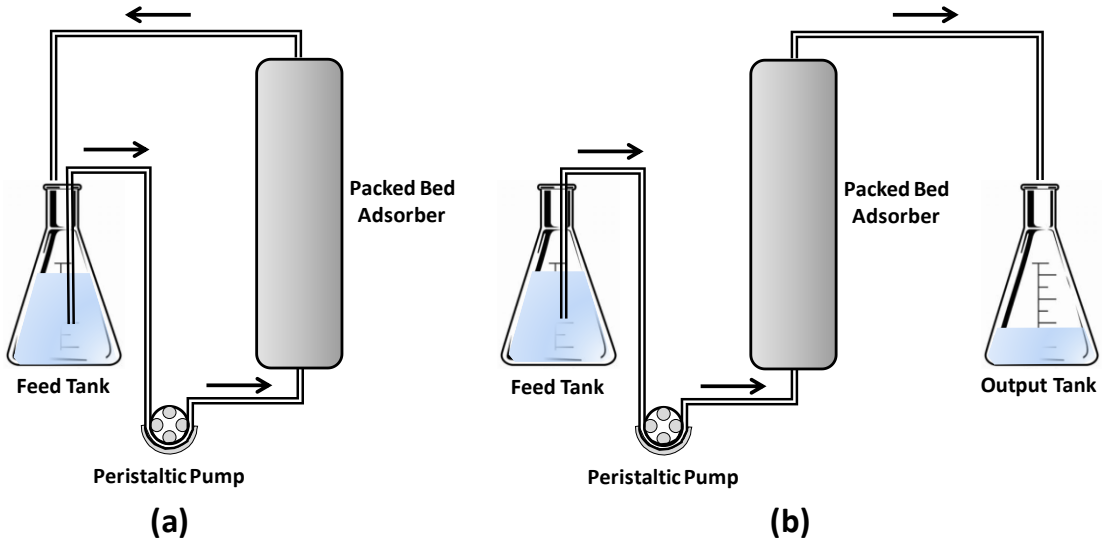

Figure 1 Schematic diagram of the adsorption setup used for (a) isotherm experiments and (b) breakthrough experiments.

Breakthrough experiments were performed in our laboratory using an adsorption packed bed (stainless steel column 0.195 m in height and 0.02 m in diameter) containing 39 g of AC F-400 adsorbent. The inlet solution had a butanol concentration between 13-14 g/L and three different flow rates were used [6]. Since there was an uncertainty as to the actual solution flow rate, the flow rates were corrected in such a way that the predicted amount of butanol adsorbed corresponded to the amount of butanol adsorbed experimentally. In the breakthrough experiments, the feed solution was pumped to the column using a peristaltic pump and samples of the effluent were taken at specific time intervals until the adsorbent bed reached complete saturation. The concentration of butanol was determined using HPLC. Integrating the area above the breakthrough curve to determine the equilibrium adsorption capacity of butanol provided an additional isotherm point (Abdehagh et al., 2014b). For the literature data, experiments were performed in a similar way and data from the breakthrough curve were extracted using a digitalizing software (GRABIT) for the experimental conditions reported in

Xue et al. [5]. For the butanol breakthrough curve, a feed solution of 10 g/L was fed at a flow rate of 20 mL/min in a glass packed column (0.15 m high and 0.021 m in diameter) containing 25 g of activated carbon Norit ROW 0.8.

4.3 Description of the different models

4.3.1 Problem statement

A packed bed of length L_C and diameter D_C is filled with porous solid adsorbent particles of uniform radius R as illustrated schematically in Figure 2. An aqueous solution with an initial concentration C_{L0} of species A and flow rate F enters the packed adsorption bed at a constant superficial velocity u_S at time zero. The solution flows in the void space of the packed bed ε_C at an interstitial velocity u_L toward the exit of the column. A fraction of species A is transferred convectively from the bulk of the flowing solution to the external surface of the particles. Species A then diffuses in the mobile phase occupying the void fraction of the particles ε_P, where it is progressively adsorbed. The concentration of species A in the mobile phase inside the particle is denoted as C_P whereas the concentration of species A of the adsorbed phase is C_A. The density of an adsorbent particle is ρ_P and the density of the solid phase is ρ_S, where ρ_P is equal to $\rho_S(1 - \varepsilon_P)$.

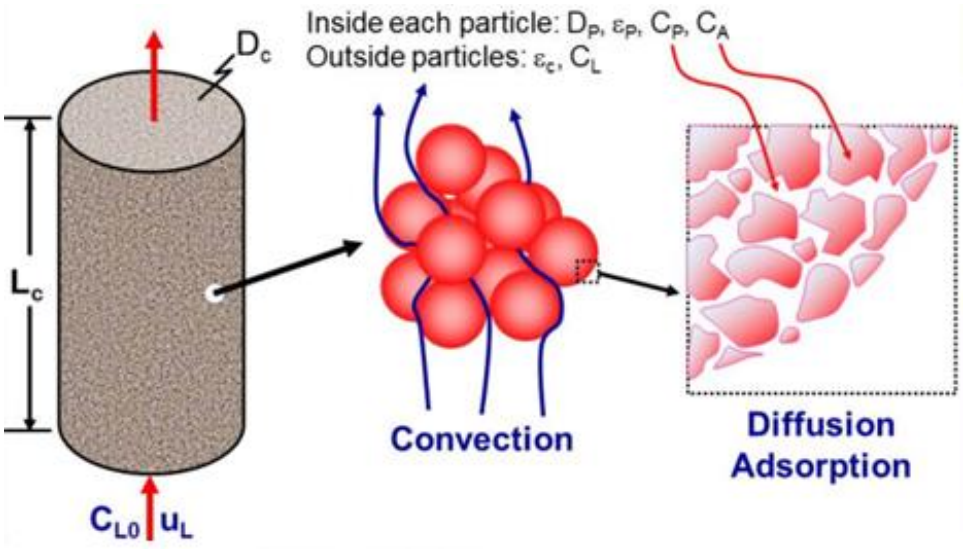

Figure 2 Packed bed filled with adsorbent porous particles

To simulate the breakthrough curve, it is required to calculate the column exit concentration of species A as a function of time. Different models of various levels of complexity may to be used to obtain this information. Five different models, used in this investigation, are presented in the next section.

4.3.2 Different Mathematical Models

4.3.2.1 Model 1

In Model 1, the temporal variation of the concentration of species A at any position in the bulk solution occupying the void space ε_C of the adsorption packed bed, as described in Eq. (1), accounts for the axial dispersion, the bulk flow through the column and convective mass transfer from the bulk to the surface of the particles.

$$\varepsilon_C \frac{\partial C_L}{\partial t} = \varepsilon_C E \frac{\partial^2 C_L}{\partial z^2} - \varepsilon_C u_L \frac{\partial C_L}{\partial z} - k_L a\left(C_L - C_L^*\right) \tag{1}$$

where k_L is the mass transfer coefficient of species A, E is the axial diffusion coefficient of species A, and a is the volumetric surface area of the packed bed. C_L is the concentration of species A in the mobile phase of column and C_L^* is the hypothetical concentration of species A in the solution that would be in equilibrium with the concentration of the adsorbed phase. Eq. (1) is the basic equation that applies to all models used in this investigation.

In Model 1, it is assumed that species A is adsorbed instantaneously when it reaches the surface of the particles in a way that it becomes immediately in equilibrium with the concentration of species A in the solution at the surface of the particle. Therefore, the amount of species A adsorbed by the adsorbent particles is equivalent to the amount convectively transferred to the particle. This relationship is governed by Eq. (2).

$$\left(1-\varepsilon_C\right)\rho_P \frac{\partial C_A}{\partial t} = k_L a\left(C_L - C_L^*\right) \tag{2}$$

Eqs, (1) and (2) constitute the two differential equations used for Model 1. In the extreme case where the volumetric mass transfer coefficient $K_L a$ is very large, it would be possible to combine the two equations into a single equation.

$$\varepsilon_C \frac{\partial C_L}{\partial t} = \varepsilon_C E \frac{\partial^2 C_L}{\partial z^2} - \varepsilon_C u_L \frac{\partial C_L}{\partial z} - \left(1-\varepsilon_C\right)\rho_P \frac{\partial C_A}{\partial t} \tag{3}$$

Eq. (3) can be modified to express it in terms of C_L only.

$$\varepsilon_C \frac{\partial C_L}{\partial t} = \varepsilon_C E \frac{\partial^2 C_L}{\partial z^2} - \varepsilon_C u_L \frac{\partial C_L}{\partial z} - \left(1-\varepsilon_C\right)\rho_P \frac{\partial C_A}{\partial t}\frac{\partial C_L}{\partial C_L} \tag{4}$$

$$\left[\varepsilon_C + \left(1-\varepsilon_C\right)\rho_P \frac{\partial C_A}{\partial C_L}\right]\frac{\partial C_L}{\partial t} = \varepsilon_C E \frac{\partial^2 C_L}{\partial z^2} - \varepsilon_C u_L \frac{\partial C_L}{\partial z} \tag{5}$$

The term $\partial C_A/\partial C_L$ is the slope of the adsorption isotherm equation evaluated at concentration C_L.

4.3.2.2 Model 2

Model 2 shares the same assumptions as Model 1. The only difference is in the expression of the convective mass transfer term where it is expressed in terms of the concentration of the adsorbed phase concentration C_A instead of the liquid concentration C_L. For instance, in the case where the Freundlich model describes the adsorption isotherm:

$$C_A = K\left(C_L\right)^{\frac{1}{n}} \tag{6}$$

where K and n are the isotherm constants for species A for the given adsorbent. By isolating C_L in Eq. (6) and substituting its inverse Freundlich expression into Eq. (1), Eq. (7) is obtained.

$$\varepsilon_C \frac{\partial C_L}{\partial t} = \varepsilon_C E \frac{\partial^2 C_L}{\partial z^2} - \varepsilon_C u_L \frac{\partial C_L}{\partial z} - \frac{k_L a}{K^n}\left(\left(C_A^*\right)^n - \left(C_A\right)^n\right) \tag{7}$$

However, Eq. (7) is often linearized with respect to C_A and an effective mass transfer coefficient K_{eff} is defined as shown in Eq. (8). This equation has the advantage that the isotherm equation does not need to be inverted as in Model 1 to calculated C^*_L.

$$\varepsilon_C \frac{\partial C_L}{\partial t} = \varepsilon_C E \frac{\partial^2 C_L}{\partial z^2} - \varepsilon_C u_L \frac{\partial C_L}{\partial z} - K_{eff}\left(C_A^* - C_A\right) \tag{8}$$

4.3.2.3 Model 3

In Model 3, Eq. (1) is used as in Models 1 and 2 to determine the temporal and spatial variation of the concentration of species A in the bulk liquid phase of the adsorption packed column. However, Model 3 does not assume instantaneous diffusion and adsorption to reach equilibrium with the concentration of the solution at the boundary of the particles. Instead, species A diffuses inside the mobile phase of the adsorbent particles where it is adsorbed progressively at a rate that is finite and proportional to a concentration driving force. The rate of change of the concentration C_P of species A in the mobile phase inside the pores of spherical adsorbent particles is given by Eq. (9). The rate of change of C_P considers the rate of diffusion and the rate of adsorption.

$$\varepsilon_P D_e \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial C_P}{\partial r}\right) - k_{Ads} a_P \left(C_P - C_P^*\right) = \varepsilon_P \frac{\partial C_P}{\partial t} \tag{9}$$

where D_e is effective diffusion coefficient of species A in the pores of adsorbent particle, r is the radial location in the adsorbent particle and k_{Ads} is the adsorption kinetic rate constant.

