Université d'Ottawa · University of Ottawa

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

HALSALL-WHITNEY, Hayley
AUTEUR DE LA THÈSE - AUTHOR OF THESIS

M.A.Sc. (Chemical Engineering)
GRADE - DEGREE

Department of Chemical Engineering
FACULTÉ, ÉCOLE, DÉPARTEMENT - FACULTY, SCHOOL, DEPARTMENT

TITRE DE LA THÈSE - TITLE OF THE THESIS

The Development and Application of a Multi-Objective Optimization Technique for Chemical Processes and Controller Design

Jules Thibault / David Taylor
DIRECTEUR DE LA THÈSE - THESIS SUPERVISOR

EXAMINATEURS DE LA THÈSE - THESIS EXAMINERS

M.A. Dubé

J. Zhang

J.-M. De Koninck, Ph.D.

LE DOYEN DE LA FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

SIGNATURE

DEAN OF THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

The Development and Application of a Multi-Objective Optimization Technique for Chemical Processes and Controller Design

The Development and Application of a Multi-Objective Optimization Technique for Chemical Processes and Controller Design

by

Hayley Halsall-Whitney

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of

Masters of Applied Science

in

Chemical Engineering

Department of Chemical Engineering
Faculty of Engineering
University of Ottawa

March, 2004

Your file Votre référence
ISBN: 0-494-01484-9
Our file Notre référence
ISBN: 0-494-01484-9

Acknowledgments

I would like to thank my supervisors **Dr. Jules Thibault** and **Dr. David Taylor** for their support. I would especially like to thank Dr. Jules Thibault for his policy of always having time for his students, for his constant encouragement and patience throughout my MASc. in Chemical Engineering studies and research at the University of Ottawa, and for introducing me to the exciting and rapidly growing field of Multi-objective Optimization, one of his research interests.

I would also like to thank my grandmother, **Adeline Grant-Halsall**, my parents**, Douglas and Gloria Halsall**, and my husband, **Christopher Whitney,** for their complete and unconditional support and continuous encouragement.

Statement of Contributions of Collaborators

I hereby declare that I am the sole author of this thesis, and the sole software developer for the Java program used for conducting the simulations in this work. No part of this work has been submitted or accepted for any other degree.

Multi-objective optimization, which is the focus of this thesis, is the research interest of **Dr. Jules Thibault**. Primarily Dr. Jules Thibault, along with **Dr. David Taylor**, supervised this thesis. Dr. Jules Thibault provided continual guidance throughout this work and made editorial comments and corrections to the written work presented. The responsibilities of the author, MASc. student **Ms Hayley Halsall-Whitney**, in order to fulfill the requirements of this thesis were as follows.

1. To conduct research in the area of multi-objective optimization in order to determine the current state of the art.
2. To design, develop, and test a software program, using object oriented programming principles, in order to study the application of multi-objective optimization techniques to chemical processes and controller design.
3. To produce three papers for publication based on the research.
4. To produce a written thesis in partial fulfillment of the requirements for obtaining a Masters in Applied Science.

Although a number of undergraduate students have completed their undergraduate theses using some of the techniques included in this thesis, on no occasion has any undergraduate student or graduate student contributed to or was involved in any part of this research or in the writing of the papers included in this thesis or in the writing of this thesis. Under no occasion was their any collaboration with any person, association, or organization beyond the thesis supervisors.

Signature: ______________________________

Date: ______________________________

Abstract

In recent years, the development and application of multi-objective optimization techniques have received increasing attention in the literature. Recent innovations in the fields of systems science, artificial intelligence, and operations research have led to the development of rigorous multi-objective optimization techniques to address the problem of optimizing complex processes in the presence of multiple conflicting objectives.

This thesis is a collection of three papers that focuses on the development and application of a multi-objective optimization strategy for selecting optimum operating conditions for the production of gluconic acid and for determining optimum tuning parameters for PID controllers. The optimization strategy developed is performed in two steps: the approximation of the set of feasible solutions called the Pareto domain using the Dual Population Evolutionary Algorithm and the classification of the domain using the Net Flow Method, which incorporates information on the process provided by an expert. This strategy has been proven to be robust in determining the optimal solution after studying twelve standard test cases, which have been used frequently in the literature, and two engineering problems. In addition, the Pareto domain per se provides very useful information on the quality of the optimal zone.

Résumé

Au cours des dernières années, le développement et l'application des techniques d'optimisation à objectifs multiples ont suscité davantage d'intérêt dans la littérature. Les innovations récentes dans les domaines de la science des systèmes, de l'intelligence artificielle et de la recherche opérationnelle ont mené au développement de rigoureuses techniques d'optimisation à objectifs multiples pour adresser le problème d'optimisation des procédés complexes en présence d'objectifs contradictoires multiples.

Cette thèse consiste principalement en une collection de trois articles qui se concentrent sur le développement et l'application d'une stratégie d'optimisation à objectifs multiples dans le but de déterminer des conditions d'opération optimales pour la production d'acide gluconique et des paramètres d'ajustement optimaux pour des contrôleurs PID. Cette stratégie d'optimisation est effectuée en deux étapes: l'approximation du domaine de Pareto par un grand nombre de solutions cohérentes et la classification de ces solutions en utilisant la méthode de bilan des flux (Net Flow Method), qui incorpore des informations sur le procédé fournies par un expert. Il a été démontré à travers de nombreux exemples que cette stratégie est très robuste pour l'obtention d'une solution qui rencontre les critères souhaités de l'expert du domaine. De plus, la zone de Pareto fournit des informations intéressantes sur la qualité de la région optimale du procédé.

Table of Contents

List of Figures

Paper 1

Page

Paper 2

Paper 3

List of Tables

Paper 1

Page

Paper 2

Paper 3

Chapter 1

Introduction

1.1 Multi-objective Optimization

In recent years, the development and application of multi-objective optimization techniques to chemical engineering problems (Silva and Biscaia, 2003; Costa *et al.*, 2003; Gergely *et al.*, 2003; Yee *et al.*, 2003; Bhaskar *et al.*, 2001; Rajesh *et al.*, 2001) and robust controller design (Liu *et al.*, 2003; El-Kady *et al.*, 2003; Herreros *et al.*, 2002; Kang and Bien, 2000; Hu *et al.*, 1998; Rangan and Poolla, 1997; Fonseca and Fleming, 1995; Porter and Jones, 1992, Flemming, 1985) have received wide attention in the literature, with varying degrees of success. In complex chemical processes, the ability to select optimum operating conditions in the presence of multiple conflicting objectives, given the various constraints, can present a major challenge. For this reason, multi-objective optimization has been applied to many chemical optimization problems, such as, styrene polymerization (Costa *et al.*, 2003), batch free radical polymerization (Silva and Biscaia, 2003), wine filtration (Gergely *et al.*, 2003), styrene production (Yee *et al.*, 2003), adsorption processes (Ko and Moon, 2002), and reactor design (Bhaskar et al., 2001), fermentation (Wang and Sheu, 2000), and waste minimization (Dantus and High, 1999) to name a few. In modern control theory determining optimum tuning parameters for a controller in order to achieve various design objectives is a multi-objective problem with often conflicting criteria. There have been many applications of multi-objective optimization techniques to controller design, such as, model parameter optimization for controller (Vlachos *et al.*, 1999; Onnen *et al*, 1997; Wang and Kwok, 1992; Porter and Jones, 1992; Oliveira *et al.*, 1991) and plant models (Koza *et al*, 2000, Choi *et al.*, 2000, Billings and Mao, 1998), fuzzy and neural control (Linkens and Nyongesa, 1996; Ichikawa and Sawa, 1992; Angeline *et al.*, 1994; Tzes *et al.*, 1998;Kim *et al.*, 1995), fault diagnosis (Patton *et al.*, 1997; Marcu *et al.*, 1998; Chen *et al.*, 1995; Miller *et al.*, 1993), and closed-loop system performance analysis (Schubert and Stengel, 1998; Marrison and Stengel, 1997), to name a few. In both cases the application of a single-objective optimization technique does not always provide acceptable results.

Traditional optimization techniques have dealt with multiple objectives by combining them into a single objective function composed of the weighted sum of the individual objectives, or by focusing on one objective while transforming the others into constraints (Edgar *et al.*, 2001). These techniques are based on the assumption that the objective functions are well behaved, in the sense that they are either concave shaped or convex shaped and continuous, and that there exists an optimal point that will resolve the issue of conflicting objectives. Although single objective algorithms are sufficient in many cases for optimization, there are many drawbacks to their use, especially when the objectives under consideration are conflicting. Some of these issues are listed as follows.

1. Combining multiple objectives into a single objective function does not provide the decision-maker with information about trade-offs amongst the various objectives, or about alternative operating conditions.

2. Transforming a multi-objective problem into a single objective function composed of the weighted sum of the objectives relies only, and sometimes heavily, on the selection of weights, which is not a trivial task. The application of different weights leads to a variety of solutions.

3. An optimization technique that constrains some of the objectives may bias the final solution, which can result in misleading conclusions. The decision in regards to which objective function to optimize may not be a simple one.

4. Single objective optimization techniques provide only one optimal operating solution even if there are other possible solutions. These techniques are often plagued with the problem of finding global optima or multiple global optima, and could miss possible solutions if the functions are non-convex, multi-modal or discontinuous. They also require information about function derivatives and an initial estimate of the solution, which may not be readily available.

5. Techniques that provide only one optimal operating solution instead of an optimal operating region do no take into account the difficulty associated with controlling the process.

6. Traditional optimization techniques do not incorporate the practical experience and knowledge of the decision-maker in regard to the overall behavior of the process.

In reality, many chemical processes and control design problems are defined by complex equations where the application of single-objective optimization techniques do not provide satisfactory results in the presence of multiple conflicting objectives or in situations where the solution space is non-convex, non-concave, multi-modal or discontinuous. Instead, the solution lies with the application of multi-objective optimization techniques.

Multi-objective optimization refers to the simultaneous optimization of multiple conflicting objectives, which produces a set of alternative solutions called the Pareto domain (Deb, 2001). These solutions are optimal in the sense that no one solution is better than any other in the domain when compared on all criteria. The decision-maker's experience and knowledge is then incorporated into the optimization procedure in order to classify the available alternatives in terms of his or her preference (Doumpos and Zopounidis, 2002). Some of the advantages of using multi-objective optimization techniques are as follows.

1. They can simultaneously optimize multiple and conflicting objectives.

2. These techniques have a global perspective and are not affected by multiple global or local optima, and do not require information about function derivatives.

3. They have the ability to generate multiple solutions in a given iteration that cover the entire search space.

4. They can be applied to functions that are non-convex, non-concave and discontinuous.

In recent years, new methodologies for generating and classifying the Pareto domain have been developed. In terms of generating the Pareto domain, a general class of multi-objective optimization techniques called Evolutionary Algorithms has gained increasing attention in the literature and has been successfully applied to many engineering problems (Silva and Biscaia, 2003; Liu *et al.*, 2003; Shim *et al.*, 2002; Deb, 2001; Coello, 1999; Fonseca *et al*, 1995; Viennet *et al.*, 1996). However, there has been no one acceptable technique and current methodologies can vary a lot in terms of the solutions. For this reason, many standard test cases with varying degrees of difficulty have been developed to allow researchers to compare their techniques to others reported in the literature (Silva and Biscaa, 2003; Deb, 2001; Poloni *et al.*, 2000; Viennet, 1996; Kursawe, 1990). In terms of classifying the Pareto domain, many techniques have been developed that incorporate the experience and knowledge of the decision-maker into the optimization process. This thesis concentrates on a class of classification techniques called ELECTRE, which has its origins in the field of economics and finance (Doumpos and Zopounidis, 2002; Scarelli and Narula, 2002; Triantaphyllou, 2000; Evangelos, 2000; Derot *et al.*, 1997; Roy, 1991; Brans *et al.*, 1984; Roy, 1978) and has also been successfully applied to the field of chemical engineering (Halsall-Whitney *et al*, 2003; Thibault *et al.*, 2001; Perrin *et al.*, 1997; Couroux *et al.*, 1995; Viennet *et al.*, 1995).

1.2 Scope of Current Research

This research focuses on the development and application of a multi-objective optimization strategy for selecting optimum operating conditions for the production of gluconic acid and for determining optimum tuning parameters for the PID family of controllers. This multi-objective optimization strategy uses an evolutionary algorithm called the Dual Population Evolutionary Algorithm to generate the Pareto domain and a classification algorithm called Net Flow. The results of this work is presented here as a collection of papers which are listed and summarized as follows.

1. The first paper, Multi-objective Optimization for Chemical Processes and Controller Design: Approximating and Classifying the Pareto Domain, proposes a multi-objective optimization strategy. This work examines the robustness of three techniques for generating a set of Pareto optimal solutions and demonstrates how

these solutions can be classified using Net Flow, an algorithm that incorporates the use of 'Expert Knowledge'.

2. The second paper, Multicriteria Optimization of Gluconic Acid Production Using Net Flow, applies a multi-objective evolutionary algorithm and Net Flow to determine optimum operating conditions for the production of gluconic acid. This paper has been published in the Journal of Bioprocess and Biosystems Engineering (Halsall-Whitney *et al.*, 2003). The evolutionary algorithm used has since been referred to as the Dual Population Evolutionary Algorithm, which was developed during the course of this research.

3. The third paper, Multi-objective PID Controller Design for First Order Processes with Deadtime, uses the Dual Population Evolutionary Algorithm, a multi-objective evolutionary algorithm, and Net Flow, a classification algorithm, to determine optimum tuning parameters for the PID family of controllers and compares the results to the Rovira tuning relations for minimizing the ITAE (Rovira, 1981) and the Ciancone correlations (Marlin, 2000).