Mass conservation dictates that the amount of species A adsorbed on the solid phase of the adsorbent particles is equal to the amount of species A lost in the mobile phase within the pores of the adsorbent particles, as expressed in Eq. (10):

$$\left(1-\varepsilon_P\right)\rho_S\frac{\partial C_A}{\partial t}=k_{Ads}a_P\left(C_P-C_P^*\right) \tag{10}$$

4.3.2.4 Model 4

Model 4 is identical to Model 3 except that the adsorption dynamic of species A between adsorbate in the mobile phase inside the pores of the adsorbent is assumed instantaneous. This implies that the adsorption kinetic rate constant k_{Ads} is very high. To circumvent using small integration time steps in the solution of Eq. (9) when the adsorption kinetic rate constant is very high, Eq. (10) can be used to transform Eq. (9) into Eq. (11)-(13) to include the instantaneity of adsorption in the mobile phase inside the pores of the particles.

$$\varepsilon_P D_e\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial C_P}{\partial r}\right)-\left(1-\varepsilon_P\right)\rho_S\frac{\partial C_A}{\partial t}=\varepsilon_P\frac{\partial C_P}{\partial t} \tag{11}$$

$$\varepsilon_P D_e\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial C_P}{\partial r}\right)=\varepsilon_P\frac{\partial C_P}{\partial t}+\left(1-\varepsilon_P\right)\rho_S\frac{\partial C_A}{\partial t}\frac{\partial C_P}{\partial C_P} \tag{12}$$

$$\varepsilon_P D_e\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial C_P}{\partial r}\right)=\left(\varepsilon_P+\left(1-\varepsilon_P\right)\rho_S\frac{\partial C_A}{\partial C_P}\right)\frac{\partial C_P}{\partial t} \tag{13}$$

where $\partial C_A/\partial C_L$ is the slope of the adsorption isotherm equation evaluated at mobile phase concentration C_P of species A inside the particles. If the adsorption isotherm can be described by Freundlich model as expressed in Eq. (6), the local derivative can be calculated using Eq. (14).

$$\frac{\partial C_A}{\partial C_P}=\frac{K}{n}C_P^{\left(\frac{1}{n}-1\right)} \tag{14}$$

4.3.2.5 Model 5

For Models 1 to 4, the diffusion inside the adsorbent particles was characterized by Fick's second law, which is usually used in the description of "common" diffusion. This type of diffusion is referred to as Fickian diffusion which was proposed in 1855 [7]. Although Fick's first and second laws of diffusion provide essential equations to predict the concentration of a diffusing species on a macroscopic point of view, the corresponding microscopic description of the mechanism of diffusion was not considered until 1905 when Einstein published his famous work about the Brownian motion of molecules, which was verified experimentally by Perrin [8]. Einstein successfully derived the Fick's second law from a microscopic perspective based on a novel idea, which is now referred to as a stochastic process. This idea was generalized by the continuous time random walk process model (CTRW), which assumes that for every independent solute particle in Brownian motion, particle jump length x is governed by a probability density function $\lambda(x)$ and particle waiting time between consecutive jumps τ is

governed by a probability density function $w(\tau)$ microscopically [9] [10]. If $\lambda(x)$ obeys Gaussian probability density function and $w(\tau)$ obeys Poisson probability density function, the CTRW model allows to derive the Fick's second law from a microscopic perspective. Both the CTRW and Einstein's model for the Brownian motion uncover a Markovian characteristic of the Fick's diffusion. Einstein suggested in 1905 that the mean squared displacement of a particle or molecule is proportional to the product of the diffusion coefficient D and the time t: $<x^2(t)> \sim 2Dt$ [9]. However, through different experiments [11]–[14], the mean squared displacement of a particle in some process was not proportional to time. Diffusion processes that do not obey a proportional relationship with time are referred to as anomalous diffusion or non-Markovian diffusion. Some anomalous diffusion experiments found that the mean squared displacement of particles could be expressed with $<x^2(t)> \sim 2Dt^{\alpha}/\Gamma(1+\alpha)$. To describe such anomalous diffusion, the power law distribution was used to replace the Poisson probability density function for $\lambda(x)$ in the CTRW model. Power law distribution has a long tail, which means the particle waiting time between consecutive jumps can be theoretically infinite. When α ranges from 0 to 1, the associated type of diffusion is defined as subdiffusion, which means the diffusion is slower than the Fickian diffusion for a given diffusivity. Inversely, when α ranges from 1 to 2, such diffusion is defined as superdiffusion that is quicker than Fickian diffusion. This property is usually interpreted as there exists the probability that a particle may be trapped in the diffusion media. From the CTRW model, the corresponding macroscopic fractional diffusion law was obtained by introducing fractional calculus to replace classical calculus as shown in Eq. (15). In Eq. (15), C is the concentration of the adsorbate at position x and time t. D is the fractional diffusion coefficient and ${}_0D_t^{1-\gamma}$ is the factional partial differential operator with respect to time whose order is 1 - γ. The physical interpretation of Eq. (15) is that the rate of change of concentration occurs because of anomalous diffusion.

$$\frac{\partial C}{\partial t} = {}_0D_t^{1-\gamma}\left[D\frac{\partial^2 C}{\partial x^2}\right] \tag{15}$$

In Model 5, the diffusion in the bulk of the adsorption packed bed is assumed as Fickian diffusion. Thus, the dispersion in a packed bed can be described using Eq. (1). However, the diffusion inside the adsorbent particles is considered as anomalous diffusion [15], [16]. In addition, the rate of adsorption can be considered finite as in Model 3 or instantaneous as in Model 4. In this investigation, Model 5 considers instantaneous adsorption. In Model 5, it is assumed that Brownian motion with particle jump length governed by Gaussian probability density function and particle waiting time governed by a long-tail distribution function describes the anomalous diffusion inside adsorbent particles. There exist mathematical tools to describe anomalous diffusion. Fractional partial differential equations (FPDE) can macroscopically represent adequately the dynamics of anomalous diffusion and are flexible

with respect to boundary conditions. Therefore, to describe anomalous diffusion and instantaneous adsorption inside adsorbent particles, the diffusion term of Eq. (13) can be replaced with a FPDE to give Eq. (16) [9], [17]. The main difference lies in the substitution of the integer partial differential operator by a fractional partial differential operator whose order is 1 - γ.

$$ {}_0D_t^{1-\gamma}\left[\varepsilon_P D_e \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial C_P}{\partial r}\right)\right]=\left(\varepsilon_P+\left(1-\varepsilon_P\right)\rho_S\frac{\partial C_A}{\partial C_P}\right)\frac{\partial C_P}{\partial t} \tag{16} $$

where D_e is the effective fractional diffusion coefficient in the adsorbent particle with units of m^2/s^γ, ${}_0D_t^{1-\gamma}$ the Riemann–Liouville fractional derivative with fractional order 1-γ between time 0 and time t. The definition of the Riemann–Liouville fractional derivative of function $f(x,t)$ is given in Eq. (17).

$$ {}_0D_t^{1-\gamma} f\left(x,t\right)=\frac{1}{\Gamma\left(\gamma\right)}\frac{\partial}{\partial t}\int_0^t\frac{f\left(x,\tau\right)}{\left(t-\tau\right)^{1-\gamma}}d\tau \tag{17} $$

where 1-γ is the order of the fractional derivative and Γ is the gamma function.

4.3.3 Initial and boundary conditions

To solve the set of partial differential equations in these models, proper initial and boundary conditions need to be specified. It is assumed that, at time t equal to zero, the concentrations of species A in the liquid occupying the void of the column (C_L), in the mobile phase inside the pores of the adsorbent particles (C_P), and adsorbed on the particles (C_A) are all equal to zero. Then, at time t equal to zero, the concentration of the feed solution is subjected to a concentration step change from zero to C_{L0} at the entrance ($z = 0$) of the packed bed. Therefore, the initial conditions can be summarized by the following equations.

$$ \forall \text{z and } \forall \text{r},\ t=0 \quad \begin{matrix} C_L=0 \\ C_P=0 \\ C_A=0 \end{matrix} \tag{18} $$

$$ t=0^+,\ z=0 \qquad C_L=C_{L0} \tag{19} $$

Because of the presence of the axial dispersion term in Eq. (1), two boundary conditions are required. It is assumed that there is no diffusion at the entrance and exit of the packed bed such that the boundary conditions can be expressed by Eq. (20):

$$ z=0 \qquad \frac{\partial C_L}{\partial z}=0 \tag{20} $$

$$z = L_C \qquad \frac{\partial C_L}{\partial z} = 0 \tag{21}$$

Similarly, the second order differential equation (Eq. (9)) accounting for the diffusion of species A requires two boundary conditions. At the surface of the particle, the rate of convective mass transfer of species A from the bulk of the liquid solution to the surface of the particle must equal the rate of diffusion of species A inside the particles, as expressed in Eq. (22).

$$r = R \quad -D_e \frac{\partial C_P}{\partial r} = k_L \left(C_L - C_P\big|_{r=R} \right) \tag{22}$$

where $C_P\big|_{r=R}$ is the concentration of species A in the solution at the interface of the particle. At the center of the particle, symmetry condition prevails.

$$r = 0 \quad \frac{\partial C_P}{\partial r} = 0 \tag{23}$$

4.4 Numerical Methods

For the solution of partial differential equations, analytical solutions may not exist or are sometimes very difficult to derive such that one must resort to numerical methods. In this investigation, standard finite difference method was adopted to solve the partial differential equations for Models 1 to 4, where the governing equation of dispersion in the column was discretized by an explicit finite difference method and the governing partial differential equation of diffusion inside the adsorbent particle was solved by an implicit finite difference method.

The set of partial differential equations of Model 5 was solved similarly to Model 4 except for that special treatment is required to solve the fractional partial differential equation (FPDEs) given in Eq. (16). In this investigation, the Grünwald–Letnikov approximation was used to approximate the Riemann–Liouville fractional derivative term in the fractional partial differential equation governing the diffusion in the adsorbent particles. For solving the equations by finite differences, with a time step of Δt, the derivative of order (1-γ) of a function *f(t)* is calculated as the weighted sum of the past values of the function *f(t)* at time *t* using the Grünwald–Letnikov approximation of the Riemann–Liouville fractional derivative with fractional order 1-*γ* as shown in Eqs. (24) and (25) [17], [18].

$${}_0D_t^{1-\gamma} f(t) \approx \frac{1}{\Delta t^{1-\gamma}} \sum_{j=0}^{N} \omega(j) f(t - j\Delta t) \tag{24}$$

$$\omega(j) = (-1)^j \binom{1-\gamma}{j} = \frac{\Gamma(2-\gamma)}{\Gamma(j+1)\Gamma(2-\gamma-j)} \tag{25}$$

where N is the number of past values of *f(t)* over which the weighted summation is performed, $\omega(j)$ is fractional weight; Γ is the gamma function. Therefore, Eq. (24) can be interpreted as a weighted average of function $f(t)$ at different time interval, which contain historical information of function $f(t)$.

Compared to the solution of integer partial differential equations, the Grünwald-Letnikov approximation for fractional partial differential equations is significantly more computationally intensive because of the number of terms in the summation increases with time. If N is very large, the fractional derivative can be interpreted as a memory of the complete history of the function $f(t)$. However, using the short memory principle, it possible to perform the summation using a reduced number of terms to calculate the Grünwald–Letnikov approximation [19]. The short memory principle can therefore reduce the computation time necessary to solve the FPDE. The short memory principle is valid since, by virtue of the exponentially decreasing weighting term $\omega(j)$, the value of the fractional derivative is impacted more heavily by the most recent terms of *f(t)*. Indeed, mathematically speaking, the absolute fractional weights are larger for smaller j (close history) than larger j (remote history). Using the short memory principle, Eq. (24) can be approximated with Eq. (26).

$$ {}_0D_t^{1-\gamma} f(t) \approx {}_{t_M} D_t^{1-\gamma} f(t) = \frac{1}{\Delta t^{1-\gamma}} \sum_{j=0}^{M} \omega(j) f(t - j\Delta t) \qquad M \le N \tag{26} $$

As Eq. (26) shows, the memory of complete history was approximated by history from t_M to t. M, the length of the moving approximation window, is the number of past values used in the evaluation of the derivative. In this paper, a window length of 100 was used such that the past 100 concentrations corresponding to the butanol concentrations for each position within the column and within the particles of the last 100 time steps.