This collection of papers is self-contained. Therefore, additional information beyond the scope of each paper is not required in order to understand the research. Supporting information for Papers 1-3 is located in Appendices A-E. Additional information, including published papers, graduate seminar and conference presentations, and a copy of this thesis, is contained on the compact disk provided.

The simulations in this research were performed using software called *Multi-objective Optimization* that was designed, developed and tested by the author using the Java programming language. The software and the Pareto domains produced by the simulations are contained on the companion compact disk in Appendix C and Appendix D, respectively.

References

Angeline, P. J., Saunders, G. M., and Pollack, J. B., An Evolutionary Algorithm That Constructs Recurrent Neural Networks, *IEEE Transactions on Neural Networks*, Vol. 5, No.1, pgs 54-65, 1994

Bhaskar, V., Gupta, S. K., and Ray, A. K., Multi-objective Optimization of Industrial Wiped-film Poly (Ethylene Terephthalate) Reactor: Some Further Insights, *Computers and Chemical Engineering*, Vol. 25, pgs 391, 2001

Billings, S. A., and Mao, K. Z., Structure Detection for Non-linear Rational Models Using Genetic Algorithms, *Internaltional Journal of Systems Science*, Vol. 144, No.1, pgs. 223-231, 1998

Brans, J. P., Mareschal, B., and Vincke, P. H., PROMETHEE: A New Family of Outranking Methods in Multicriteria Decision Making, *Operations Research*, Vol. 84, pgs. 477-498, 1984

Chen, B., Cheng, Y., and Lee, C., A Genetic Approach to Mixed H_2/H_∞ Optimal PID Control, *Control Systems*, Vol. 15, No. 5, pgs. 51-60, 1995

Choi, S. H., Lee, C. O., and Cho, H. S., Friction Compensation Control of an Electopnewmatic Servovalve by using an Evolutionary Algorithm, *Proceedings of the Institution of Mechanical Engineers*, Vol. 214, Part I, pgs. 173-184, 2000

Coello, C. A., A Comprehensive Survey of Evolutionary-Based Multi-objective Optimization Techniques, *Knowledge and Information Systems*, Vol. 1, pgs. 269-308, 1999

Costa, E. F., Lage, P. L. C., and Biscaia, E. C., On The Numerical Solution and Optimization of Styrene polymerization in Tubular Reactors, *Computers and Chemical Engineering*, Vol. 27, Issue 11, pgs. 1591-1604, 2003

Couroux, P., Qannar, E. M., Melcion, J. P., and Morat, J., Optimisation Multiréponse: Application a un Procédé de Granulation d'Aliments, *Récents Progrès en Génie des Procédés: Strtégie Expérimentale et Procédés Biotechnologiques*, Vol. 36, No. 9, pgs. 41-47, 1995;

Dantus, M. M., and High, K. A., Evaluation of Waste Minimization Alternatives Under Uncertainty: A Multiobjective Optimization Approach, *Computers and Chemical Engineering*, Vol. 23, Issue 10, pgs. 1493-1508, 1999

Deb, K., Multi-objective Optimization Using Evolutionary Algorithms, Wiley, NY, 2001

Derot, B., Gareau, J., Kiss, L. N., and Martel, J. M., The Solver of Volvox Multicriteria Table. Multicriteria Analysis, Climaio, J., (ed) Springer-Verlag, pgs. 113-126, 1997

Doumpos, M., and Zopounidis, C., Multicriteria Decision Aid Classification Methods, *Applied Optimization*, Vol. 73, Kluwer Academic, 2002.

Edgar, T. F., Himmelblau, D. M., and Lasdon, L. S., Optimization of Chemical Processes. 2nd ed., McGraw-Hill, 2001

El-Kady, M. M., Salim, M. S., and El-Sagheer, A. M., Numerical Treatment of Multiobjective Optimal Control Problems, *Automatica*, Vol. 39, Issue 1, pgs. 47-55, 2003

Evangelos, T., Multi-criteria Decision Making Methods: A Comparative Study, *Applied Optimization*, Vol. 44, Kluwer Academic, 2000.

Flemming, P. J., Computer Aided Design of Regulators Using Multi-objective Optimization. *IFAC Control Applications of Nonlinear programming and Optimization*, Capri, Italy, 1985

Fonseca, C. M., and Fleming, P. J., An overview of Evolutionary Algorithms in Multi-objective Optimization, *Evolutionary Computation*, Vol. 3, No. 1, pgs. 1-16, 1995

Gergely, S., Bekassy-Molnar, E., and Vatai, G., The Use of Multi-objective Optimization in Improved Wine Filtration, *Journal of Food Engineering*, Vol. 58, pgs 311-316, 2003

Halsall-Whitney, H., Taylor, D., and Thibault, J., Multicriteria Optimization of Gluconic Acid Production Using Net Flow, *Bioprocess and Biosystems Engineering*, Vol. 25, No. 5, pgs. 299-307, 2003

Herreros, A., Baeyens, E., and Peran, J. R., MRCD: A Genetic Algorithm for Multi-objective Robust Control Design, *Engineering Applications of Artificial Intelligence*, Vol. 15, pgs. 285-301, 2002

Hu, A., Salcudean, S. E., and Loewen, P. D., A Numerical Solution of Multiobjective Control Problems, *Optimal Control Applications and Methods*, Vol. 19, pgs. 411-422, 1998

Ichikawa, Y., and Sawa, T., Neural Network Application for Direct Feedback Controllers, *IEEE Transactions on Neural Networks*, Vol. 3, No. 2, pgs. 224-231, 1992

Kang, D., and Bien, Z., Design of Multiobjective Satisfactory Fuzzy Logic Controller Using Reinforcement Learning, *International Journal of Fuzzy Systems*, Vol. 2, No. 2, 2000

Kim, J., Moon, Y., and Zeigler, B. P., Designing Fuzzy Net Controllers Using genetic Algorithms, *IEEE Control Systems Magazine*, Vol. 15, Issue 3, pgs 62-72, 1995

Ko, D., and Moon, I., Multi-objective Optimization of Cyclic Adsorption Processes, *Industrial Engineering Chemical Research*, Vol. 41, pgs. 93-104, 2002

Koza, J. R., Keane, M. A., Yu, J., Bennett III, F. H., and Mydlowec, W., Automatic Creation of Human-Competitive Program Controllers by Means of Genetic Programming, *Genetic Programming and Evolvable Machines*, Vol. 1, pgs. 121-164, 2000

Kursawe, F., A Variant of Evolution Strategies for Vector Optimization, *Parallel Problem Solving From Nature I (PPSN-I)*, pgs 193-197, 1990

Linkens, D. A., and Nyongesa, H. O., Learning Systems in Intelligent Control: An Appraisal of Fuzzy, Neural, and Genetic Control Applications, *IEE Proceedings – Control Theory Applications*, Vol. 143, No. 4, pgs 367-386, 1996

Liu, G.P., Yang, J. B., and Whidborne, J. F., Multiobjective Optimisation and Control, *Research Studies Press*, England, 2003

Marcu, T., A Multi-objective Evolutionary Approach to pattern Recognition for Robust Diagnostics of Process Faults, *SAFEPROCESS '97: Fault Detection, Supervision, and safety for Technical Processes*, Vols. 1-3, pgs. 1183-1188, 1997

Marlin, T. E., Process Control: Designing Processes and Control Systems for Dynamic Performance, 2nd ed., McGraw Hill, NY, 2000

Marrison, C. I., and Stengel, R. F., Robust Control System Design Using Random Search and Genetic Algorithms, *IEEE Transactions on Automatic Control*, Vol. 42, No. 6, pgs. 835-839, 1997

Miller, J.A., Potter, W. D., Gandham, R. V., and Lapena, C. N., An Evaluation of Local Improvement Operators for Genetic Algorithms, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 5, pgs. 1340-1351, 1993

Oliveira, P., Sequeira, J., and Sentieiro, J., Selection of Controller Parameters Using genetic Algorithms, *Engineering Systems with Intelligence: Concepts, Tools, and Applications*, Kluwer Academic Publishers, pgs. 431-438, 1991

Onnen, C., Babuska, R., Kaymak, U., Sousa, J. M., Verbruggen, H. B., and Isermann, R., Genetic Algorithms for Optimization in Predictive Control, *Control Engineering Practice*, Vol. 5, Issue 10, pgs. 1363-1372, 1997

Patton, R. J., Chen, J., and Liu, G. P., Robust Fault Detection of dynamic Systems via Genetic Algorithms, *Proceedings of the Institution of Mechanical Engineers*, Vol. 211, Part 1, pgs 357-364, 1997

Perrin, E., Mandrille, A., Oumoun, M., Fonteix, C., and Marc, I., Optimisation Globale par Stratégie d'Évolution: Technique Utilisant la Génétique des Individus Diploïdes. *RAIRO-Recherche Opérationnelle*, Vol. 31, pgs.161-201, 1997

Poloni, C., Giurgevich, A., Onesti, L., and Pediroda, V., Hybridization of a Multi-objective Genetic Algorithm, a Neural Network, and a Classical Optimizer for Complex Design problem in Fluid Dynamics, *Computer Methods in Applied Mechanics and Engineering*, Vol. 186, (2-4), pgs. 403-420, 2000

Porter, B., and Jones, A. H., Genetic Tuning of PID Controllers, *Electronics Letters*, Vol., 28, No. 9, pgs. 843-844, 1992

Rajesh, J. K., Gupta, S. K., Rangaiah, G. P., and Ray, A. K., Multi-objective Optimization of Industrial Hydrogen Plants, *Chemical Engineering Science*, Vol. 56, pgs. 999-1010, 2001

Rangan, S., and Poolla, K., Weighted Optimization for Multi-objective Full-information Control Problems, *Systems and Letters*, vol. 31, pgs. 207-213, 1997

Rovira, D. E., Marari, M., and Skogestad, S., Internal Model Control for PID Controller Design, *I&EC Process Design and Development*, Vol. 25, pg. 252, 1981

Roy, B., ELECTRE III: Algorithme de Classement Basé sur une résentation Floue des Préférences en presence de Critères Multiples. Cahiers du CERO, Vol. 20, pgs.3-24, 1978

Roy, B., The Outranking Approach and the Foundations of the ELECTRE Methods, *Theory and Decision*, Vol. 31, pgs 49-73, 1991

Scarelli, A., and Narula, C. N., A Multicriteria Assignment Problem, *Journal of Multi-Criteria Decision Analysis*, Vol. 11, pgs 65-74, 2002

Schubert, W. M., and Stengel, R. F., Parallel Synthesis of Robust Control Systems, *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 6, pgs. 701-706, 1998

Shim, M., Suh, M., Furukawa, T., Yagawa, G., and Yoshimura, S., Pareto-based Continuous Evolutionary Algorithms for Multi-objective Optimization, *Engineering Computations*, Vol. 19, No. 1, pgs. 22-48, 2002

Silva, C. M., and Biscaia, E. C., Genetic Algorithm Development for Multi-objective Optimization of Batch Free-radical Polymerization Reactors, *Computers and Chemical Engineering*, Vol. 27, pgs 1329-1344, 2003

Thibault, J., Taylor, D., and Fonteix, C., Multicriteria Optimization for the Production of Gluconic Acid, *Proceedings of the 8th International Conference on Computer Applications in Biotechnology: Modeling and Control of Biological Processes*, IFAC, Quebec, pgs. 24-27 2001;

Tzes, A., Peng, P. Y., and Guthy, J., Genetic-Based Fuzzy Clustering for DC-Motor Friction Identification and Compensation, *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 4, pgs. 462-472, 1998

Viennet, R., Fonteix, C., and Marc, I., Multicriteria Optimization Using a Genetic Algorithm for Determining a Pareto Set, *International Journal of Systems Science*, Vol. 27, No. 2, pgs. 255-260, 1995

Vlachos, C., Williams, D., and Gomm, J. B., Genetic Approach to Decentralized PI Controller Tuning for Multivariable Processes, *IEE Proceedings – Control Theory and Applications*, Vol. 146, No. 1, pgs 58-64, 1999

Wang, P. and Kwok, D. P., Auto-tuning of Classical PID Controllers Using an Advanced Genetic Algorithm, *International Conference on Industrial Electronics, Control, Instrumentation and Automation*, IECON 92, Vol. 3, pgs. 1224-1229, 1992

Wang, F., and Sheu J., Multi-objective Parameter Estimation Problems of Fermentation processes using a High Ethanol Tolerance Yeast, *Chemical Engineering Science*, Vol. 55, pgs. 3685-3695, 2000

Yee, A K. Y., Ray, A. K., and Rangaiah, G. P., Multi-objective Optimization of an Industrial Styrene Reactor, *Computers and Chemical Engineering*, Vol. 27, pgs. 111-130, 2003

Chapter 2

Paper 1

Multi-Objective Optimization for Chemical Processes and Controller Design:

Approximating and Classifying the Pareto Domain

Hayley Halsall-Whitney and Jules Thibault

Department of Chemical Engineering
University of Ottawa, Ottawa, Ontario,
Canada K1N 6N5
Tel: (613) 562-5920; Fax: (613) 562-5172
thibault@genie.uottawa.ca

Journal Publication: Computers and Chemical Engineering
Publisher: Elsevier
Publication Status: Submitted

Abstract

In recent years, the development of multi-objective optimization techniques for simultaneously optimizing multiple and conflicting objectives have received wide attention in the literature. However, most of these techniques have concentrated primarily on the generation of the Pareto domain without addressing the resulting problem of selecting the best solution from the available alternatives identified.