4.5 Results and discussion

4.5.1 Experimental results with activated carbon F-400

Isotherm experiments and breakthrough experiments for the adsorption of butanol in binary aqueous solution with activated carbon F-400 were performed to obtain the experimental isotherm curve and breakthrough curves. The isotherm experimental data were fitted using the Freundlich model (Eq. (6)) for butanol adsorption using activated carbon F-400 [20]. The parameters of the isotherm for the adsorption of butanol on the activated carbon AC F-400 are 0.189 kg butanol/kg adsorbent and 6.63 for K and n, respectively. A series of breakthrough experiments of a butanol-water binary solution (13-14 g/L) were performed at three different inlet flow rates. Results for the experimental and simulated breakthrough curves are presented in Figure 3 for the three flow rates tested for the adsorption column of activated carbon F-400.

As expected, a decrease in flow rate leads to a breakthrough curve that is delayed. Experimentally, the shape of the breakthrough concentration profiles for the three flow rates are very similar with a sharp increase when the adsorption column is nearing saturation.

Based on experimental results of isotherms and breakthrough curves obtained, key model parameters including, namely k_L, E, D_e, k_{ADS}, and γ, were determined by minimizing the summation of the squares of the errors (SSE) between the model prediction and experimental data. Table 2 lists the key parameters found with different models. The simulated concentration profiles predict accurately the breakthrough curves of butanol. Here, for simplicity, only the breakthrough curves simulated by Model 3 and the experimental data for different flow rates are presented in Figure 3.

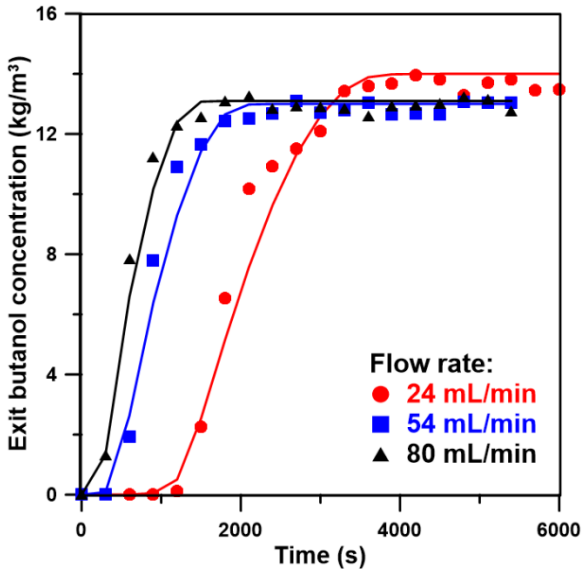

Figure 3 Breakthrough curves of butanol with AC F-400 for three different flow rates. The continuous lines and the symbols correspond to the simulated (using Model 3) and experimental breakthrough curves, respectively.

Models 1 and 2 are mathematically equivalent. The only difference is in the expression used to represent the convective mass transfer coefficient. It is therefore not surprising that an identical SSE is obtained. Because Models 1 and 2 only have two degrees of freedom, namely the mass transfer coefficient k_L and the axial dispersion E, to fit the experimental data, their SSEs are larger than the other three models where the diffusion within the particles is taken into consideration. Based on all experiments that were performed to fit the breakthrough curves with all the models, it was determined that the axial dispersion coefficient E had a negligible effect on the breakthrough curves and could realistically be dropped from the equation. Results of Table 2 clearly show that to adequately model the butanol breakthrough curves, the diffusive resistance within the particles needs to be considered in addition to the convective mass transfer coefficient.

The comparison of the SSE values for Models 3 and 4 suggests that the assumption of instantaneous adsorption may not prevail for the adsorption of butanol on AC F-400. Indeed, Model 3, which assumes non-instantaneous adsorption within the adsorbent particles, has a significantly smaller SSE for the three different flow rates. In addition, Models 1, 2 and 4, which assumes instantaneous adsorption and corresponding to infinitely high kinetics of adsorption k_{ADS}, have a much larger SSE. Therefore, the adsorption of butanol on AC F-400 appears to be controlled by both the diffusion of butanol within the activated carbon adsorbent and the kinetics of adsorption. Model 3 can therefore be a good model to represent the adsorption of butanol on AC F-400.

Model 5 also has smaller SSE values in the same order of magnitude of Model 3 for the three different flow rates. Model 5 provides an alternative explanation of the slower adsorption. Given the small values of the fractional order associated with the minimum SSE, it is hypothesized that the mass transfer process within the particles is controlled by subdiffusion instead of Fickian diffusion while adsorption could be considered instantaneous. These results may suggest that the reason of the slower mass transfer inside the particle may not result from slow adsorption but from a subdiffusion process. When the diffusion inside particles is slower than the rate of adsorption on the solid surface, the apparent rate of adsorption is strongly influenced by the subdiffusion. The subdiffusion inside adsorbent particles may suggest that the tortuosity and the complex network of pores make butanol molecules trapped and rest for a longer time before they jump to another location. This increase in the solute particle waiting time between subsequent jumps results in subdiffusion. Mathematically speaking, the influence of the slower mass transfer can be compensated by either a smaller kinetic rate constant of adsorption k_{ADS} or a smaller derivative order γ. In Model 5, the derivative parameter γ is about 0.16, which is smaller in comparison to Fickian diffusion order of unity.

Table 2 Model parameters giving the minimum SSE for breakthrough butanol adsorption on activated carbon F-400 for the three different flow rates

Model	k_L (m/s)	E (m^2/s)	D_e (m^2/s)	k_{ADS} (m/s)	γ (-)	SSE
F = 24 mL/min						
1	1.18×10^{-5}	2.00×10^{-9}	-	-	-	47.26
2	1.20×10^{-5}	2.00×10^{-9}	-	-	-	47.26
3	5.70×10^{-4}	1.00×10^{-10}	1.20×10^{-8}	1.00×10^{-2}	-	12.83
4	1.00×10^{-5}	1.00×10^{-10}	2.30×10^{-7}	-	-	23.85
5	1.47×10^{-4}	1.00×10^{-10}	1.01×10^{-8}	-	0.16	13.66
F = 54 mL/min						
1	2.50×10^{-5}	2.00×10^{-9}	-	-	-	18.29
2	2.50×10^{-5}	2.00×10^{-9}	-	-	-	18.29
3	2.90×10^{-4}	1.00×10^{-10}	2.00×10^{-8}	2.50×10^{-2}	-	5.94
4	2.60×10^{-5}	1.00×10^{-10}	3.00×10^{-8}	-	-	7.10
5	1.62×10^{-4}	1.00×10^{-10}	9.31×10^{-9}	-	0.17	6.85
F = 80 mL/min						
1	2.50×10^{-5}	2.00×10^{-9}	-	-	-	14.49
2	2.55×10^{-5}	2.00×10^{-9}	-	-	-	14.49
3	3.80×10^{-4}	1.00×10^{-10}	2.60×10^{-8}	1.26×10^{-2}	-	3.69
4	1.50×10^{-3}	6.00×10^{-10}	3.00×10^{-8}	-	-	4.68
5	1.81×10^{-4}	1.00×10^{-10}	1.00×10^{-8}	-	0.16	2.25

The model parameters of Table 2 were obtained independently for the three experiments performed at different flow rates. Different parameters, albeit close to each other, were obtained for the three experiments. However, the diffusion coefficient D_e, the kinetic rate constant of adsorption k_{ADS} and the derivative order γ should not depend on the liquid bulk flow rate. In addition, the convective mass transfer coefficient k_L could be correlated with the liquid bulk flow rate to render the models more general. Therefore, new sets of model parameters were determined in order to minimize simultaneously the three sums of squares of the errors between the predicted and experimental butanol concentrations for the three flow rates. Unique values of D_e, k_{ADS} and γ for the three flow rates were used and the convective mass transfer coefficient k_L was correlated as a function of the superficial velocity using Eq. (27).

$$k_L = C\, u_s^m \tag{27}$$

The new sets of parameters are presented in Table 3. Since the effect of the axial dispersion E was found to be negligible, it was set to a constant values of 1.0×10^{-10} m^2/s. The summation of the three sums of squares of the errors is obviously larger that the sum of the individual SSE of Table 2. However, the difference is small for Models 1 and 2 with an 11% increase in the total SSE. For Model 3, the difference is much larger with a value of 90% whereas it is 35% for Model 4. For Model 5, the difference is about 27% larger.

In the new sets of model parameters presented in Table 3, the best Model is Model 5, which has the smallest total SSE at 29.1; other models have larger total SSE. It is hypothesized that Model 5 shows good performance because it accounts for the slow adsorption characteristics of AC F-400 for butanol. The smaller value of γ at 0.16, implies that subdiffusion within the adsorbent mainly compensates the slow adsorption characteristics.

Model 3 based on non-instantaneous adsorption assumption has a larger total SSE at 43.3 when correlation Eq. (27) is applied. Without correlation Eq. (27), Model 3 has a smaller individual SSE ranging from 3.69 to 12.83 (Table 2). However, it is reasonable to use a correlation to account for the convective mass transfer coefficient varying with the liquid bulk flow rate and to find a unique parameter for the diffusion coefficient D_e, the kinetic rate constant of adsorption k_{ADS} and the derivative order γ. Parameters obtained in Table 3 for the combined optimization have a clearer physical meaning than the parameters of Table 2. The possible explanation for the larger total SSE of Model 3 is that the slow adsorption of butanol on AC F-400 is due to subdiffusion instead of the slower adsorption kinetics. To sum up, Model 5 with subdiffusion charateristics appears to be the best model for predicting the adsorption of butanol on AC F-400 during breakthrough experiments. Nevertheless, Models 3 and 4 could be used to avoid the more complex and computationally intensive fractional order models.

Table 3 Model parameters giving the combined minimum SSE for the breakthrough curve of butanol adsorption on activated carbon F-400 for the three different flow rates

Model	C (m/s)	m	D_e (m^2/s)	k_{ADS} (m/s)	γ (-)	Σ SSE
1	3.76×10^{-4}	0.68	-	-	-	90.0
2	3.76×10^{-4}	0.68	-	-	-	90.0
3	4.44×10^{-4}	0.17	7.19×10^{-10}	1.45×10^{-2}	-	43.3
4	9.62×10^{-5}	0.46	3.10×10^{-8}	-	-	37.9
5	4.44×10^{-4}	0.17	7.98×10^{-9}	-	0.16	29.1

4.5.2 Experimental results with activated carbon Norit ROW 0.8

Experimental data of the isotherm and the breakthrough curves for the adsorption of butanol on activated carbon Norit Row 0.8 obtained from the literature [5] was fitted in the same way as the data for AC F-400. The isotherm data was fitted to the Freundlich model (Eq. (6)) and the resulting isotherm parameters K and n were 0.1932 and 3.72, respectively. With this isotherm model, it was then possible to determine the best model parameters that would minimize the sum of squares of the errors between the experimental and simulated breakthrough curves for the five different models.

The process to search for the group of model parameters that minimize the SSE to obtain the best possible fitting of the experimental data and the sensitivity of the model parameters in the vicinity of the optimal solution are illustrated in Figures 4 and 5. Figure 4a describes the relationship between the SSE and the convective mass transfer coefficient k_L and the diffusion coefficient D_e in the vicinity of the best fit for Model 3. As D_e increases, the SSE drops rapidly and then increases smoothly showing an optimum diffusion coefficient at approximately 4×10^{-9} m^2/s. On the other hand, the influence of k_L is very small over a relatively large range of values, which seems to suggest that the controlling factor is the butanol diffusion within the particles. The minimum SSE value of 8.765 for the data of Figure 5a is obtained with values of k_L and D_e of 0.002 and 4.22×10^{-9}. The tendency of Figure 4b is highly similar to the one of Figure 4a where the influence of the kinetic rate constant of adsorption k_{ADS} is weak compared to the effect of the diffusion coefficient D_e. In Figure 4b, there exists a clear minimum for the SSE at a specific value of D_e. Figure 5 shows the variation of the SSE when k_L and D_e for Model 4 are changed in the vicinity of the optimal parameters. Model 4, where instantaneous adsorption prevails, has only two important parameters to fit considering that the impact of the axial diffusion coefficient E is negligible. Figure 5 shows a typical concave surface where both parameters influence the SSE even though the effect of the diffusion coefficient is more pronounced. The central location of the red section of Figure 5 is the locus of the minimum SSE.

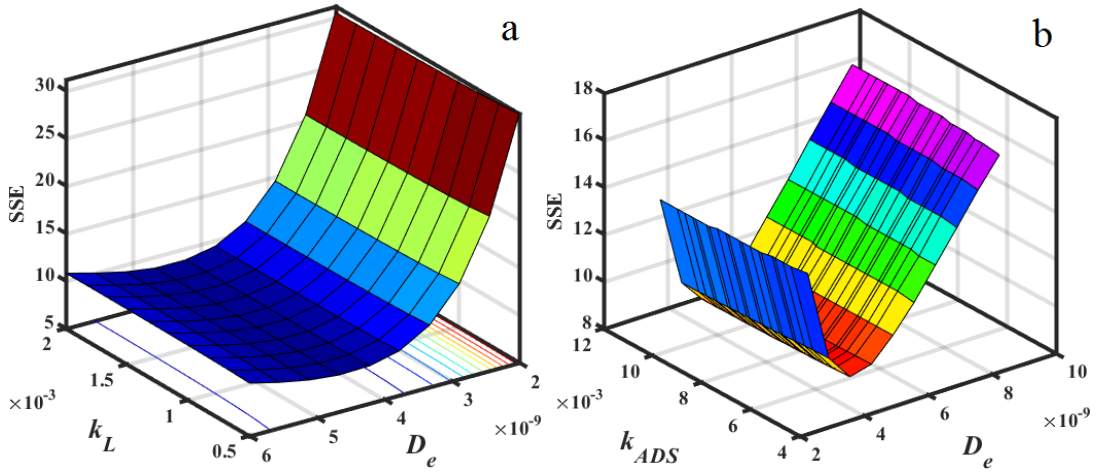

Figure 4 Sum of squares of the errors (SSE) as a function of k_L and D_e (a) and k_{ADS} and D_e (b) for Model 3

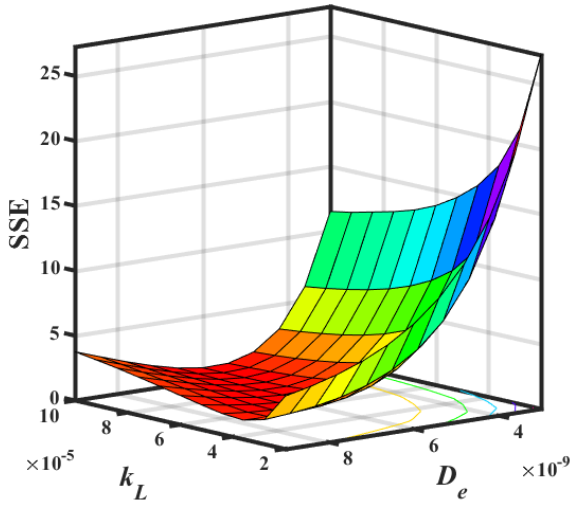

Figure 5 Sum of squares of the errors (SSE) as a function of k_L and D_e for Model 4

Figure 6 presents the variation of the SSE as a function of the diffusion coefficient D_e and the fractional order γ for Model 5. Although both variables have a major impact on the SSE, the fractional order γ has a more important impact especially a low values of γ. It appears that for the diffusion of butanol on AC Norit Row 0.8 is Fickian since the optimal value of γ is unity.

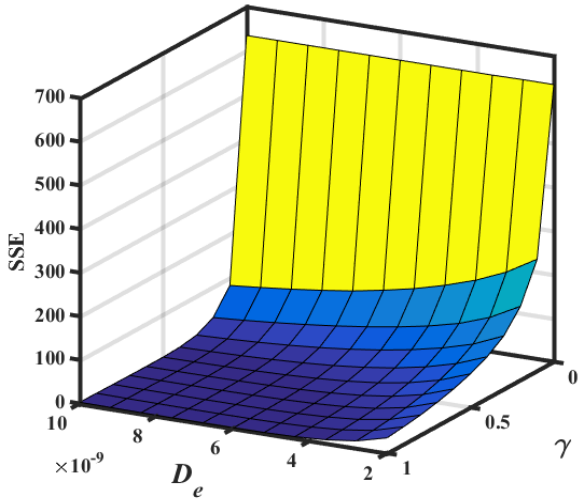

Figure 6 Sum of squares of the errors (SSE) as a function of D_e and γ for Model 5

Based on the model parameters that minimize the SSE for each model, all simulated breakthrough curves represented well the literature experimental data. Figure 7 presents the experimental and simulated breakthrough curves. The simulated breakthrough curves correspond to Models 4 and 5. The butanol concentration values and the trend of the simulated breakthrough curves for Models 4 and 5 are highly consistent with the experimental breakthrough curve. Both models predict nearly identical breakthrough curves. The small discrepancy between the simulated and the experimental breakthrough curves is observed when the adsorption column becomes close to saturation. However, the comparison of Figure 3 for butanol adsorption with AC F-400 with Figure 7 shows that the fitting near the saturation point is much better.

Table 3 gives the model parameters for the five different models. Similarly to results of AC F-400, Models 1 and 2 have the same SSE that are much larger than the SSE of other three models. Again for this system, it can be concluded that Models 1 and 2 with essentially only one significant degree of freedom (k_L) are not suitable for predicting the breakthrough curve of this system. In the results of the adsorption of butanol with AC F-400, Model 3 with the assumption of non-instantaneous adsorption fitted the experimental breakthrough curves quite well. However, for the adsorption of butanol with AC Norit ROW 0.8, Model 3 has a much larger values of the SSE than Models 4 and 5, almost one order of magnitude greater. Thus, these results suggest that the non-instantaneous adsorption does not prevail for this system and models with instantaneous adsorption are more representative for this system. Indeed, when the kinetic constant of adsorption was increased in Model 3, the SSE values decreased. However, it was not possible to use large k_{ads} values because extremely small integration time steps are

required, leading to excessively long computation time. Therefore, it makes sense to assume instantaneous adsorption.

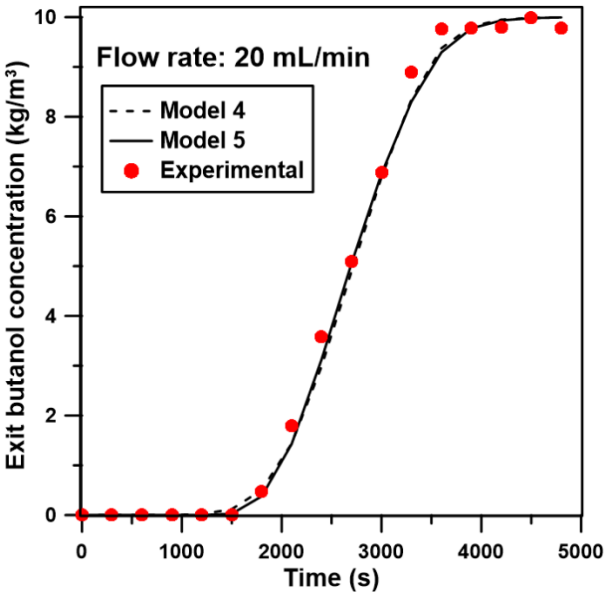

Figure 7 Experimental and simulated breakthrough curves for the adsorption of butanol on AC Norit Row 0.8. Lines and symbols represent the predicted breakthrough curves with Models 4 and 5 and the experimental breakthrough curve, respectively

Models 4 and 5 were derived with the underlying assumption of instantaneous adsorption. Their smaller values of the SSE can be verified by their very good predictions of the breakthrough curves as presented in Figure 7. The main difference between Models 4 and 5 is type of diffusion occurring within the adsorbent particles. For Model 4, the diffusion within the adsorbent particles is assumed to be Fickian diffusion whereas it is assumed potentially anomalous diffusion in Model 5. In Model 5, the derivative order γ for the smallest SSE was obtained at 0.9, which very close to unity. For this system, it can be assumed that the diffusion is essentially Fickian diffusion. The fractional diffusion coefficient for Models 4 and 5 are nearly identical, which further suggest Fickian diffusion. The small decrease in the value of SSE for values of the fractional order of 0.9 compared to the one at a value of γ of unity does not justify the use of Model 5 for this system and Model 4 is recommended.

Table 4 Model parameters determined by the method of least squares to minimize the SSE for experiments with activated carbon Norit Row 0.8

Model	k_L (m/s)	E (m^2/s)	D_e (m^2/s)	k_{ADS} (m/s)	γ (-)	SSE
1	1.63×10^{-5}	1.00×10^{-9}	-	-	-	10.267
2	1.63×10^{-5}	1.00×10^{-9}	-	-	-	10.267
3	3.00×10^{-4}	5.00×10^{-9}	4.60×10^{-9}	8.00×10^{-3}	-	8.963
4	8.40×10^{-5}	1.00×10^{-10}	6.00×10^{-9}	-	-	0.976
5	1.23×10^{-4}	1.00×10^{-10}	5.9×10^{-9}	-	0.9	0.987

4.6 Conclusion

Experimental data for the adsorption of butanol on two types of activated carbon adsorbents, namely F-400 and Norit Row 0.8, were obtained from past experiments performed in our laboratory and from the literature. The equilibrium experimental data were fitted to the Freundlich isotherm model, which was subsequently used within the models for the breakthrough curves. Five models were derived for predicting the temporal and spatial variation of the concentration of butanol in the adsorption packed bed column. It was determined that for the two systems studied, a model needs to consider the diffusion within the particles in order to properly predict the breakthrough curves. For the adsorption of butanol on activated carbon F-400, it is recommended using a fractional order diffusion with instantaneous adsorption. For the adsorption of butanol on activated carbon Norit ROW 0.8, it is recommended using an integer diffusion model with instantaneous adsorption. The gain of using fractional order diffusion equation, given the intensity in computation, may be not sufficient to recommend its use for the latter adsorption system.

4.7 Nomenclature

C_A	Concentration of species A in the adsorbed phase	kg A/kg solid
C_A^*	Hypothetical equilibrium concentration of species A in the adsorbed phase	kg A/kg solid
C_L	Concentration of species A in the mobile phase	kg A/m^3 mobile phase
C_{L0}	Initial concentration	g/L
C_L^*	Hypothetical equilibrium concentration of species A	kg A/m^3 mobile phase
C_P	Concentration of species A in the mobile phase	kg A/m^3 mobile phase
C_P^*	Hypothetical equilibrium concentration of species A in mobile phase	kg A/m^3 mobile phase
D_C	Diameter of packed bed	cm

D_e	Diffusion coefficient	m^2 mobile phase /s
${}_0D_t^{1-\alpha}$	Riemann–Liouville fractional derivative with fractional order 1- α	[time]$^{-1}$
E	Axial diffusion coefficient	m^2 mobile phase /s
F	Flow rate	mL/min
k_L	Mass transfer coefficient	m mobile phase /s
k_La	Volumetric mass transfer coefficient	m^3 mobile phase /s
k_{Ads}	Adsorption kinetic rate constant	m mobile phase /s
K	Parameter of Freundlich model	kg A/kg particles
L_C	Length of packed bed	cm
n	Parameter of Freundlich model	-
r	Position in Particle of adsorbent particles	mm
R	Radius of porous solid adsorbent particles	mm
SSE	Summation of squared errors of curve fitting	-
t	Time	s
u_L	Interstitial velocity	m/s
u_S	superficial velocity	m/s
$w(\tau)$	Possibility density function of particle waiting time between consecutive jumps τ	-
x	Particle jump length x	[length]
$<x^2(t)>$	Mean squared displacement of particle	[length]^2
z	Position in length of packed column	m packed bed
γ	Parameters of fractional order	-
Γ	Gamma function	-
ε_C	Void space of the packed bed	m^3 mobile phase / m^3 packed bed
ε_P	Void fraction of the particles	m^3 mobile phase / m^3 particles
$\lambda(x)$	Possibility density function of particle jump length x	-
ρ_P	Density of an adsorbent particle	kg solid/ m^3 particles
ρ_S	Density of the solid phase	kg solid/ m^3 solid
τ	particle waiting time between consecutive jumps	[time]
$\omega(j)$	Fractional weight	-

4.8 References

[1] G. Morse, R. Jones, J. Thibault, and F. H. Tezel, “Neural network modelling of adsorption isotherms,” *Adsorption*, vol. 17, no. 2, pp. 303–309, Apr. 2011.