In this paper, three algorithms for generating the Pareto domain were studied in the development of a multi-objective optimization strategy. Twelve standard test cases, which have been used frequently in the literature, were considered along with two engineering problems, namely, the determination of optimum operating conditions for the production of gluconic acid and the determination of optimal tuning parameters for a PI controller. These multi-objective optimization problems were selected in order to evaluate the robustness and versatility of the algorithms studied. The results of three of these test cases and both engineering problems are presented.

The results identify a robust optimization strategy that generates a Pareto domain using a Dual Population Evolutionary Algorithm and classifies it using Net Flow, a technique that incorporates the knowledge of an expert into the optimization routine.

Keywords: Multicriteria Optimization, Pareto domain, Net Flow, Gluconic Acid Production, PI Controller Tuning

Nomenclature

C	Dissolved oxygen concentration in the broth, $g.L^{-1}$ and Global concordance index
CV_p	Controlled variable response
c_k	Individual concordance index
C^*	Concentration of oxygen in liquid in equilibrium with gas phase, $g.L^{-1}$
D_k	Discordance index,
K_p	Process gain
K_c	Controller gain
K_La	Volumetric oxygen transfer coefficient, h^{-1}
k_l	Michaelis constant for lactone production, $g.L^{-1}$
k_p	Gluconolactone hydrolysis rate constant, h^{-1}
k_o	Monod rate constant of growth with respect to oxygen, $g.L^{-1}$
k_s	Monod rate constant of growth with respect to glucose, $g.L^{-1}$
l	Gluconolactone concentration, $g.L^{-1}$
MV	Manipulated variable
p	Gluconic acid concentration, $g.L^{-1}$
p_f	Final Gluconic acid concentration, $g.L^{-1}$
p_f/t_B	Gluconic acid productivity, $g.L^{-1}.h^{-1}$
P_k	Preference threshold
P/V	Power supply per unit volume for agitation in the reactor, $W.m^{-3}$
Q_k	Indifference threshold
S	Substrate concentration, $g.L^{-1}$
SP	Set point
S_o	Initial substrate concentration, $g.L^{-1}$
t_B	Batch time, h
t	Time, h
Δt	Integration time step, *min*
U_g	Superficial gas velocity of air entering the reactor, $m.s^{-1}$
V_k	Veto threshold
v_l	Velocity constant for lactone production, $mg.UOD^{-1}.h^{-1}$

W_k	Weight, dimensionless
X	Cell concentration, *UOD.ml⁻¹*
X_o	Initial cell concentration, *UOD.ml⁻¹*
Y_o	Yield of growth based on oxygen, *UOD.mg⁻¹*
Y_s	Yield of growth based on glucose, *UOD.mg⁻¹*

Greek Characters

μ_m	Maximum specific growth rate, h^{-1}
τ_p	Process time constant, *min*
τ_I	Integral time constant, *min*
θ	Dead time, *min*
Γ	Delay

1 Introduction

In complex chemical processes, the ability to select optimum operating conditions in the presence of multiple conflicting objectives, given the various constraints of the process, can present a major challenge. While in process control, determining optimum tuning parameters for a controller in order to achieve simultaneously various design objectives is a multi-objective problem with conflicting criteria. In both cases, the application of a single-objective optimization technique does not always provide acceptable results.

Traditionally, optimization techniques have dealt with multiple objectives by combining them into a single objective function composed of the weighted sum of the individual objectives, or by focusing on one objective while transforming the others into constraints (Edgar *et al.*, 2001). These techniques are based on the assumption that the objective functions are well behaved, in the sense that they are either concave shaped or convex shaped and continuous, and that there exists an optimal point that will resolve the issue of conflicting objectives. Although single objective algorithms are sufficient in many cases for optimization, there are many drawbacks to their use, especially when the objectives under consideration are conflicting. Some of these issues are listed as follows.

1. Combining multiple objectives into a single objective function does not provide the decision-maker with information about trade-offs amongst the various objectives, or about alternative operating conditions.
2. Transforming a multi-objective problem into a single objective function composed of the weighted sum of the objectives relies only, and sometimes heavily, on the selection of weights, which is not a trivial task. The application of different weights leads to a variety of solutions.
3. An optimization technique that constrains some of the objectives may bias the final solution, which can result in misleading conclusions. The decision in regards to which objective function to optimize may not be a simple one.
4. Single objective optimization techniques provide only one optimal operating solution even if there are other possible solutions. These techniques are often plagued with the problem of finding the global optimum or multiple global optima, and could miss

solutions if the functions are non-convex, multi-modal or discontinuous. They also require information about function derivatives and an initial estimate of the solution, which may not be readily available.

5. Techniques that provide only one optimal operating solution instead of an optimal operating region do no take into account the difficulty associated with controlling the process.
6. Traditional optimization techniques do not incorporate the practical experience and knowledge of the decision-maker in regard to the overall behavior of the process.

In recent years, the development and application of multi-objective optimization techniques to chemical engineering problems (Silva and Biscaia, 2003; Costa *et al.*, 2003; Gergely *et al.*, 2003; Yee *et al.*, 2003; Bhaskar *et al.*, 2001; Rajesh *et al.*, 2001) and robust controller design (Liu et al., 2003; El-Kady *et al.*, 2003; Herreros *et al.*, 2002; Kang and Bien, 2000; Hu *et al.*, 1998; Rangan and Poolla, 1997; Fonseca and Fleming, 1994; Porter and Jones, 1992) have received wide attention in the literature, with varying degrees of success.

This paper examines the robustness of three methods for approximating the Pareto domain, and proposes an optimization strategy that generates the Pareto domain using a Dual Population Evolutionary Algorithm and classifies the entire domain using Net Flow, a technique that incorporates the knowledge of an expert into the optimization routine.

2 Theory

2.1 Multi-objective Optimization

Multi-objective optimization refers to the simultaneous optimization of multiple conflicting objectives, which produces a set of alternative solutions called the Pareto domain. These solutions are optimal in the sense that no one solution is better than any other in the domain when compared on all criteria. The decision-maker's experience and knowledge is then incorporated into the optimization procedure in order to classify the available alternatives in terms of his or her preference (Doumpos and Zopounidis, 2002).

Multi-objective Optimization Problem

A multi-objective optimization problem is one in which multiple objectives are explicitly maximized, minimized or a combination of both, can be subjected to constraints that the feasible solutions must satisfy, and where the ranges for the input space variables are defined. A multi-objective optimization problem can be defined as follows.

$$Min\,/\,Max\,/\,Combination \quad F(x) = \left[f_1(x_1..x_n), f_2(x_1..x_n), ... f_n(x_1..x_n) \right]$$

$$\begin{aligned} \textit{subject to} \quad & G_{i=1...n}(x) \geq 0, \\ & H_{i=1...n}(x) = 0, \quad i = 1,2,3...n \end{aligned} \qquad ....(1)$$

$$\textit{where} \qquad x_{i=1...n}^{(Lower\ Bound)} \leq x_{i=1...n} \leq x_{i=1...n}^{(Upper\ Bound)}$$

The input space is predefined by the ranges associated with the independent variables $\boldsymbol{X} = (x_1, x_2, x_3...x_n)^T$. This input space is called the *Decision Variable Space* in the literature, while the output or solution space, expressed by ***F(x)***, is called the *Pareto domain* (Deb, 2001).

Concept of Domination and Pareto Optimal Solutions

The concept of dominance is an approach used to determine a fitness score for each member of a population of data points. It was first introduced as a method of determining a point's fitness within the current population of data by MOGA, Multi-objective Genetic Algorithm (Fonseca and Fleming, 1993). The concept of domination is defined in this paper as follows. Given two points, $\boldsymbol{X_1} = (x_1, x_2, x_3...x_n)^T$ and $\boldsymbol{X_2} = (x_1, x_2, x_3...x_n)^T$ within a population's decision variable space, the following properties are defined.

1. $\boldsymbol{X_1}$ dominates $\boldsymbol{X_2}$ if the values of all objective criteria for $\boldsymbol{X_1}$ are better than the values for the corresponding objective criteria for $\boldsymbol{X_2}$.
2. $\boldsymbol{X_1}$ does not dominate $\boldsymbol{X_2}$ if the value of at least one of the objective criteria for $\boldsymbol{X_1}$ is equal or worse than the value of the corresponding objective criterion for $\boldsymbol{X_2}$.

If X_1 dominates X_2 then X_2 is called a dominated point, and if X_1 does not dominate X_2 and X_2 does not dominate X_1 then both are called non-dominated points with respect to one another.

Pareto optimal solutions, called the *Pareto domain*, are a set of non-dominated solutions where no one solution is better than any other in the domain when compared on all objective criteria. The Pareto domain is generated by simultaneously optimizing multiple criteria. It provides the decision-maker with information about the trade-offs and alternative solutions arising from the use of multiple conflicting objectives. If the objectives are not conflicting the Pareto domain will converge to one solution (Deb 2001). In this case a single criteria optimization technique may be more appropriate.

2.2 Optimization Strategy

The multi-objective optimization strategy proposed in this work consists primarily of the following two steps.

1. The generation of the Pareto domain.
2. The classification of the domain by ranking the data using the Net Flow method, which incorporates *a priori* process knowledge supplied by the decision-maker.

2.2.1 Techniques for Approximating the Pareto Domain

The algorithms studied for generating the Pareto domain include the Single Population Evolutionary Algorithm, Dual Population Evolutionary Algorithm and the Grid Search Approach. The Single Population Evolutionary Algorithm (SPEA) has been used in the literature under the general title of Evolutionary Algorithm or Genetic Algorithm (Thibault *et. al.*, 2001; Perrin *et al.*, 1997). The Dual Population Evolutionary Algorithm (DPEA) has not been identified in the literature and was developed, in this work, as an improved form of the Single Population Evolutionary Algorithm. The Grid Search Approach (GSA) has been used in the literature with varying descriptions for the algorithm. This paper presents another approach.

General Evolutionary Approach

The evolutionary algorithms studied in this work, SPEA and DPEA, incorporate the concept of domination and fitness score assignment, otherwise called a domination score, when generating the Pareto domain. The general approach used is described as follows (Halsall-Whitney *et al.*, 2003; Thibault *et. al.*, 2001; Perrin *et al.*, 1997).

1. For each input variable, a value was randomly selected from within its predefined range of variation, using a random number generator that allowed for the selection of the boundary values of the input ranges. The model was then solved for this set of inputs in order to generate values for the objective criteria. Initially, M points were generated to fully span the decision space using a pseudo-random number generator, which was altered so that the boundaries of the decision space would be taken into consideration. The number of points M is specified by the user and must be large enough to adequately approximate the Pareto domain.
2. All M points were compared two by two, based on the values for their objective criteria, in order to determine the number of times a point had been dominated by another in the current output space. Each time a point was dominated, its domination score was incremented by one; otherwise it remained unchanged from an initial score of zero. At the end of the comparison process, the data set was sorted in order of domination scores, starting from data points that were never dominated (i.e. having a score of 0) and ending with the data point displaying the highest frequency of domination.
3. All non-dominated data points, N_0, were then selected, along with a fraction of the dominated points having the lowest frequency of domination, to be used in generating the next approximation to the Pareto Domain. This selection process can be described by the following equation:

$$N = N_0 + INT\left[(F_s)(M - N_0)\right] \qquad (2)$$

In this equation, F_S is the survival fraction of the dominated data points and $INT(x)$ yields the integer value of x. In this study a survival fraction of 0.3 was used. This ensures that only the better elements of the population survived to participate in the creation of the new generation of possible solutions.

4. A total of *(M-N)* new data points were then generated to replace those points that were eliminated during the selection process. Combining these new points with those points selected from the previous set yielded a new set of *M* data points. An evolutionary algorithm, SPEA or DPEA, was used to generate each new point, $I_{p,k}$, by randomly selecting two points, $I_{p,i}$ and $I_{p,j}$, from the set of points retained in the previous generation. The new set of inputs for the new point was then determined according to the following equation.

$$I^{n}_{p,k} = D_p I^{n}_{p,i} + (1 - D_p) I^{n}_{p,j} \qquad (3)$$

   In this equation, D_p is a randomly selected number between 0 and 1. A new value for D_p was selected each time a new input, in the vector of inputs representing $I_{p,k}$, was determined.

5. Steps (1)-(4) were repeated until *M* non-dominated data points, which was specified by the user as the number of points required to adequately define the Pareto domain, were generated. By eliminating in successive generations all data points that were not Pareto-efficient, it was assumed that this final set of *M* possible solutions adequately approximates the Pareto domain.

The Single Population Evolutionary Algorithm and the Dual Population Evolutionary Algorithm differ in the method used to select points $I_{p,i}$ and $I_{p,j}$ from the set of points retained in the previous generation in step 4.

Single Population Evolutionary Algorithm

The Single Population Evolutionary Algorithm randomly selects two points from the entire population of *N* points retained for the previous iteration to generate a new point, regardless of their domination score. The process is repeated until *(M-N)* data points are generated. The two points selected can be either (1) both dominated points, or (2) both non- dominated points, or (3) one dominated point and one non-dominated point. This approach does not discriminate amongst the solutions in the population of N points. Any combination of two points can be selected to create a new point.

Dual Population Evolutionary Algorithm

The Dual Population Evolutionary Algorithm randomly selects one non-dominated point and one dominated point from the set of *N* points retained from the previous iteration to generate a new point. The process is repeated until *(M-N)* data points are generated. This approach views the current set of *N* points as consisting of two distinct populations, namely, the population of non-dominated points and the population of dominated points. It is believed that this method will better approximate the boundary of the decision variable space.

Grid Search Approach

The Grid Search Approach is not an evolutionary algorithm. Instead it generates the Pareto domain by selecting values for the input variables from their predetermined range of variation according to a decision space grid. However, this technique employs the concept of domination when generating the Pareto domain. The following method was used to generate the Pareto domain.

1. Using the predetermined range of variation for the input variables, the decision space is mapped out as a grid by initially dividing each range by 5 thus creating 5^n identical n-dimensional parallelepipeds, where n is the number of inputs. This avoids the use of pseudo-random number generators and increases the probability of selecting values for the input variables that will span the complete decision space. For each successive iteration, the number of divisions for each range was incremented by 5.
2. Values for the inputs were selected from the corners of each parallelepipeds outlined by the grid. The model was then solved for this set of inputs in order to generate values for the objective criteria. The number of points generated was $(m+1)^n$, where m is the number by which each range was divided, and n is the number of input variables.

3. The population of points generated was compared two by two, based on the values for their objective criteria, in order to determine all dominated and non-dominated points.
4. All the non-dominated points were selected, while the dominated points were discarded. A new range for each of the input variables was determined by locating the minimum and maximum values of the non-dominated point's input variables.
5. In order to avoid missing possible Pareto-efficient points, half the current length of a grid parallelepipeds expanded the new range for each input. If the new range for the input variables was equal to the original range for the input then this expansion did not occur.
6. If the new range is equal to the old range for all the inputs describing the decision space, then the current population of points represents the final Pareto domain. If not, then the new range replaced the old range and steps (1-5) were repeated.

Evaluating Algorithm Performance

These algorithms used for generating the Pareto domain were evaluated based on the following performance measures.

1. The efficiency of the algorithms in terms of the time required for converging to the Pareto domain.
2. The ability of the algorithm to produce a well-defined Pareto domain.
3. The total number of data points analyzed during the simulation before obtaining the final Pareto domain. This performance measure was applied only to the evolutionary algorithms studied.

2.2.2 Algorithm for Classifying the Pareto Domain

The optimization strategy used in this work incorporates a classification technique, called Net Flow, into the optimization strategy. This method incorporates *a priori* process knowledge supplied by the decision-maker, into the optimization process. The Pareto domain was established from a domination perspective without the bias of *a priori* knowledge concerning the relative importance of the various criteria. However, to classify each point in the domain the expert needs to incorporate his or her knowledge of the process into the optimization routine. The Net Flow method (NFM), which developed as a result of modifications made to the ELECTRE III method, was used to rank the Pareto domain (Doumpos and Zopounidis, 2002; Scarelli and Narula, 2002; Triantaphyllou, 2000; Derot *et al.*, 1997; Brans *et al.*, 1984; Roy, 1978).

In the NFM, *a priori* knowledge of the process, expressed by the decision maker, is incorporated into the optimization routine using four sets of parameters to rank all the data points of the Pareto domain. These parameters are described as follows:

1. The first parameter gives the relative importance of each criterion, expressed as a weight W_k. In this algorithm, the weights are normalized:

$$\sum_{k=1}^{n} W_k = 1 \qquad (4)$$

2. The second parameter refers to the indifference threshold (Q_k), which defines the range of variation of each criterion for which it is not possible for the decision-maker to favor the criterion of one point over the corresponding criterion of another.
3. The third parameter refers to the preference threshold (P_k). If the difference between two values for a given criterion exceeds this threshold, a preference was given to the better criterion.
4. The fourth parameter refers to the veto threshold (V_k), which serves to ban a point relative to the other if the difference between the values of a criterion was too high to be tolerated. A point was banned if the veto threshold was violated for at least one of the criteria even if the other criteria were acceptable.

These three thresholds are defined for each criterion such that the following relationship holds:

$$0 \le Q_k \le P_k \le V_k \qquad (5)$$

They represent a reference range established by the decision-maker to assess the values of the objective criteria for each alternative in the Pareto domain (Roy, 1978). Although this method makes use of weights it does not only rely on them, but instead uses them in combination with the other parameters, which incorporate additional preferences of the decision-maker. The NFM algorithm is described as follows.

1. First, for each combination of points in the Pareto domain, the difference between the values F_k of each criterion k was calculated by comparing point *i* with point *j* using the following equation.

$$\Delta_k[i,j] = F_k(i) - F_k(j) \quad \begin{cases} i \in [1, M] \\ j \in [1, M] \ , j \neq i \\ k \in [1, n] \end{cases} \qquad (6)$$

   In subsequent equations, minimizing a criterion required using $\Delta_k[i,j]$, while maximizing a criterion required using $-\Delta_k[i,j]$.

2. Using the values of $\Delta_k[i,j]$, the concordance index $c_k[i,j]$ for each criterion was determined for all *n* criteria and for each pair of data points using the following relationships.

$$c_k[i,j] = \begin{cases} 1 & \text{if } \Delta_k[i,j] \le Q_k \\ \dfrac{P_k - \Delta_k[i,j]}{P_k - Q_k} & \text{if } Q_k < \Delta_k[i,j] \le P_k \\ 0 & \text{if } \Delta_k[i,j] > P_k \end{cases} \qquad (7)$$

   The concordance index measures the strength of the argument that when comparing point *i* to point *j* for a given criterion k the value of '*F_k (i) is at least as good as F_k (j)*' when compared to values specified by the decision-maker in the reference range for a given criterion (Roy, 1978). Figure 2a illustrates how the concordance index was determined using the values of the calculated differences, the indifference threshold, and the preference threshold. For a difference smaller than the indifference threshold, the corresponding concordance index is 1. Between the indifference and preference thresholds, it varies linearly from 1 to 0. For a difference larger than the preference threshold, the concordance index was set to 0.

3. The weighted sum of individual concordance indices was calculated to determine the global concordance index.

$$C[i,j] = \sum_{k=1}^{n} W_k c_k [i,j] \quad \begin{cases} i \in [1,M] \\ j \in [1,M] \end{cases} \qquad (8)$$

4. A discordance index $D_k[i, j]$ was calculated for each criterion k using the preference and veto thresholds.

$$D_k[i, j] = \begin{cases} 0 & \text{if } \Delta_k[i, j] \leq P_k \\ \dfrac{\Delta_k[i, j] - P_k}{V_k - P_k} & \text{if } P_k < \Delta_k[i, j] \leq V_k \\ 1 & \text{if } \Delta_k[i, j] > V_k \end{cases} \qquad (9)$$

The discordance index measures the strength of the argument that when comparing point *i* to point *j* for a given criterion k the value of 'F_k *(i) is significantly worse than* F_k *(j)*' when compared to values specified by the decision-maker in the reference range for a given criterion (Roy, 1978). Figure 2b illustrates how the discordance index was determined using the preference and veto thresholds. For a difference smaller than the preference threshold, the discordance index is 0. Between the preference and veto thresholds, it varies linearly from 0 to 1, and for a difference larger than the veto threshold, the discordance index is set to 1.

5. Using the concordance and discordance indices, the relative performance of each pair of domain points was evaluated by calculating each element of the outranking matrix $\sigma[i, j]$ using the following equation.

$$\sigma[i, j] = C[i, j]\left(\prod_{k=1}^{n} 1 - (D_k[i, j])^3\right) \begin{cases} i \in [1, M] \\ j \in [1, M] \end{cases} \qquad (10)$$

Each element $\sigma[i, j]$ measures of the quality of point *i* relative to point *j* in terms of all criteria. An element $\sigma[i, j]$ close to 0 indicates that point *j* outranks point *i*. If the value is near 1, then data point *i* may outrank point *j* or simply be located in the vicinity of point *j*.

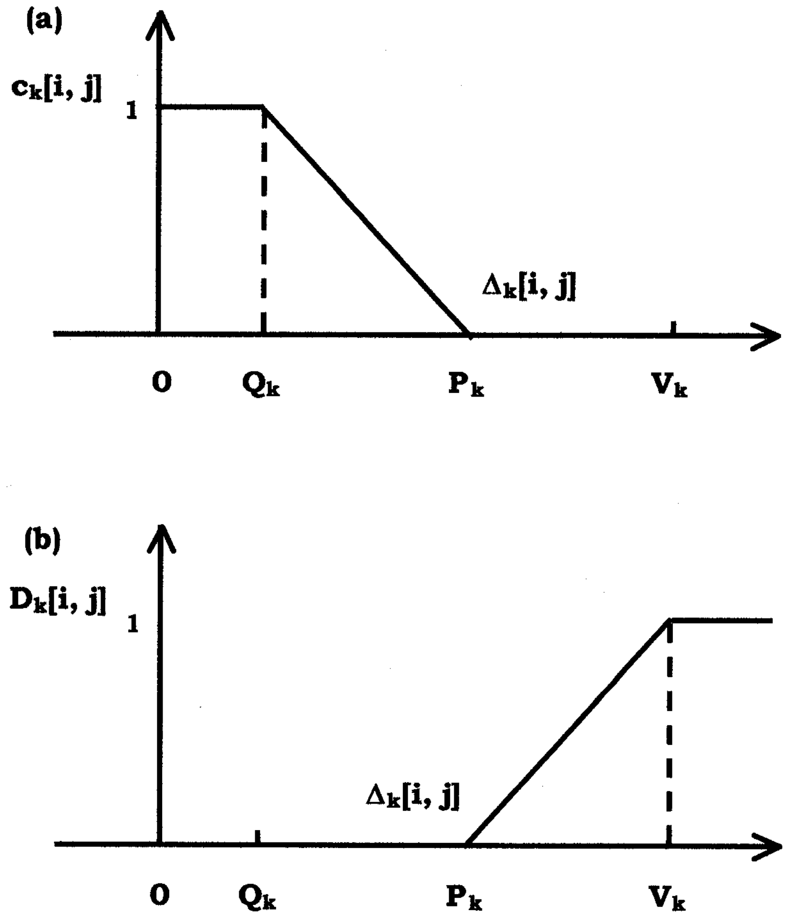

Figure 1 -(a) Concordance index, and (b) discordance index calculations used in the Net Flow algorithm to determine ranking scores for the Pareto domain points

6. The final ranking score for each data point in the Pareto domain was obtained by summing individual outranking elements associated with each domain point as follows.

$$\sigma_i = \sum_{j=1}^{M} \sigma[i, j] - \sum_{j=1}^{M} \sigma[j, i] \qquad (11)$$

The first term evaluates the extent to which element i performs relative to all the other points in the Pareto domain, while the second term evaluates the performance of all the other points relative to point i. The points were then sorted from highest to lowest according to ranking score. The data point with the highest ranking was the one that best satisfies the set of preferences provided by the decision-maker.

Finally, the results of the Net Flow method was used to divide the Pareto domain into zones containing the high-ranked, mid-ranked, and low-ranked domain points in order to identify graphically where the optimal region was located.

2.3 Case Studies

In this work twelve standard test cases found in the literature and two engineering problems were used to evaluate the performance and efficiency of the Single Population Evolutionary Algorithm, the Dual Population Evolutionary Algorithm and the Grid Search Approach in approximating the Pareto domain (Halsall-Whitney, 2004). The results of three standard test cases and both engineering problems are presented in this paper.

The test cases represent a variety of multi-objective optimization problems with varying degrees of difficulty. They are designed to test the ability of multi-objective optimization algorithms to converge to the true Pareto domain. The test cases used in this work were selected based on their ability to produce Pareto domains that were concave shaped, convex shaped, non-concave and non-convex shaped, and disjoint. Test cases involving multi-modal functions were also studied. The two engineering case studies selected included a biochemical engineering problem, which involved determining optimum operating conditions for the production of gluconic acid, and a control design problem, which involved determining optimum tuning parameters for a PI controller.