[2] M. Petkovska, “Discrimination between adsorption isotherm models based on nonlinear frequency response results,” *Adsorption*, vol. 20, no. 2–3, pp. 385–395, Feb. 2014.

[3] W. Wiechert, “Modeling and simulation: tools for metabolic engineering,” *Journal of Biotechnology*, vol. 94, no. 1, pp. 37–63, Mar. 2002.

[4] M. C. Mauguet, A. Montillet, and J. Comiti, “Macrostructural characterization of granular activated carbon beds,” *J Mater Sci*, vol. 40, no. 3, pp. 747–755, Feb. 2005.

[5] C. Xue *et al.*, “Butanol production in acetone-butanol-ethanol fermentation with in situ product recovery by adsorption,” *Bioresource Technology*, vol. 219, no. Supplement C, pp. 158–168, Nov. 2016.

[6] N. Abdehagh, P. Gurnani, F. H. Tezel, and J. Thibault, “Adsorptive separation and recovery of biobutanol from ABE model solutions,” *Adsorpt.-J. Int. Adsorpt. Soc.*, vol. 21, no. 3, pp. 185–194, Apr. 2015.

[7] I. M. Sokolov, “Models of anomalous diffusion in crowded environments,” *Soft Matter*, vol. 8, no. 35, pp. 9043–9052, 2012.

[8] A. Einstein, *Investigations on the Theory of the Brownian Movement*. Courier Corporation, 1956.

[9] R. Metzler and J. Klafter, “The random walk’s guide to anomalous diffusion: a fractional dynamics approach,” *Physics Reports*, vol. 339, no. 1, pp. 1–77, Dec. 2000.

[10] V. M. Kenkre, E. W. Montroll, and M. F. Shlesinger, “Generalized master equations for continuous-time random walks,” *J Stat Phys*, vol. 9, no. 1, pp. 45–50, Sep. 1973.

[11] B. Berkowitz, H. Scher, and S. E. Silliman, “Anomalous transport in laboratory-scale, heterogeneous porous media,” *Water Resour. Res.*, vol. 36, no. 1, pp. 149–158, Jan. 2000.

[12] S. Havlin, D. Movshovitz, B. Trus, and G. H. Weiss, “Probability densities for the displacement of random walks on percolation clusters,” *J. Phys. A: Math. Gen.*, vol. 18, no. 12, p. L719, 1985.

[13] H. Jobic, A. Méthivier, G. Ehlers, B. Farago, and W. Haeussler, “Accelerated Diffusion of Long-Chain Alkanes between Nanosized Cavities,” *Angewandte Chemie International Edition*, vol. 43, no. 3, pp. 364–366, Jan. 2004.

[14] D. S. Banks and C. Fradin, “Anomalous Diffusion of Proteins Due to Molecular Crowding,” *Biophysical Journal*, vol. 89, no. 5, pp. 2960–2971, Nov. 2005.

[15] H. Jiang, Y. Cheng, L. Yuan, F. An, and K. Jin, “A fractal theory based fractional diffusion model used for the fast desorption process of methane in coal,” *Chaos*, vol. 23, no. 3, p. 033111, Jul. 2013.

[16] G. Baumann and F. Stenger, “Fractional adsorption diffusion,” *Fractional Calculus and Applied Analysis*, vol. 16, no. 3, pp. 737–764, 2013.

[17] C.-M. Chen, F. Liu, I. Turner, and V. Anh, “A Fourier method for the fractional diffusion equation describing sub-diffusion,” *Journal of Computational Physics*, vol. 227, no. 2, pp. 886–897, Dec. 2007.

[18] C. Li and F. Zeng, *Numerical Methods for Fractional Calculus*. CRC Press, 2015.

[19] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*. Elsevier, 1998.

[20] N. Abdehagh, “Improvements in Biobutanol Production: Separation and Recovery by Adsorption,” PhD Thesis, University of Ottawa, 2016.

Appendix

Appendix 1 Ranking of the Pareto domains

For space limitation because the chapters are papers that have been or will be submitted to peer reviewed journals, only some representative figures of the different Pareto domains ranked by Net Flow Algorithm were presented in Chapter 2. However, in this Appendix, all Pareto domains obtained for all four benchmark dynamic systems without or with noise, are presented in Figures 1 to 8. Points colored in yellow were ranked in the top 5% of Pareto domains; red points correspond to solutions ranked in the top 5% to 20%; blue points represent solutions that were ranked in the remaining 80%.

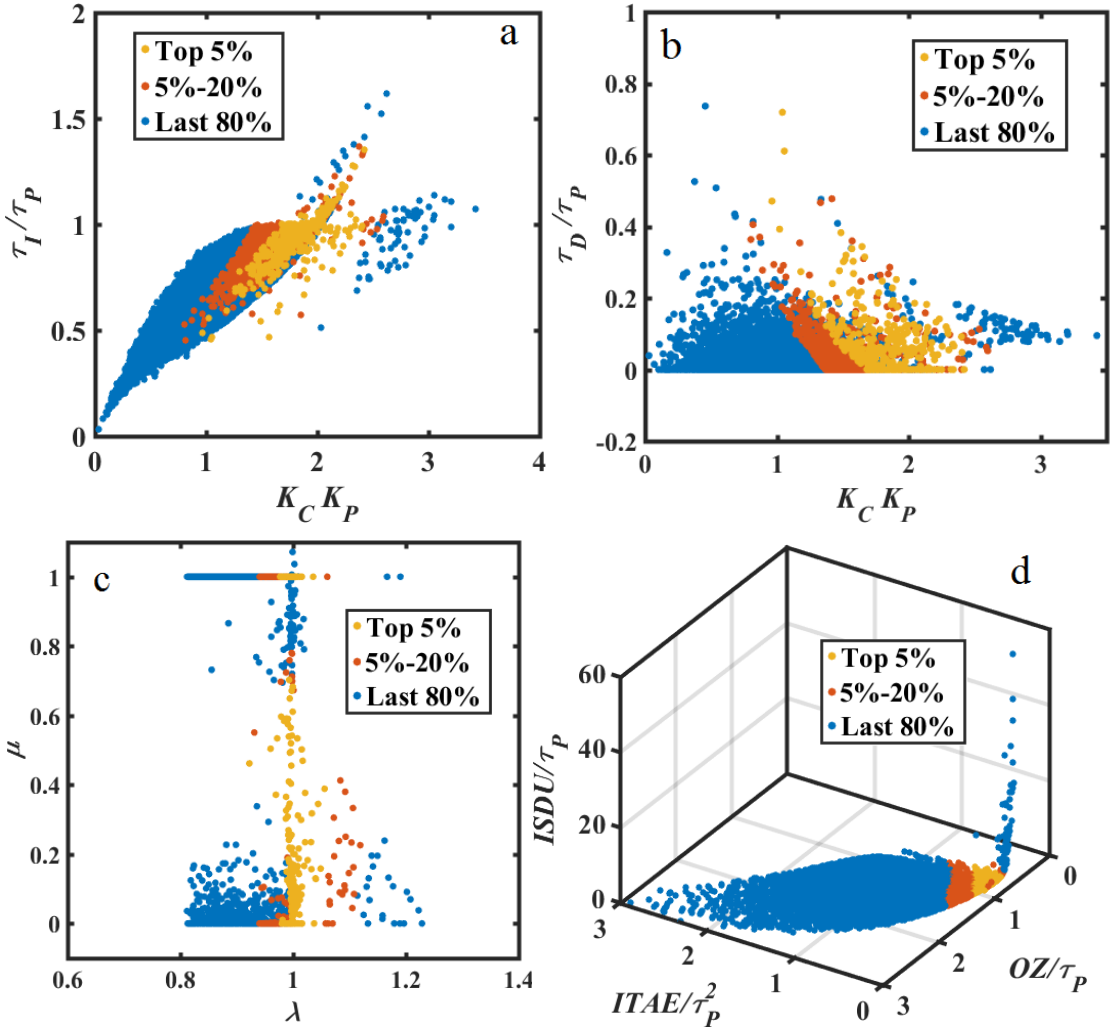

Figure 1 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPDTS without noise.

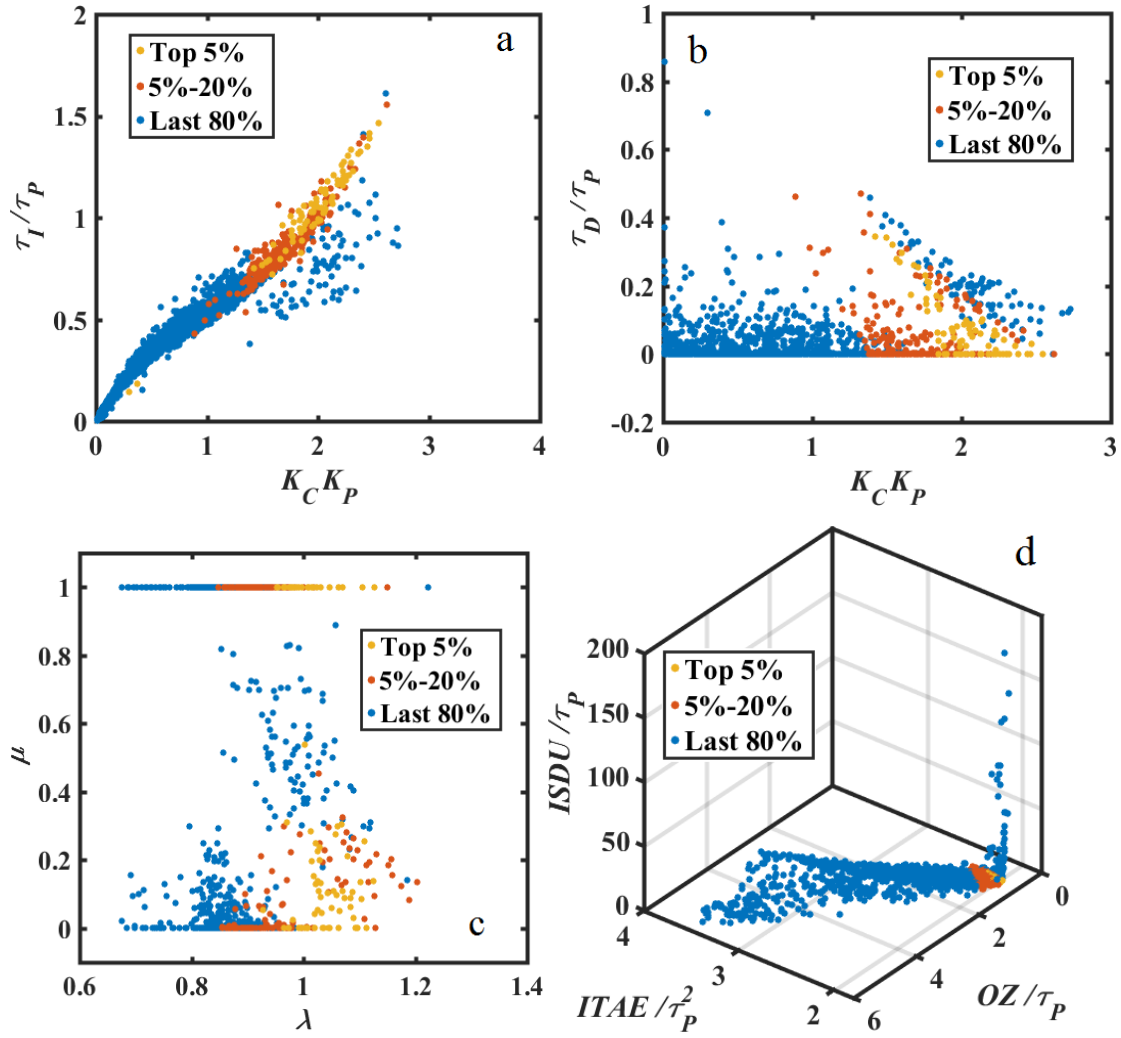

Figure 2 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPDTS with noise.