2.3.1 Test Cases

Test Case Model 1: Convex Shaped Pareto Domain

Test case model 1 is a 2×2 multi-objective optimization problem that has a convex shaped Pareto domain with a uniform and simple decision space. The ranges for the inputs are $X_1 \in [-50, 50]$ and $X_2 \in [-50, 50]$ (Deb, 2001).

$$\begin{aligned} Min\ f_1(X_1, X_2) &= (X_1)^2 + (X_2)^2 \\ Min\ f_2(X_1, X_2) &= (X_1+2)^2 + (X_2)^2 \end{aligned} \qquad \text{..........(12)}$$

Test Case Model 2: Non-convex, Non-concave Pareto Domain

Test Case Model 2 is a complex 2×3 multi-objective optimization problem that has a well-defined Pareto domain with a disconnected but well defined decision space. The ranges for the inputs are $X_1 \in [-3, 3]$ and $X_2 \in [-3, 3]$ (Viennet, 1995).

$$\begin{aligned} Min\ f_1(X_1, X_2) &= 0.5\left[(X_1)^2 + (X_2)^2\right] + \sin\left[(X_1)^2 + (X_2)^2\right] \\ Min\ f_2(X_1, X_2) &= \frac{(3X_1 - 2X_2 + 4)^2}{8} + \frac{(X_1 - X_2 - 1)^2}{27} + 15 \\ Min\ f_2(X_1, X_2) &= \frac{1}{(X_1)^2 + (X_2)^2 + 1} - 1.1\exp\left[-\left((X_1)^2 + (X_2)^2\right)\right] \end{aligned} \qquad \text{..........(13)}$$

Test Case Model 3: Disjoint Pareto Domain

Test Case Model 3 is a complex 2×2 multi-objective optimization problem that has a non-convex output space containing a Pareto domain with two distinct disconnected regions with a disconnected but well defined decision space. The ranges for the inputs are $X_1 \in [-\pi, \pi]$ and $X_2 \in [-\pi, \pi]$ (Poloni, 2000).

$$Min\ f_1(X_1, X_2) = 1 + (A_1 - B_1)^2 + (A_2 - B_2)^2$$
$$Min\ f_2(X_1, X_2) = (X_1 + 3)^2 + (X_2 + 1)^2$$

where

$$A_1 = 0.5sin(1) - 2cos(1) + sin(2) - 1.5cos(2) = 0.8736 \quad \text{.....(14)}$$
$$A_2 = 1.5sin(1) - cos(1) + 2sin(2) - 0.5cos(2) = 2.7486$$
$$B_1 = 0.5sin(X_1) - 2cos(X_1) + sin(X_2) - 1.5cos(X_2)$$
$$B_2 = 1.5sin(X_1) - cos(X_1) + 2sin(X_2) - 0.5cos(X_2)$$

2.3.2 Chemical Engineering Applications

2.3.2.1 Gluconic Acid Production

Gluconic Acid Production Model

This model simulates the fermentation of glucose to gluconic acid by the micro-organism *Pseudomonas ovalis* in a batch stirred tank reactor. The overall mechanism can be expressed as follows (Johansen and Foss, 1995):

Cells + Glucose + Oxygen → More cells

Glucose + Oxygen → Gluconolactone

Gluconolactone + Water → Gluconic Acid

The following state-space model has been derived to represent, respectively, the concentration of cells (*X*), gluconic acid (*p*), gluconolactone (*l*), glucose substrate (*S*), and dissolved oxygen (*C*) (Ghose and Gosh, 1976):

$$\frac{dX}{dt} = \mu_m \frac{SC}{k_s C + k_o S + SC} X \quad \text{.....(15)}$$

$$\frac{dp}{dt} = k_p l \quad \text{.....(16)}$$

$$\frac{dl}{dt} = v_l \frac{S}{k_l + S} X - 0.91 k_p l \qquad (17)$$

$$\frac{dS}{dt} = -\frac{1}{Y_s} \mu_m \frac{SC}{k_s C + k_o S + SC} X - 1.011 v_l \frac{S}{k_l + S} X \qquad (18)$$

$$\frac{dC}{dt} = K_L a \left(C^* - C\right) - \frac{1}{Y_o} \mu_m \frac{SC}{k_s C + k_o S + SC} X - 0.09 v_l \frac{S}{k_l + S} X \qquad (19)$$

Table 1 lists the values of the coefficients used in these equations to model the production of gluconic acid.

Table 1 Gluconic acid production model parameter values

Parameter	Value	Unit
μ_m	0.39	h^{-1}
k_s	2.50	*g/L*
k_o	0.00055	*g/L*
k_p	0.65	h^{-1}
v_l	8.30	*mg/UOD.h*
k_l	12.80	*g/L*
Y_s	0.38	*UOD/mg*
Y_o	0.89	*UOD/mg*
C*	0.00685	*g/L*
ζ	30.0	*dimensionless*
β	0.6	*dimensionless*
α	0.5	*dimensionless*

Optimization Objectives

While numerous objective criteria could be identified for optimizing the production of gluconic acid, this work focused on maximizing the overall production rate (p_f/t_B) and the final concentration (p_f) of gluconic acid, while minimizing the final substrate concentration (S_f) at the end of the fermentation process. The simulations studied differed in terms of the inputs used to define the decision space. The selection of inputs included the duration of the fermentation process ($t_B \in [5, 15\ h]$), the initial substrate concentration ($S_0 \in [20, 50\ g/L]$), the initial biomass concentration ($X_0 \in [0.05, 1.0\ UOD/mL]$), and the oxygen mass transfer coefficient ($K_La \in [50, 300\ h^{-1}$) (Halsall-Whitney *et al.*, 2003; Thibault *et al.*, 2001). Figure 2 represents the multi-input, multi-output optimization strategy used.

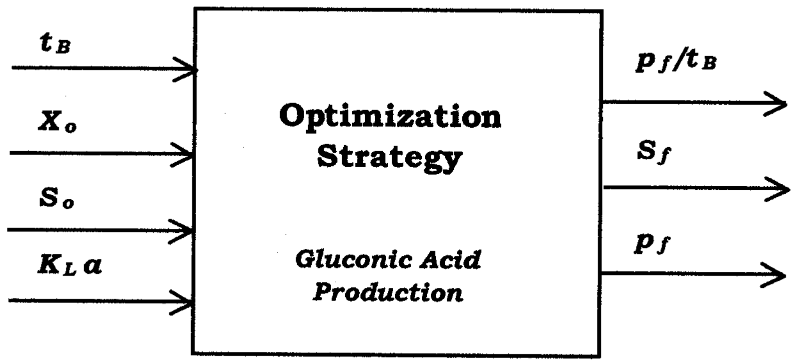

Figure 2 - Objectives used during the development of the Pareto domain and optimization of gluconic acid.

2.3.2.2 PI Controller Design

PI controller tuning parameters for a first-order-plus-dead-time process affected by set point changes was determined by simulating a single, closed loop feedback control system. Figure 3 represents the block diagram of a single loop feedback control system used during the simulation.

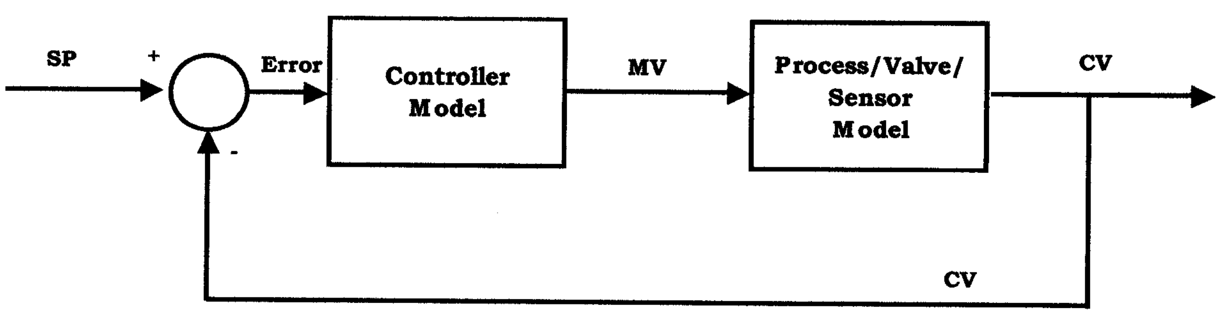

Figure 3 Block diagram of a single-loop feedback control system.

Closed Loop Control System Models

The first-order-plus-dead-time model was approximated using the following algebraic equation.

$$CV_{P_n} = \frac{K_p \Delta t}{\tau_p} MV_{(n-\Gamma)} + \left(1 - \frac{\Delta t}{\tau_p}\right) CV_{P(n-1)} \quad \text{..........(20)}$$

In this equation the dead time was simulated by a delay, Γ, where $\Gamma = INT\ (\theta/\Delta t)$(Marlin, 2000). The values used for the model parameters K_p, τ_p, θ and Δt, were 1, 5, 5 and 0.001, respectively. The models representing the valve and sensor were equated to 1.

PI Controller Algorithm

The velocity form of the PI controller was used during the simulation, which is described as follows.

$$MV_n = MV_{n-1} + K_c\left[\left(SP_n - CV_n\right) - \left(SP_{n-1} - CV_{n-1}\right)\right] + \left[\frac{K_c \Delta t}{\tau_I}\left(SP_n - CV_n\right)\right] \quad \text{..........(21)}$$

Optimization Objectives

Although there are a number of objective criteria that can be used to determine optimum tuning parameters for the PI controller, this research selected to minimize the ITAE, the ISDU and the settling time. Figure 4 illustrates this multi-input, multi-output strategy used.

The ITAE measures the cumulative deviation of the controlled variable from the set point. It penalizes deviations that endure for long periods of time, and is described using the following equation.

$$ITAE = \int_0^{\infty} t\left|SP - CV\right| dt \approx \sum_{n=1}^{M} t\left|SP_n - CV_n\right| \Delta t \quad \text{..........(22)}$$

The ISDU is a performance measure for the manipulated variable movements. The objective is to minimize excessive variation in the manipulated variable, and is described using the following equation.

$$ISDU = \sum_{n=1}^{M}\left(MV_n - MV_{n-1}\right)^2 \qquad (23)$$

The settling time is the amount of time that is required for the process to reach within ±5% of its new steady state value after experiencing a set point change. A short settling time is desired.

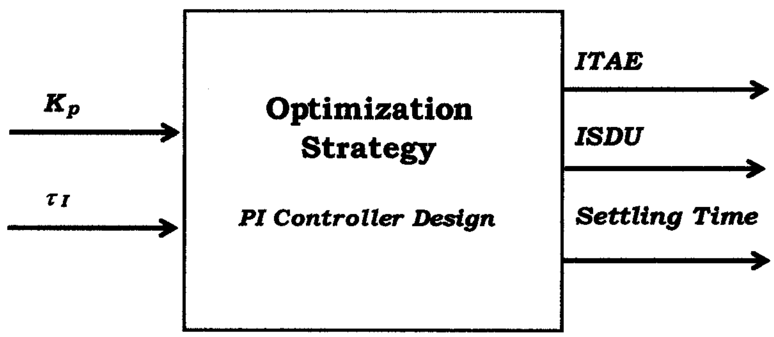

Figure 4 - Objectives used during the development of the Pareto domain and optimization of tuning parameters for a PI controller.

3 Results and Discussion

Three techniques were studied in the development of an optimization strategy that generates a Pareto domain and classifies it using Net Flow, using *a priori* information from the decision maker. The results for three test cases and two engineering problems are presented. In each case study both the decision space and the Pareto domain were generated using the SPEA, DPEA, and GSA. The results were divided according to their ranking score into three zones, namely, the high-ranked, the mid-ranked and the low-ranked points. These techniques used to generate the Pareto domain were analyzed based on their efficiency, in terms of time, and their ability to approximate the true domain. In the case of the SPEA and the DPEA an additional performance measure was used to include the number of data points analyzed before converging to the Pareto domain. The simulations were performed on a Pentium 4, 1.7 GHz, computer.

3.1 Test Cases

The Pareto domains generated using the evolutionary algorithms SPEA and DPEA consisted of 2500 unique non-dominated points. This ensured that the results were evenly distributed and representative of the true Pareto domain. The number of unique non-dominated points generated using GSA depended on the boundary of the decision space, which represents the input space for the Pareto domain, and the number of divisions used at the time it was located. The results generated using these three algorithms were then ranked using the Net Flow method. Table 2 gives the values of the Net Flow parameters used in Test Cases 1-3. Criteria 1, 2 and 3 represent F_1, F_2 and F_3, respectively.

Table 2 Net Flow parameter values used for classifying the Pareto domain for Test Cases 1-3

Test Case	Criterion k	NFM Parameters W_k	Q_k	P_k	V_k
1	1	0.5	0.5	1.0	1.5
	2	0.5	0.1	0.5	1.0
2	1	0.35	0.1	0.5	2.0
	2	0.3	0.01	0.05	0.5
	3	0.35	0.01	0.05	0.1
3	1	0.5	1.0	2.0	5.0
	2	0.5	1.0	2.0	4.0

Figure 5 presents the decision spaces and the Pareto domains generated using SPEA, DPEA, and GSA for Test Case Model 1. The objective was to minimize both objective criteria. The results show a Pareto domain that is convex shaped, where minimizing any criterion would maximize rather than minimize the other criterion. In order to determine the optimal solution that minimizes both criteria, a compromise solution must be found. Table 3 represents the optimal values found using each technique.

Table 3 Optimal values found using SPEA, DPEA, and GSA for Test Case 1

Algorithm	X_1	X_2	$F_1(X_1,X_2)$	$F_2(X_1,X_2)$
SPEA	-1.173	-0.00364	1.377	0.683
DPEA	-1.166	0.00566	1.359	0.696
GSA	-1.108	0	1.228	0.796

The results show that each technique was capable of locating approximately the same optimal point using the classification information. The domains generated are consistent with those obtained in the literature (Deb, 2001). A simple inspection of Equation (12) reveals that the decision space leading to the Pareto domain is comprised of $X_1 \in [-2.0, 0.0]$ and $X_2 = 0$. All techniques gave approximately the same results but only the GSA was able to locate with the greatest precision the decision space. The GSA is well suited for this type of two-dimensional optimization problem where the decision space lies on or is parallel to a grid axis. The other two techniques provided identical Pareto domains. Considering that the initial range of the input variables was between -50 and 50, the area of the decision space was determined fairly accurately by the SPEA and DPEA. To obtain a better accuracy, a number of points far in excess of 2500 would have to be used.

The SPEA analyzed 19493 data points in order to locate 2500 non-dominated points requiring 98 seconds to perform the optimization, whereas the DPEA analyzed 15159 and required 78 seconds. The GSA outperformed both the SPEA and the DPEA by locating the Pareto domain in 3 seconds, analyzing 3279 data points in order to locate 19 unique, non-dominated points to approximate the Pareto domain. In this case, a finer grid division of the X_1 direction could be done to provide a more defined Pareto domain and a more accurate value for the optimal point.

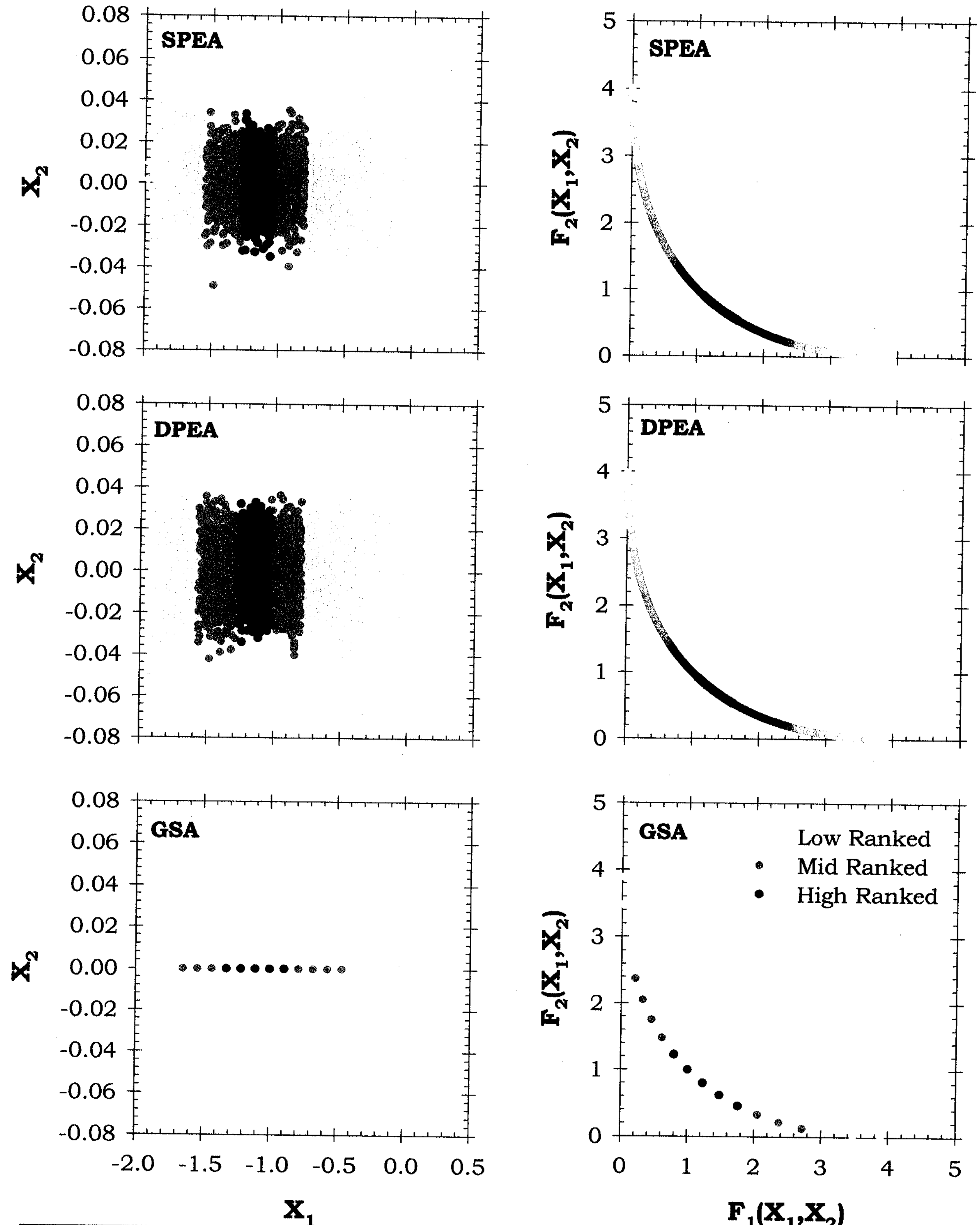

Figure 5 – Comparison of decision spaces and Pareto domains generated using SPEA, DPEA and GSA for Test Case Model 1

Test Case Model 2 has been used extensively in the literature, to demonstrate the performance of multi-objective optimization algorithms (Herreros *et al.*, 2002; Deb, 2001; Viennet, 1996). In most cases the evolutionary algorithms used could not locate the true Pareto domain, which was determined analytically by Deb, or the domain found was not well defined (Deb, 2001).

Figure 6 presents the decision spaces, and the Pareto domains, generated using SPEA, DPEA and GSA, respectively. The results show the Pareto domain and a disconnected decision space generated as a result of minimizing simultaneously the three criteria. Table 4 presents the optimal values found using each technique.

Table 4 - Optimal values found using SPEA, DPEA, and GSA for Test Case 2

Algorithm	X_1	X_2	$F_1(X_1,X_2)$	$F_2(X_1,X_2)$	$F_3(X_1,X_2)$
SPEA	-0.5483	0.3665	0.6389	15.3291	-0.0151
DPEA	-0.5562	0.3701	0.6549	15.3167	-0.0126
GSA	-0.6488	0.2704	0.7211	15.2864	-0.0019

Each method identified different values for the optimal point using the same preferences provided by the expert. During the simulation SPEA analyzed 18380 data points in order to locate 2500 non-dominated points, while DPEA analyzed 17891 data points. Both techniques took 131 seconds to generate the Pareto domain. Although the results suggest that both evolutionary algorithms had the same level of performance, the results are misleading. Figure 6 show that although both techniques located the true Pareto domain, SPEA located points in the decision space that were not located using the DPEA or the GSA technique. Although these points were ranked low, further analysis revealed that when both the SPEA and the DPEA Pareto domains were combined and re-analyzed according to domination, there were points in the DPEA domain that caused the data points in question to be re-classified as dominated points. Therefore DPEA was more efficient in generating the Pareto domain than SPEA. In the case of GSA, 1604 data points were analyzed, which resulted in the location of 34 non-dominated points. Although the GSA took 3 seconds to generate the Pareto domain, a comparison of the results with those generated using SPEA and DPEA shows that it was not capable of generating a well-defined Pareto domain for this case study.

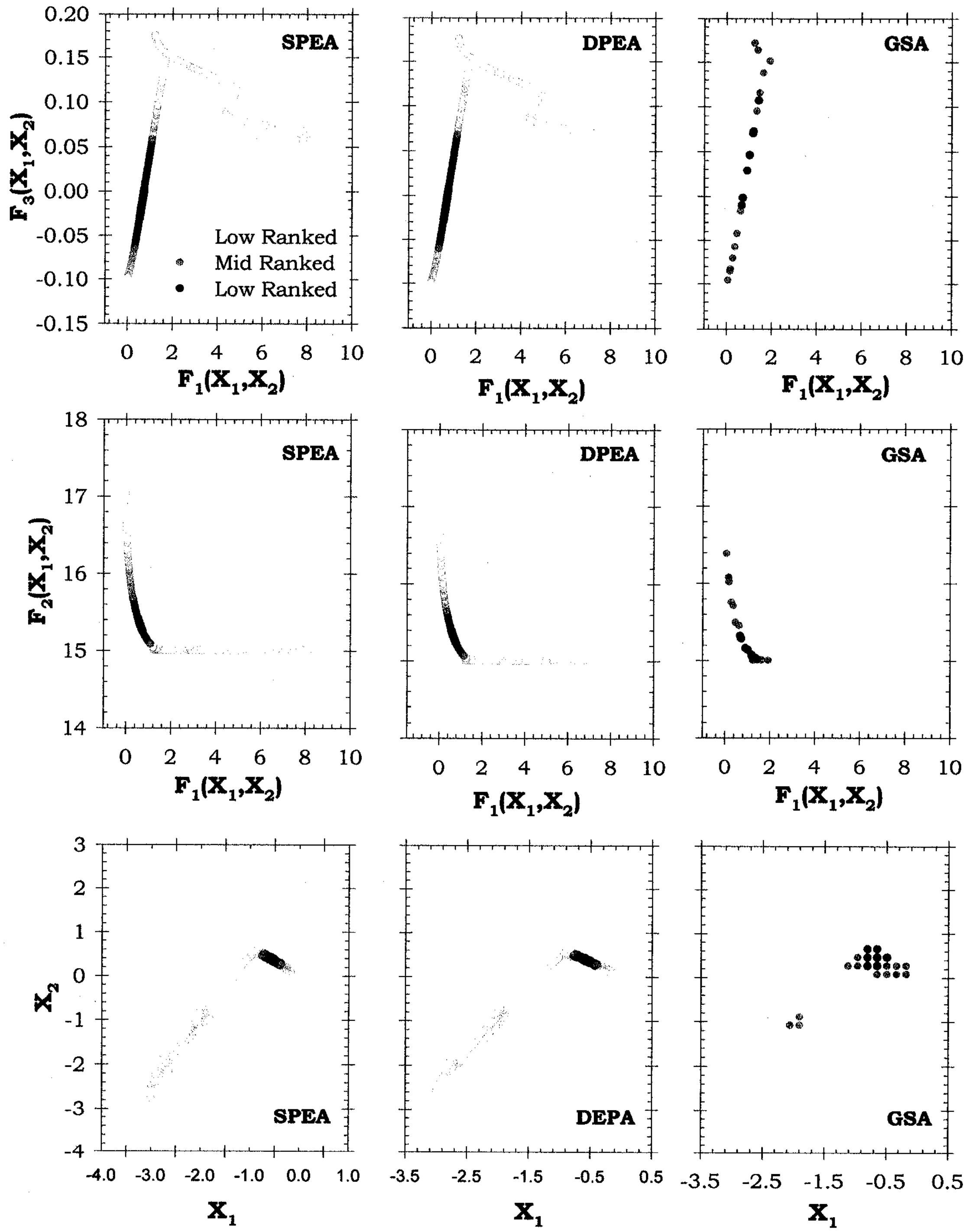

Figure 6 – Comparison of decision spaces and Pareto domains using SPEA, DPEA, and GSA for Test Case Model 2.

However, the GSA method successfully identified the region of feasible solutions. To get a better definition with the GSA, a significantly finer grid size would be required but at the expense of a longer execution time and the risk of running out of memory.

Figure 7 presents the decision spaces and Pareto domains for Test Case Model 3 generated using SPEA, DPEA and GSA. This case study has also been extensively used in the literature to demonstrate the performance of multi-objective optimization techniques (Deb, 2001; Poloni *et al.*, 2000). The results show the generation of two distinct regions of the decision space and Pareto domain. In terms of the evolutionary algorithms studied, both the SPEA and the DPEA were capable of generating the true Pareto domain. The SPEA analyzed 30570 data points in 190 seconds, while the DPEA method analyzed 23997 data points in 166 seconds in order to generate 2500 non-dominated points. The GSA did not completely define the Pareto domain. Further analysis showed that a comparison of data points generated using the GSA method with those generated using the SPEA method resulted in some of the domain points generated using the GSA becoming dominated points because of the step size of the grid. The division of the two Pareto domains into zones according to their ranking score, assigned using Net Flow, showed that each method identified one of the domains as having only low-ranked points, while the other as the location of the high- and mid- ranked domain points. Table 5 presents the results of the optimal value identified using the preferences of the expert. Each method identified different optimal points, which were all located in the same region of the Pareto domain. In this case study the optimal points are located on the boundary of one of the inputs, namely X_1.

Table 5 - Optimal values found using SPEA, DPEA, and GSA for Test Case 3

Algorithm	X_1	X_2	$F_1(X_1,X_2)$	$F_2(X_1,X_2)$
SPEA	-3.1416	0.4009	3.0590	1.9824
DPEA	-3.1413	0.3656	3.2596	1.8848
GSA	-3.1416	0.5925	2.2890	2.5562

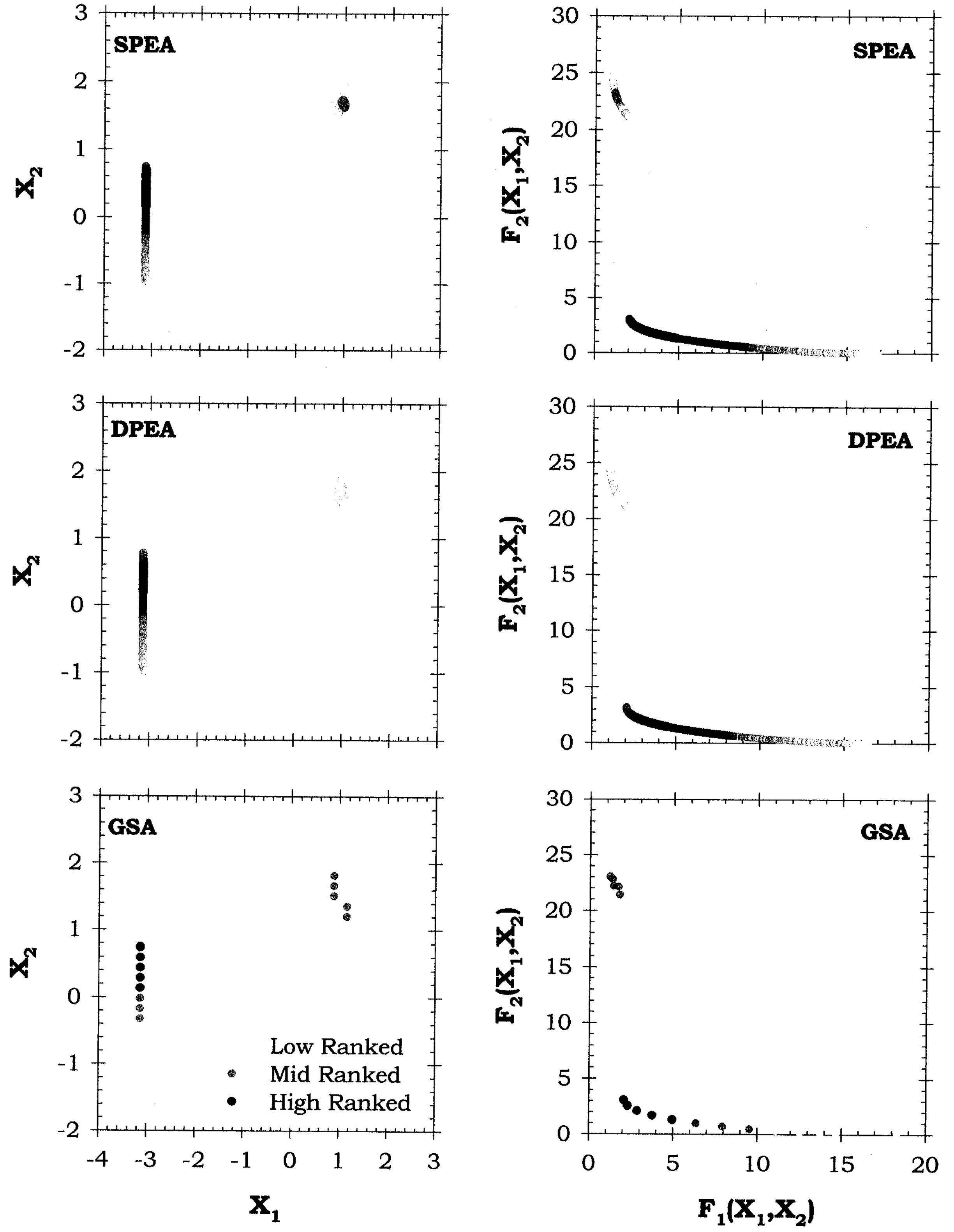

Figure 7 – Comparison of decision spaces and Pareto domains using SPEA, DPEA and GSA for Test Case Model 3.