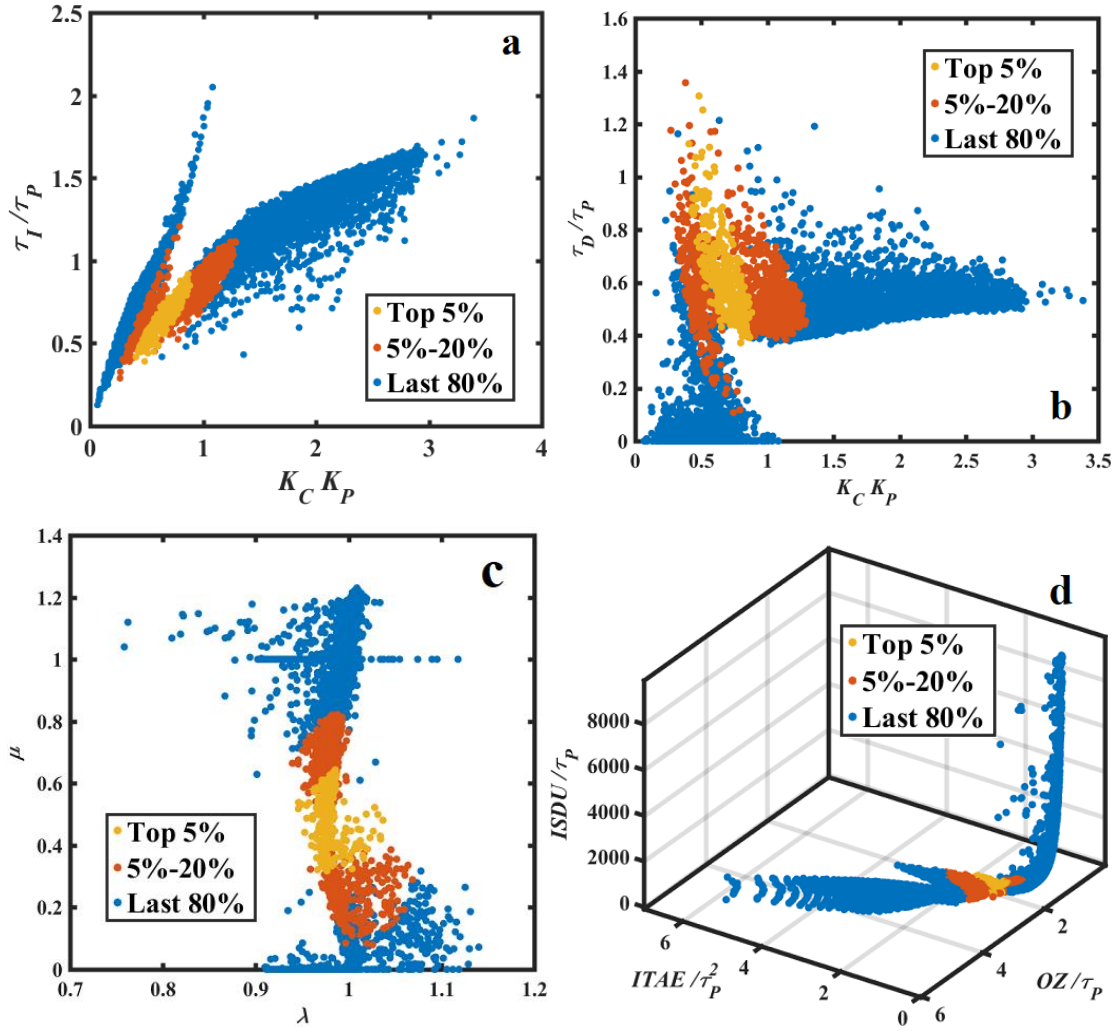

Figure 3 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the HOS without noise.

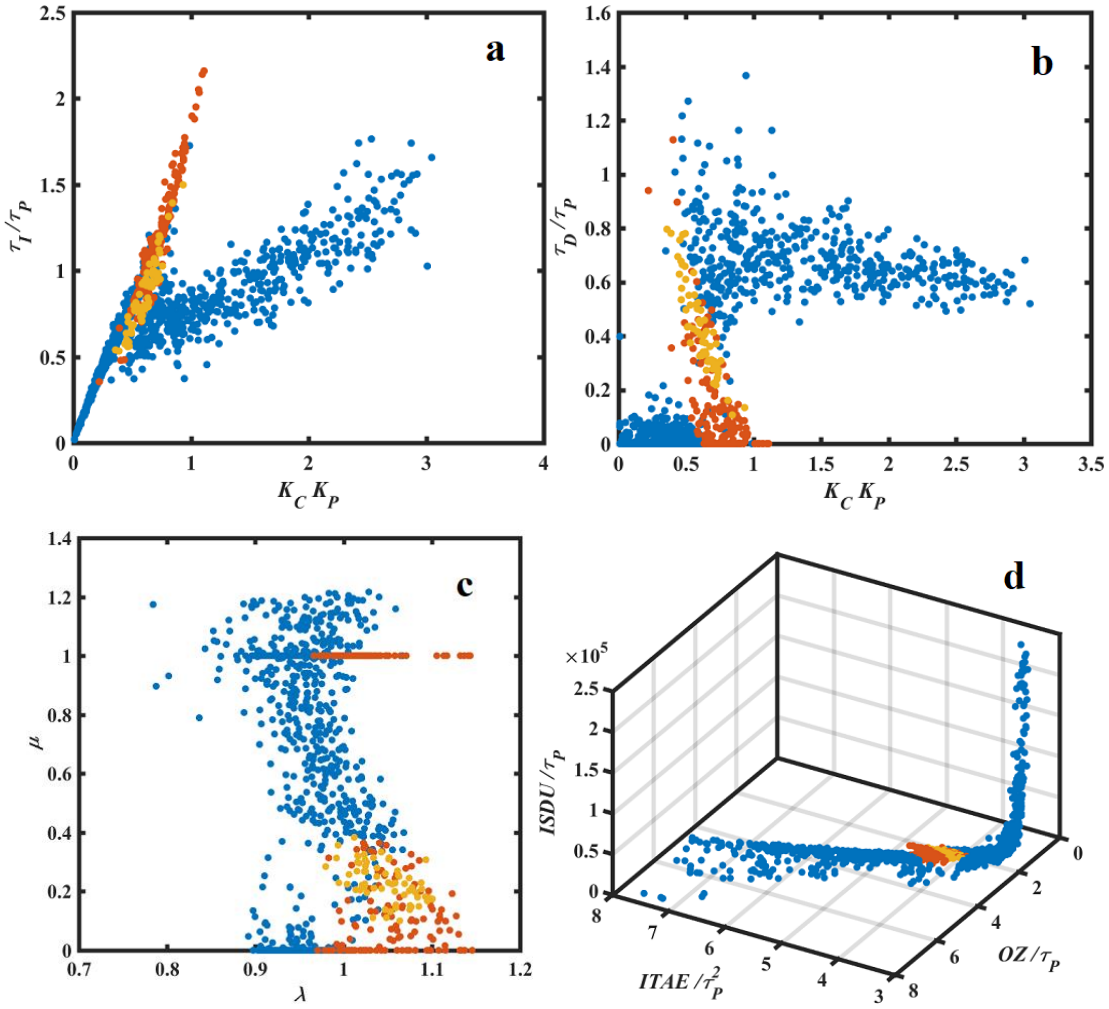

Figure 4 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the HOS with noise/

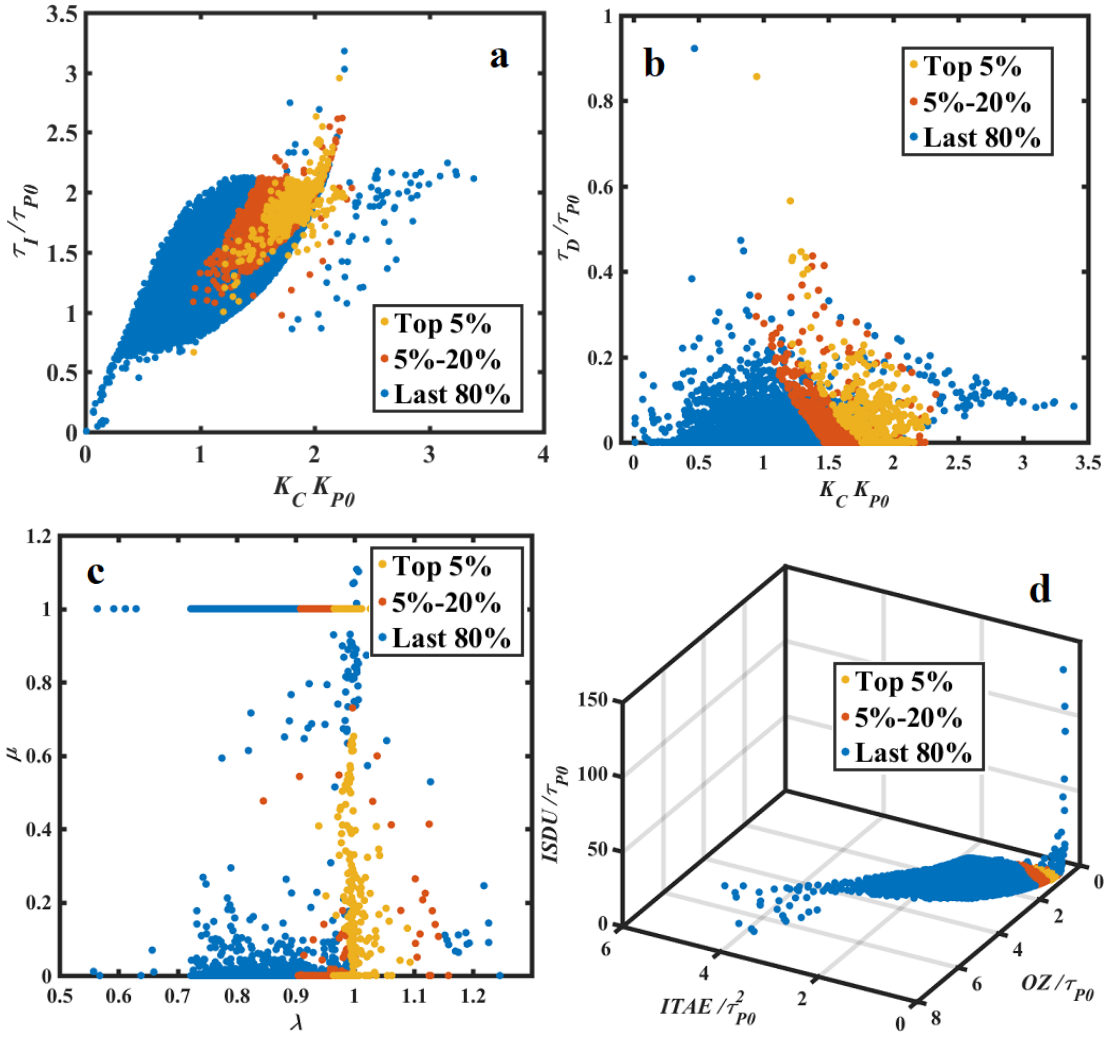

Figure 5 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the NLS without noise.