3.2 Gluconic Acid Production

The performance of SPEA, DPEA and GSA were further investigated by applying them to the problem of determining optimum operating conditions for the production of gluconic acid. The Net Flow parameters specified by the expert are provided in Table 6. This multi-input, multi-output optimization problem consisted of four inputs and three outputs. Criteria 1, 2 and 3 represent p_f/t_B, S_o and P_f, respectively.

Table 6 - Net Flow parameter values used for classifying the Pareto domain for the production of gluconic acid production

Criterion	NFM Parameters			
k	W_k	Q_k	P_k	V_k
1	0.4	0.1	0.3	0.6
2	0.2	0.01	0.1	0.2
3	0.4	0.5	1.0	1.5

The objective criteria for this simulation were to maximize the production of gluconic acid and productivity while minimizing the final substrate concentration. The resulting Pareto domain can be successfully represented in two dimensions since the final substrate for the domain points was approximately zero. During the simulations using the SPEA and the DPEA, 6000 non-dominated points were used to represent the Pareto domain. Figures 9 and 10 present the Pareto domains and decision spaces, respectively. The results show that both evolutionary algorithms were capable of locating the Pareto domain, while GSA method could only approximate its location but did not converge to the true Pareto front. However, the GSA was capable of locating the optimum operating conditions for the initial biomass concentration and the initial substrate concentration, which were 1 UOD/mL and 50 g/L respectively. These values were also located using the SPEA and the DPEA methods. The results of each method showed that the batch time, t_B, and the oxygen mass transfer coefficient, K_La varied along their pre-determined range of variation. In the case of the evolutionary algorithms, the SPEA method analyzed 89619 data points in order to generate 6000 non-dominated points, while the DPEA method analyzed 56920 data points. The time taken for the SPEA and the DPEA to approximate the Pareto domain was 65 minutes and 45 minutes, respectively. The GSA method took 65 minutes to generate the Pareto domain locating only 66 non-dominated points after analyzing 59895 data points. However, many points in the Pareto domain generated using GSA became dominated when re-analyzed using either SPEA or DPEA.

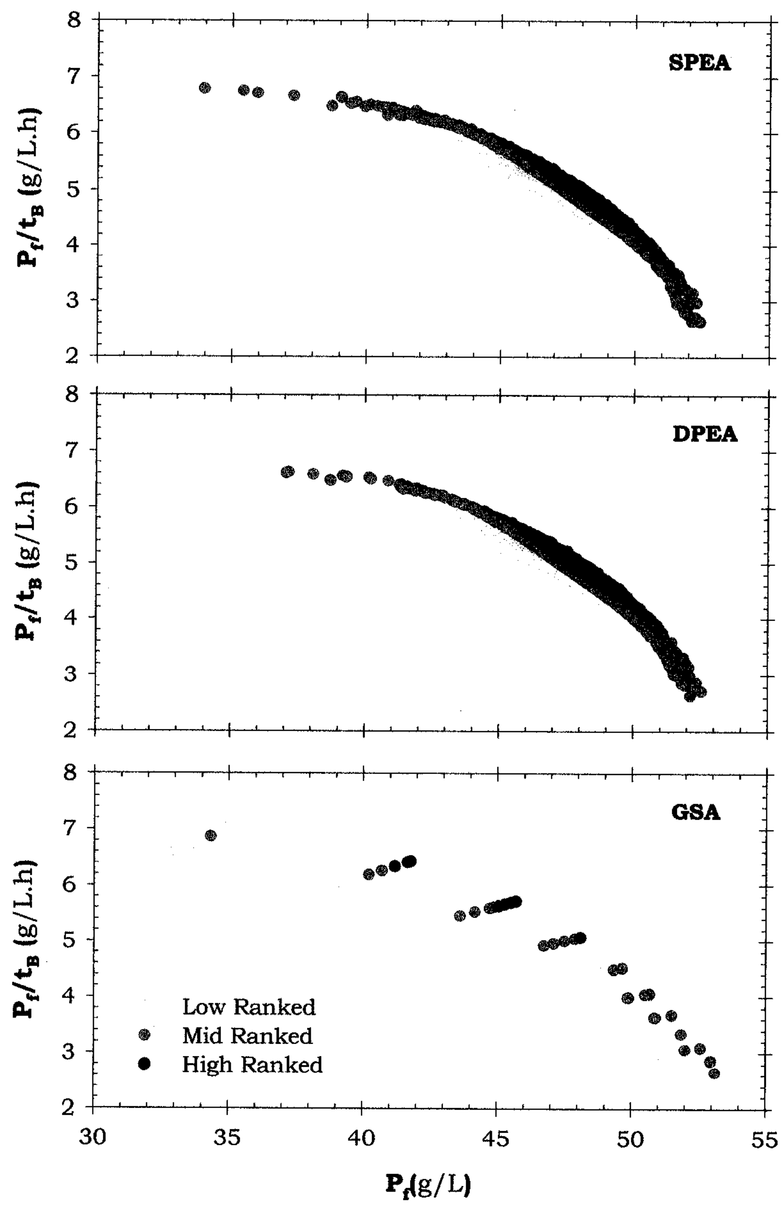

Figure 8 – A comparison of the Pareto domains generated using SPEA, DPEA and GSA methods for the production of gluconic acid.

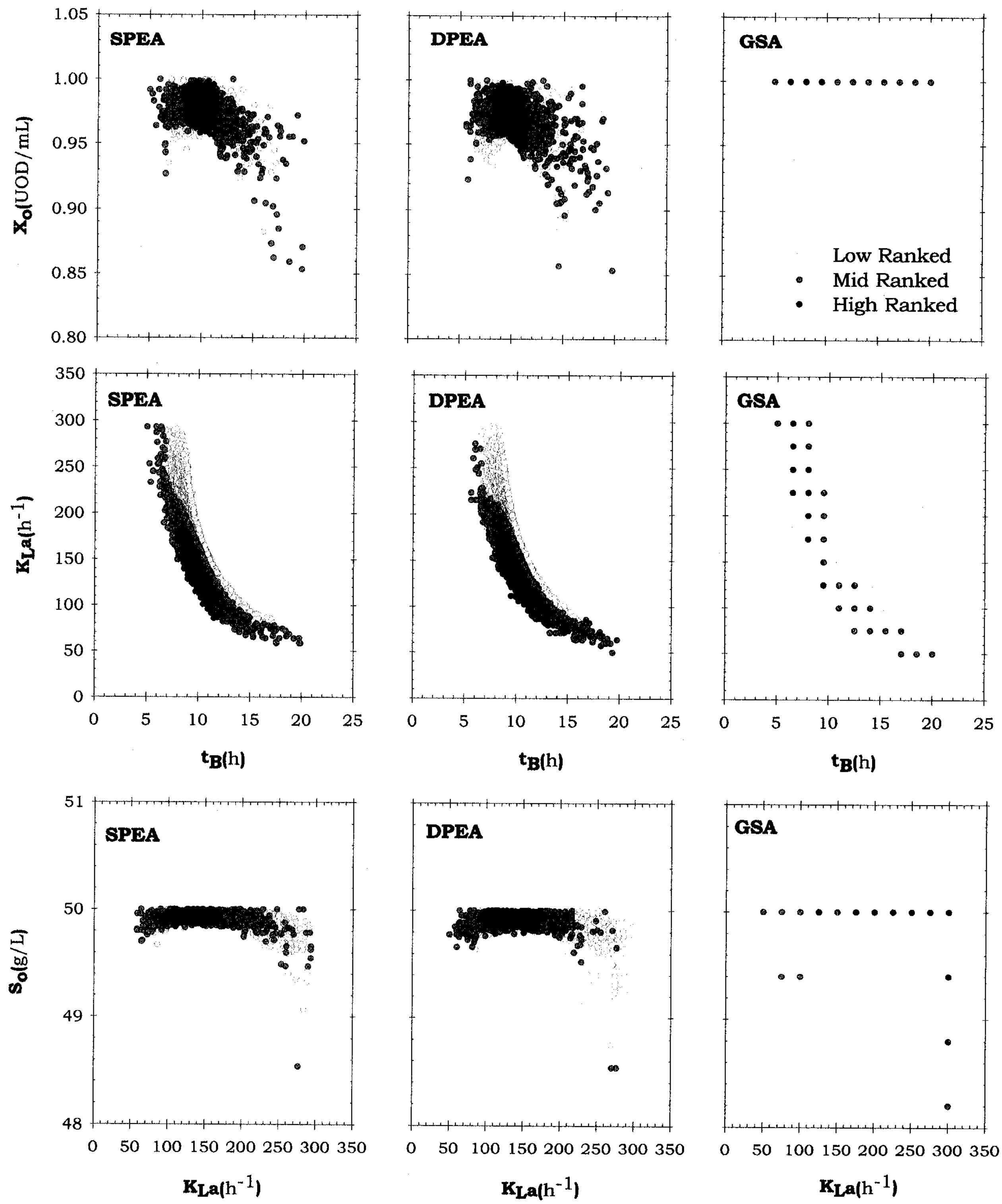

Figure 9 - A comparison of the decision spaces generated using SPEA, DPEA and GSA methods for the production of gluconic acid.

Again in this case, the refinement of the search grid in the GSA, and as a result the number of points used to approximate the Pareto domain, prevented it from converging to the Pareto domain. When a Pareto domain is associated with input variables that span over large ranges of values, it is difficult for the GSA to approximate adequately the Pareto domain unless a very large number of points are used, which is often impossible from the computational and memory point of views.

3.3 PI Controller Design

The techniques investigated in this work for generating the Pareto domain were applied to the problem of determining optimum tuning parameters for a PI controller. PI controller tuning parameters for a first-order-plus-dead-time process was determined by simulating a closed loop feedback control system. During the simulation a unit step change was made to the set point and the ITAE, ISDU, and settling time performance measures were calculated as the controlled and manipulated variables moved to their new steady state values. This was repeated for each data point analyzed during the generation of the Pareto domain. The number of non-dominated points specified to fully span the Pareto domain was 15000, in order to obtain a high level of accuracy for the optimal tuning parameters. Table 7 presents the values for the Net Flow parameters specified by the expert. Criteria 1, 2 and 3 represent the ITAE, ISDU and settling time performance measures, respectively.

Table 7 Net Flow parameter values used for classifying the Pareto domain for determining optimal tuning parameters for a PI controller.

Criterion	**NFM Parameters**			
k	W_k	Q_k	P_k	V_k
1	0.35	5.0	10.0	15.0
2	0.35	0.01	0.05	0.1
3	0.3	1.0	3.0	6.0

Figures 10 and 11 present the results of the decision spaces and Pareto domains generated using the SPEA, the DPEA and the GSA, respectively.