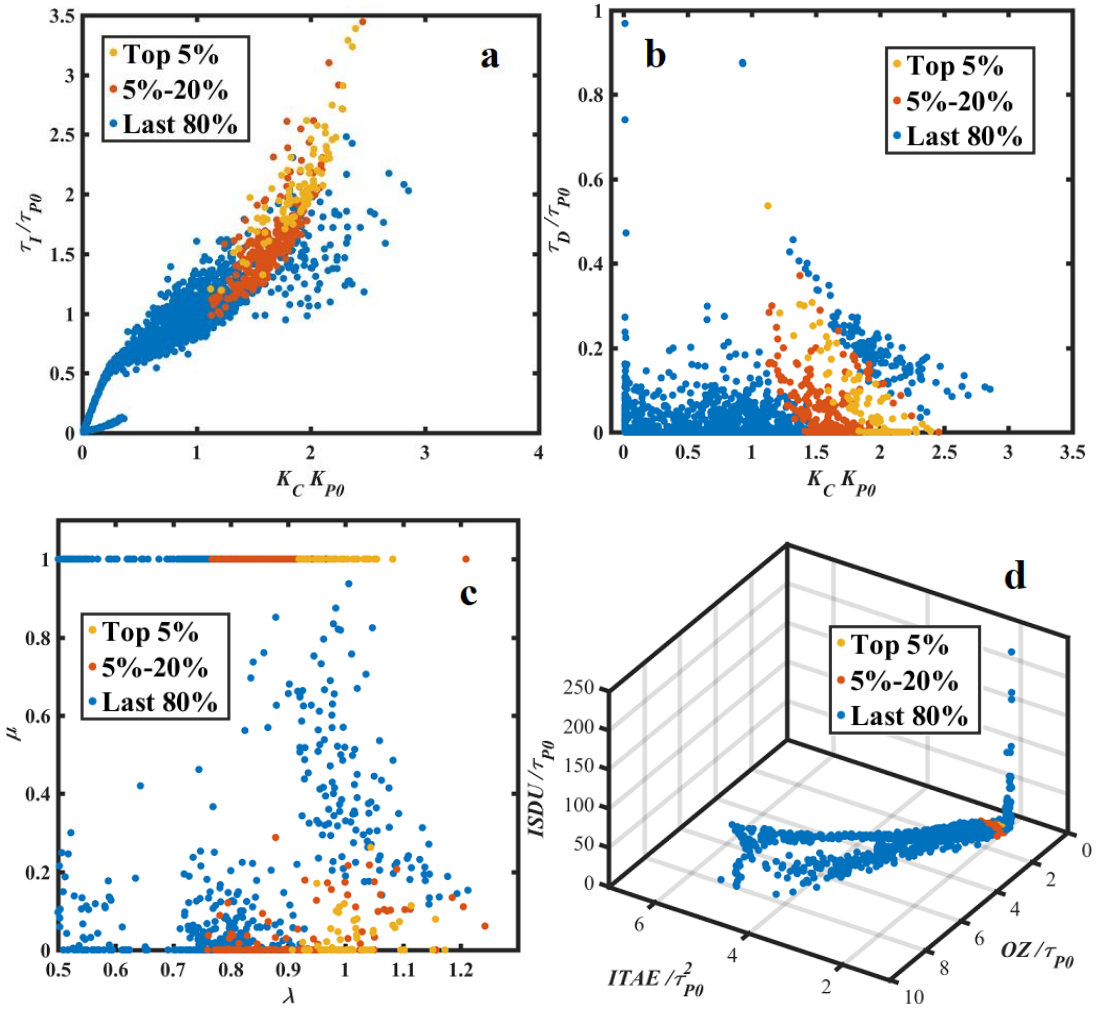

Figure 6 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the NLS with noise.

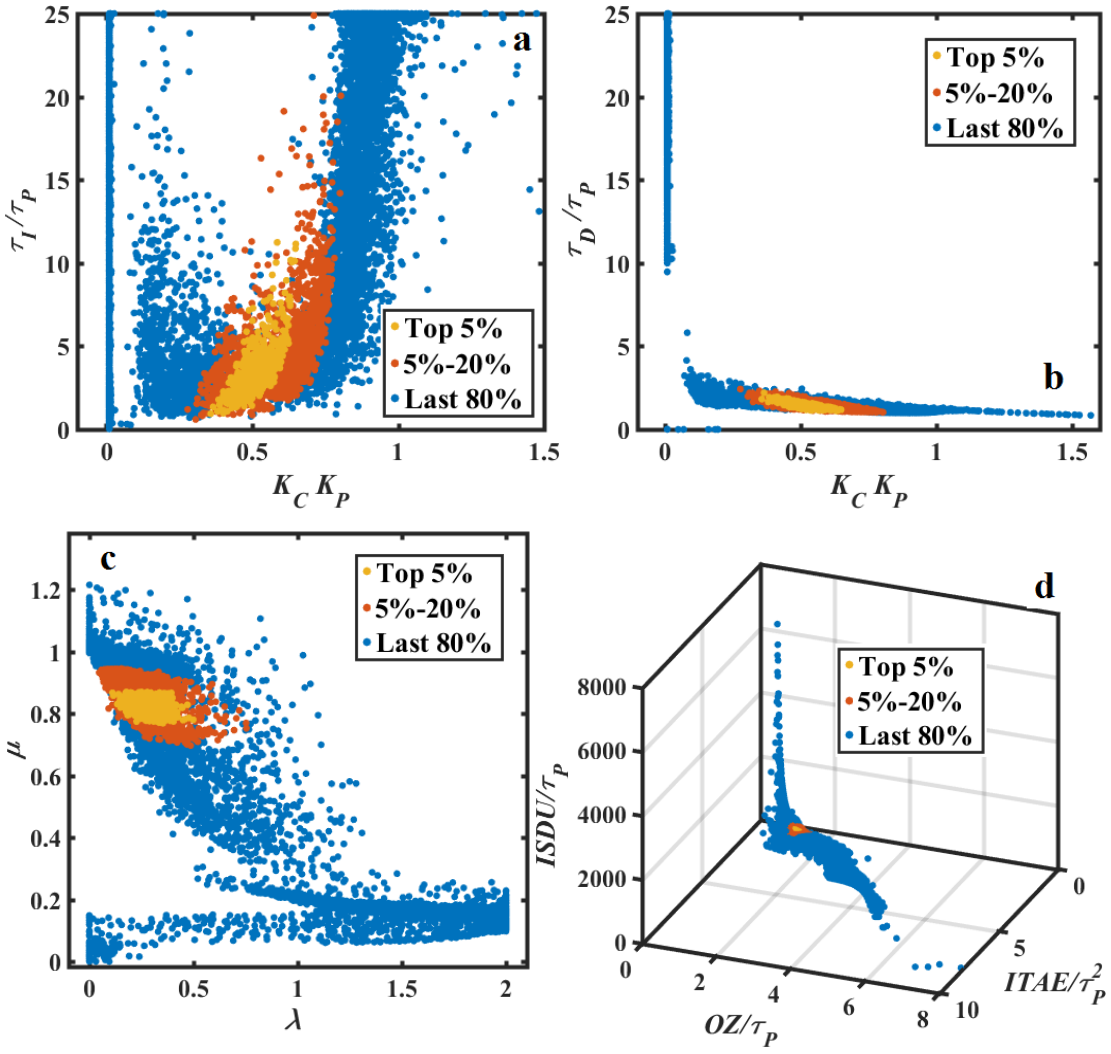

Figure 7 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPIS without noise.

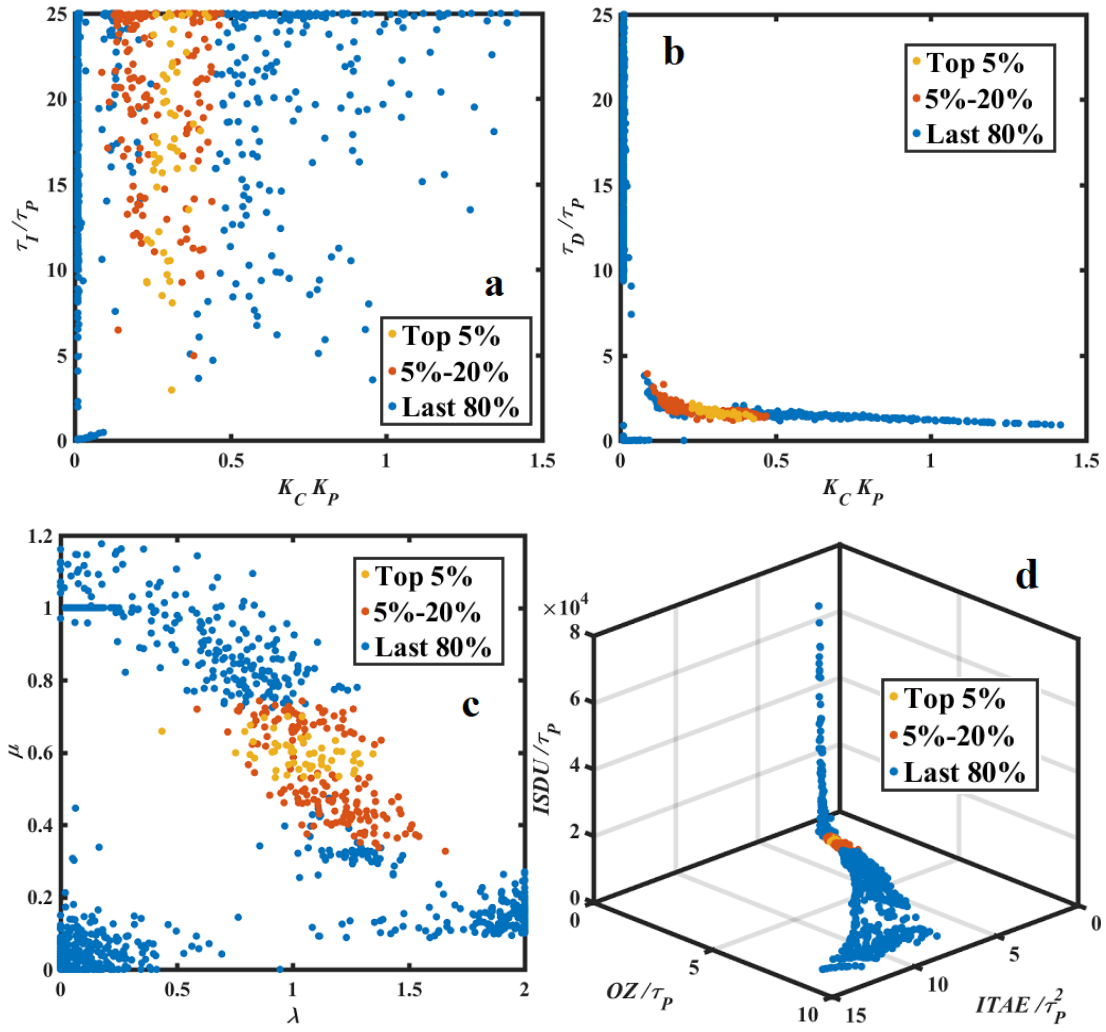

Figure 8 Ranking of the Pareto domain of the fractional $PI^{\lambda}D^{\mu}$ controller using the Net Flow Algorithm for the FOPIS with noise.

Appendix 2 Dynamic Response for the Best Pareto-Optimal Fractional and Linear Controllers

The dynamic response for four different benchmark dynamic systems was obtained by subjecting each system to a unit step change in set point. Figures 9 to 12 show the dynamic response obtained for the best Pareto-optimal fractional and linear controllers ranked by the Net Flow Algorithm, for four the different benchmark systems without or with measuring noise.

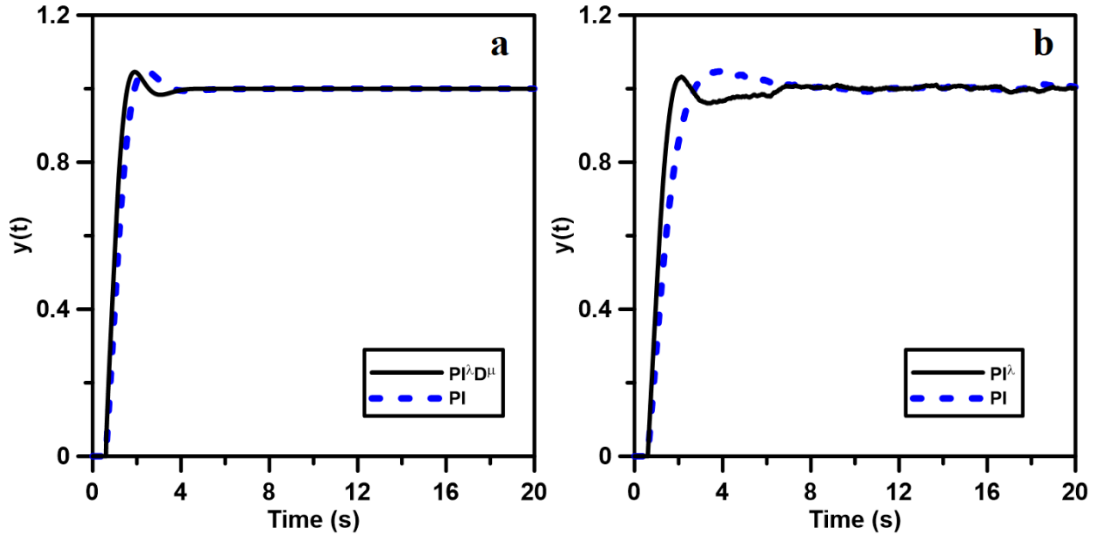

Figure 9 Dynamic response following a unit step change in the set point using the best Pareto-optimal fractional and integer order controllers: (a) FOPDTS without noise; (b) FOPDTS with noise.

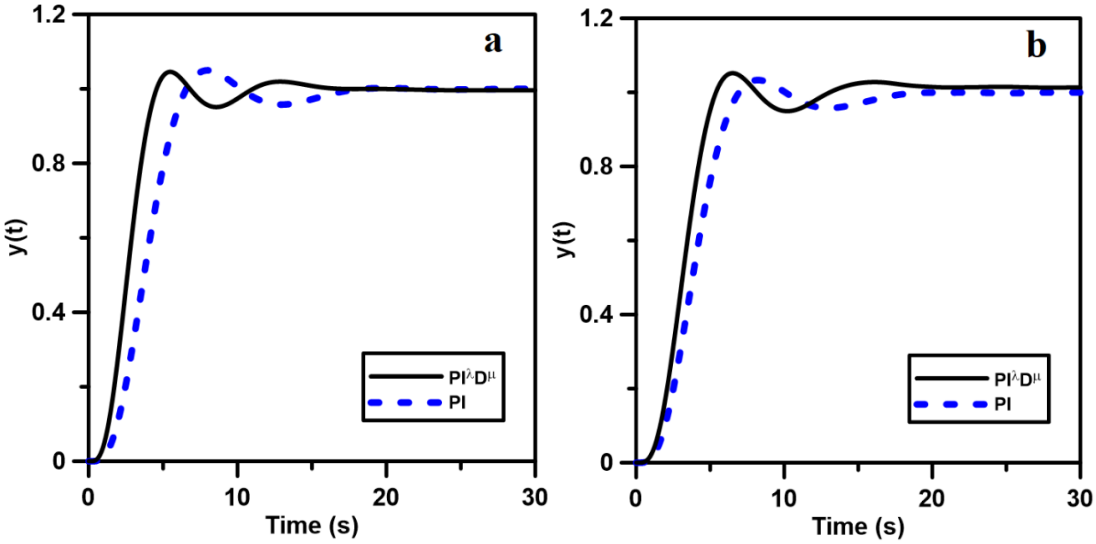

Figure 10 Dynamic response following a unit step change in the set point using the best Pareto-optimal fractional and integer order controllers: (a) HOS without noise; (b) HOS with noise.

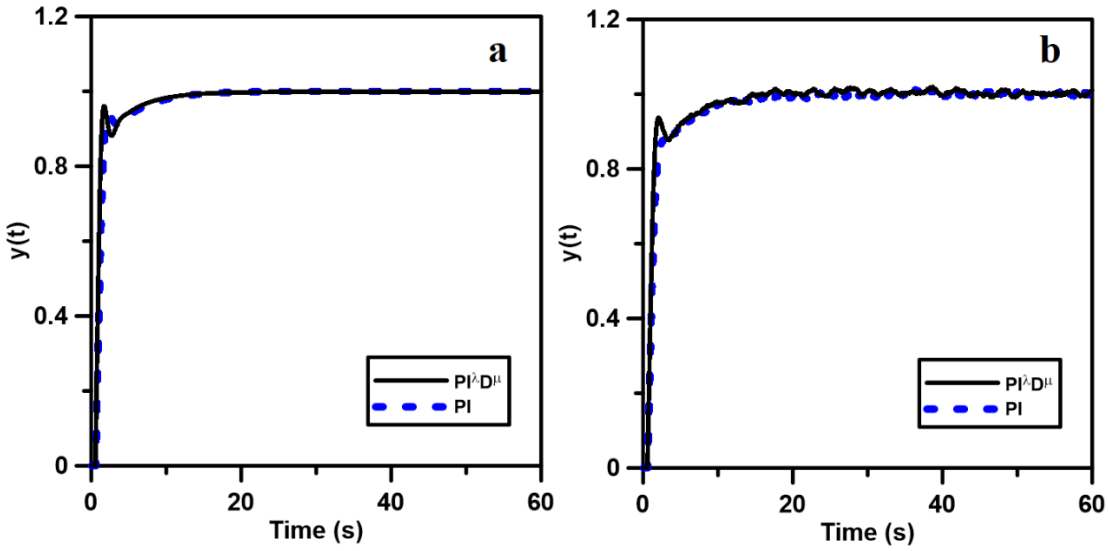

Figure 11 Dynamic response following a unit step change in the set point using the best Pareto-optimal fractional and integer order controllers: (a) NLS without noise; (b) NS with noise.

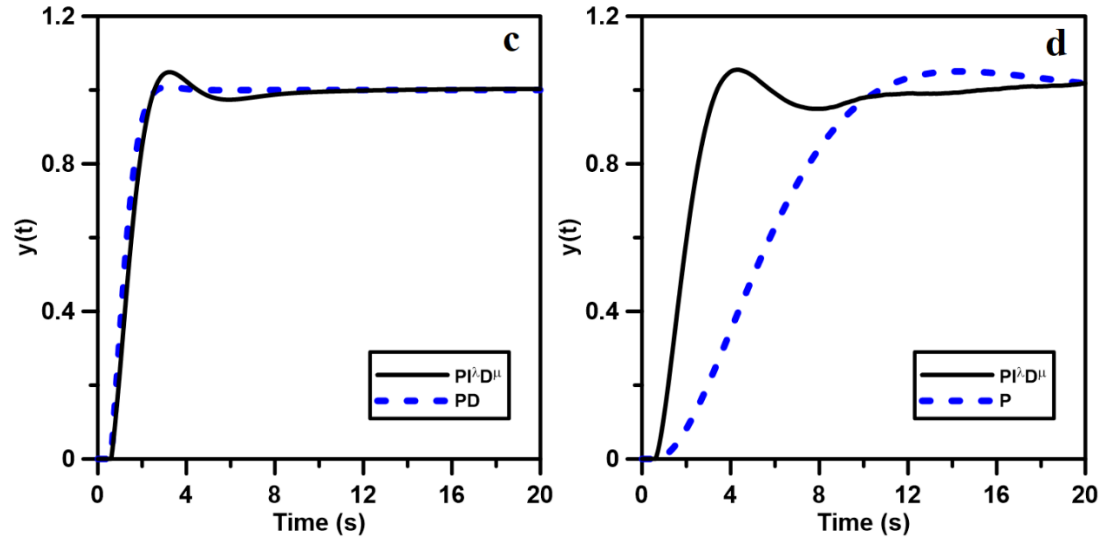

Figure 12 Dynamic response following a unit step change in the set point using the best Pareto-optimal fractional and integer order controllers: (a) FOPIS without noise; (b) FOPIS with noise.

Appendix 3 NSGA-III and Net Flow Algorithms

Pseudo codes for the two main programs used in this investigation, namely the genetic algorithm NSGA-III and the ranking algorithm Net Flow, are presented in this Appendix. NSGA-III includes three subfunctions, which were normalization, association and niching.

Algorithm 1: NSGA-III Pseudocode

Input: Population P_t
Output: P_{t+1}

1: $S_t = \Phi,\ i = 1$
2: $Q_t = Crossover + Mutation\left(P_t\right)$
3: $T_t = P_t \bigcup Q_t$
4: $\left(L_0, ..., L_{last}\right) = \text{Domination Sort}\left(T_t\right)$
5: **Loop**
6: $S_t = S_t \bigcup L_i$
7: $i = i + 1$
8: **Until** $\left|S_t\right| \geq N$
9: **If** $\left|S_t\right| == N$ **then**
10: $P_{t+1} = S_t$
11: **Break**
12: **Else**
13: $P_{t+1} = \bigcup_{j=1}^{last-1} L_j$
14: $a = N - number\ of\ individuals\ in\ P_{t+1}$
15: $Normalized\ S_t = Normalize\left(S_t\right)$
16: $\text{Associate}\left(R, Normalized\ S_t\right)$
17: $NC\left(R\right) = Calculate\ Niching\ Count\ \ for\ \forall R$
18: $P_{t+1} = Niching\left(NC\left(R\right), a, L_{last}\right)$
19: **End if**

Algorithm 2: Normalization Pseudocode

Input: S_t
Output: Normalized S_t

1: **For** $j = 1$ **to** n
2: $z_j^{\min} = \min_{s \in S_t} f_j\left(s\right)$
3: $f_j'\left(s\right) = f_j\left(s\right) - z_j^{\min}$
4: $z_j^{\max} = \max_{s \in S_t} f_j\left(s\right)$
5: **End For**
6: Compute hyperplane intercept for every dimension
7: Compute normalized objectives $f_j^{\,n}$
8: Create reference lines R on hyperplane

Algorithm 3: Association Pseudocode

Input: R, S_t

Output: $R(\boldsymbol{s})$, $d(\boldsymbol{s})$

1: **For each** $\boldsymbol{s} \in S_t$

2: **For each** $w \in R$

3: $d(\boldsymbol{s}, \boldsymbol{w}) = \left\| \left(\boldsymbol{s} - \boldsymbol{w}^T \boldsymbol{s} \boldsymbol{w} \right) / \left\| \boldsymbol{w} \right\|^2 \right\|$

4: **End For**

5: $R(\boldsymbol{s}) = \boldsymbol{w} : \ \text{argmin}_{\boldsymbol{w} \in Z_r} d^{\perp}(\boldsymbol{s}, \boldsymbol{w})$

6: $d(\boldsymbol{s}) = d^{\perp}(\boldsymbol{s}, R(\boldsymbol{s}))$

7: **End For**

Algorithm 4: Niching Pseudocode

Input: $a, NC, d, R, L_{last}, P_{t+1}$

Output: P_{t+1}

1: $k = 1$

2: **While** $k \le a$

3: $J_{\min} = \left\{ j : \text{argmin}_{j \in R} NC(R) \right\}$

4: $\bar{j} = random(J_{\min})$

5: $I_{\bar{j}} = \left\{ \boldsymbol{s} : R(\boldsymbol{s}) = \bar{j}, \ \boldsymbol{s} \in L_{last} \right\}$

6: **If** $I_{\bar{j}} \neq \Phi$ **then**

7: **If** $NC(j) = 0$ **then**

8: $P_{t+1} = P_{t+1} \bigcup \left(\boldsymbol{s} : \text{argmin}_{\boldsymbol{s} \in I_{\bar{j}}} d(\boldsymbol{s}) \right)$

9: **Else**

10: $P_{t+1} = P_{t+1} \bigcup random(I_{\bar{j}})$

11: **End if**

12: $NC(j) = NC(j) + 1$

13: $L_{last} = L_{last} - \boldsymbol{s}$

14: $k = k + 1$

15: **Else**

16: $R = R - \bar{j}$

17: **End if**

18: **End while**

Algorithm 5: Net Flow Algorithm Pseudocode

Input: number of solutions in Pareto domain *s*, and *W*, *Q*, *P*, *V* for each objective

Output: Ranked Pareto domain with scores

1: **For** each solution *i* in Pareto domain

2: $\sigma[i, j] = 0$, $\sigma[j, i] = 0$

3: **For** each solution *j* in Pareto domain

4: $C[i, j] = 0$, $temp_1 = 0$

5: **For** each objective *k*

6: Calculate $\Delta_k[i, j]$, $c_k[i, j]$, $D_k[i, j]$

7: $C[i, j] = C[i, j] + W_k c_k[i, j]$

8: $temp_1 = temp_1 \bullet \left[1 - \left(D_k[i, j]\right)^3\right]$

9: **End**

10: $\sigma[i, j] = \sigma[i, j] + temp_1 \bullet C[i, j]$

11: $C[j, i] = 0$, $temp_2 = 0$

12: **For** each objective *k*

13: Calculate $\Delta_k[j, i]$, $c_k[j, i]$, $D_k[j, i]$

14: $C[j, i] = C[j, i] + W_k c_k[j, i]$

15: $temp_2 = temp_2 \bullet \left[1 - \left(D_k[j, i]\right)^3\right]$

16: **End**

17: $\sigma[j, i] = \sigma[j, i] + temp_2 \bullet C[j, i]$

18: **End**

19: $\sigma_i = \sigma[i, j] - \sigma[j, i]$

20: **End**