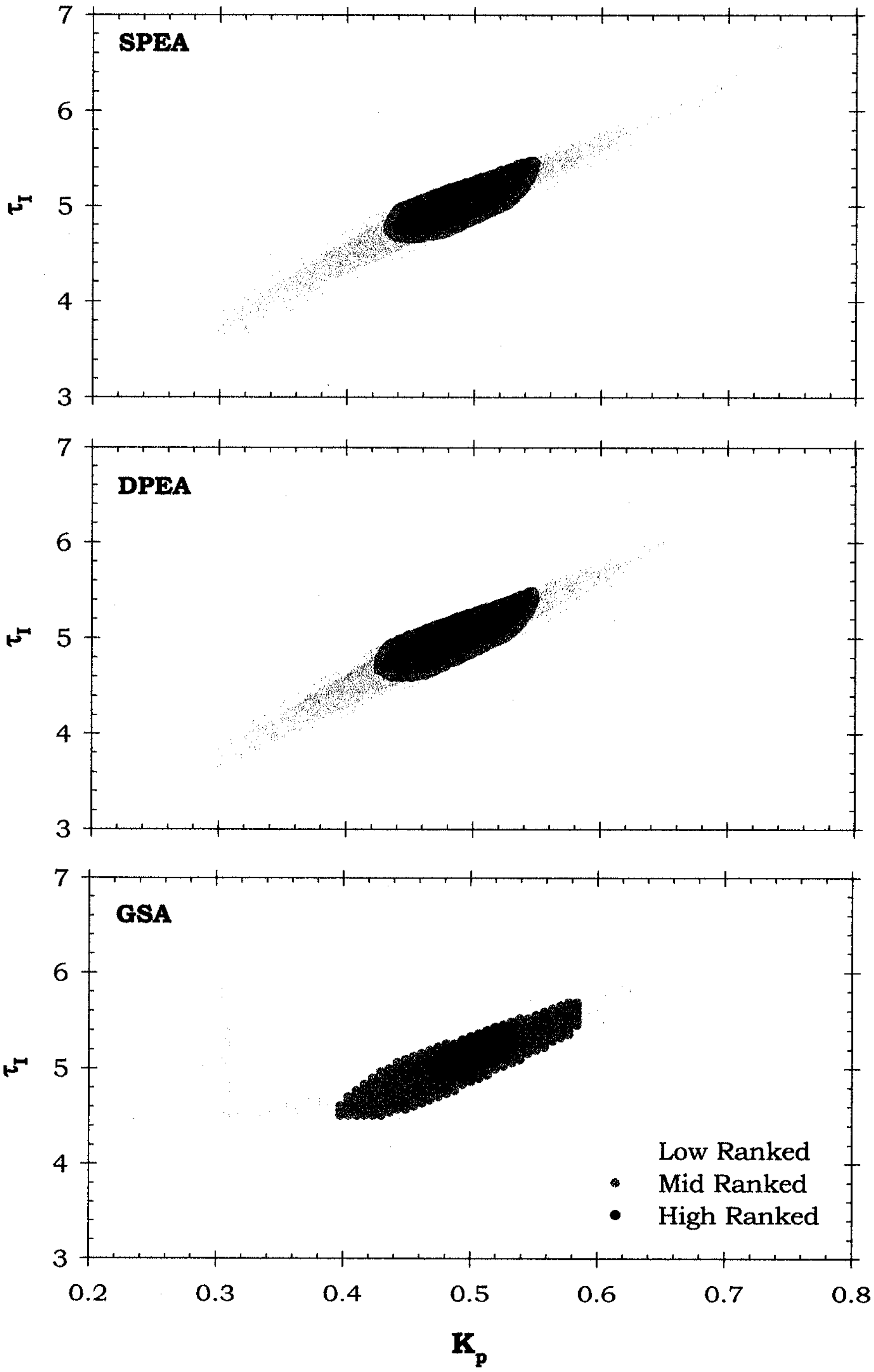

Figure 10 - A comparison of the decision space generated using SPEA, DPEA and GSA for determining optimal tuning parameters for a PI controller.

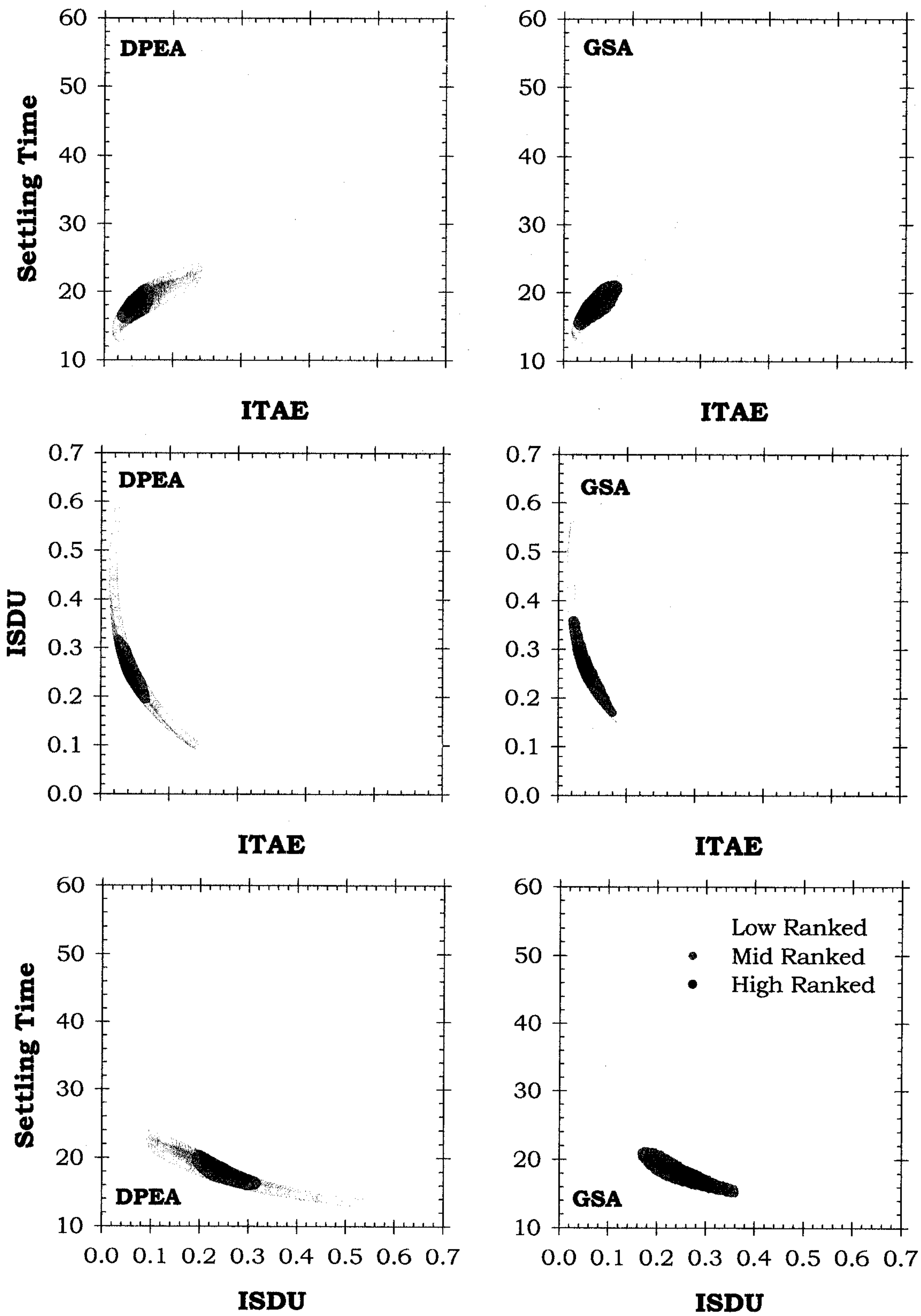

Figure 11 - A comparison of the PI controller Pareto domains generated using DPEA and GSA. The Pareto domains generated by SPEA was similar to those generated using DPEA.

The results show that the GSA was capable of converging to the true Pareto domain, while the SPEA and the DPEA converged to specific sections of the true Pareto domain. In terms of the evolutionary algorithms studied, the SPEA method analyzed 58873 data points, and the DPEA analyzed 55867 data points in 63 and 61 minutes respectively, in order to obtain 15000 non-dominated points. The GSA took 2.5 minutes to identify the boundary of the decision space and generate the Pareto domain. It retained 651 non-dominated points to describe the domain. During the optimization it analyzed 13723 data points.

Further comparison of the results show that the domain points which were not identified by the SPEA and DPEA were classified as low-ranked points in the domain generated using GSA. Each method was capable of locating the same optimal operating region identified as the high ranked points in the domain. The tuning parameters identified by the optimal point generated using the SPEA, the DPEA and the GSA, respectively, are presented in Table 8.

Table 8- The tuning parameters identified by the optimal point generated using the SPEA, the DPEA and the GSA.

Algorithm	K_p	τ_I	ITAE	ISDU	Settling Time
SPEA	0.40	5.1	69	0.25	18.1
DPEA	0.48	5.1	70	0.24	18.1
GSA	0.50	5.2	62	0.27	17.5

4 Conclusions

The objective of this paper was to investigate three techniques for generating the Pareto domain, namely, the Single Population Evolutionary Algorithm (SPEA), the Dual Population Evolutionary Algorithm (DPEA) and the Grid Search Approach (GSA) in order to develop a robust multi-objective optimization strategy.

In this investigation, a total of twelve cases were studied (Halsall-Whitney, 2004) but only three are reported in this paper. It was found that, for all test cases, the two evolutionary algorithms were fairly robust to approximate adequately the Pareto domain. On the other hand, the GSA was found to be very efficient for optimisation problems of low dimensionality of the decision space. When the number of input variables increases beyond three, the number of points that needs to be generated rapidly becomes computationally infeasible, especially when a fine grid is required. In general, the GSA was found to be more capricious compared to the other two algorithms. On the other hand, the SPEA and the DPEA were consistently able to adequately approximate to the true Pareto domain regardless of the number of inputs or outputs used to define the multi-objective optimization problem. However, in all cases the DPEA method was more consistent in identifying the true Pareto domain than the SPEA in terms of the time required to generate the desired number of non-dominated points, and the number of points analyzed during the simulation.

Therefore, the recommended optimization strategy is to use the Dual Population Evolutionary Algorithm to generate the Pareto domain and to classify it using Net Flow, which incorporates the preferences of the expert into the optimization strategy.

References

Brans, J. P., Mareschal, B., and Vincke, P. H., PROMETHEE: A New Family of Outranking Methods in Multicriteria Decision Making, *Operations Research*, Vol. 84, pgs. 477-498, 1984.

Deb, K., Multi-objective Optimization Using Evolutionary Algorithms, Wiley, NY, 2001.

Derot B., Gareau J., Kiss L. N., and Martel J. M., The Solver of Volvox Multicriteria Table. Multicriteria Analysis, Climaio J., (ed) Springer-Verlag, pgs. 113-126, 1997.

Doumpos, M., and Zopounidis, C., Multicriteria Decision Aid Classification Methods, *Applied Optimization*, Vol. 73, Kluwer Academic, 2002.

Edgar T. F., Himmelblau D. M., and Lasdon L. S., Optimization of Chemical Processes. 2nd ed., McGraw-Hill, 2001.

Evangelos, T., Multi-criteria Decision Making Methods: A Comparative Study, *Applied Optimization*, Vol. 44, Kluwer Academic, 2000.

Fonseca, C. M., and Fleming, P. J., On Performance Assessment and Comparison of Stoichastic Multiobjective optimizers, *Proceedings of Parallel Problem Solving from Nature IV (PPSN-IV)*, pgs. 584-59, 1996.

Gergely, S., Bekassy-Molnar, E., and Vatai, G., The Use of Multi-objective Optimization in Improved Wine Filtration, *Journal of Food Engineering*, Vol. 58, pgs 311-316, 2003.

Johansen T. A., and Foss, B. A., Semi-Empirical Modelling of Non-Linear Dynamic Systems Through Identification of Operating Regimes and Locals Models, In Hunt, K., Irwin, G., Warwick, K., (eds) *Neural Network Engineering in Control Systems*, Springer-Verlag, pp 105-126, 1995.

Halsall-Whitney, H., Taylor, D., and Thibault, J., Multicriteria Optimization of Gluconic Acid Production Using Net Flow, *Bioprocess and Biosystems Engineering*, Vol. 25, No. 5, pgs. 299-307, 2003.

Halsall-Whitney, H, The Development and Application of a Multi-objective Optimization Technique for Chemical Processes and Controller Design, *Masters of Applied Science in Chemical Engineering Thesis*, Chemical Engineering Department, University of Ottawa, Ottawa, Canada, 2004.

Herreros, A., Baeyens, and E., Peran, J. R., MRCD: A genetic Algorithm for Multi-objective Robust Control Design, *Engineering Applications of Artificial Intelligence*, Vol. 15, pgs. 285-301, 2002.

Marlin, T. E., Process Control: Designing Processes and Control Systems for Dynamic Performance, 2nd ed., McGraw Hill, NY, 2000.

Perrin, E., Mandrille, A., Oumoun, M., Fonteix, C., and Marc I., Optimisation Globale par Stratégie d'Évolution: Technique Utilisant la Génétique des Individus Diploïdes. *RAIRO-Recherche Opérationnelle*, Vol. 31, pgs.161-201, 1997.

Poloni, C., Giurgevich, A., Onesti, L., and Pediroda, V., Hybridization of a Multi-objective Genetic Algorithm, a Neural Network, and a Classical Optimizer for Complex Design problem in Fluid Dynamics, *Computer Methods in Applied Mechanics and Engineering*, Vol. 186, (2-4), pgs. 403-420, 2000.

Roy, B., ELECTRE III: Algorithme de Classement Basé sur une résentation Floue des Préférences en presence de Critères Multiples. Cahiers du CERO, Vol. 20, pgs.3-24, 1978.

Scarelli, A., and Narula, C. N., A Multicriteria Assignment Problem, *Journal of Multi-Criteria Decision Analysis*, Vol. 11, pgs 65-74, 2002.

Silva, C. M., and Biscaia, E. C., Genetic Algorithm Development for Multi-objective Optimization of Batch Free-radical Polymerization Reactors, *Computers and Chemical Engineering*, Vol 27, pgs 1329-1344, 2003.

Silva, C. M., and Biscaia, E. C., Genetic Algorithm Development for Multi-objective Optimization of Batch Free-radical Polymerization Reactors, *Computers and Chemical Engineering*, Vol 27, pgs 1329-1344, 2003.

Thibault, J., Taylor, and D., Fonteix, C., Multicriteria Optimization for the Production of Gluconic Acid, *Proceedings of the 8th International Conference on Computer Applications in Biotechnology: Modeling and Control of Biological Processes*, IFAC, Quebec, pgs. 24-27 2001.

Viennet, R., Fonteix, C., and Marc, I., Multicriteria Optimization Using a Genetic Algorithm for Determining a Pareto Set, *International Journal of Systems Science*, Vol. 27, No. 2, pgs. 255-260, 1995.

Chapter 3

Paper 2

Multicriteria Optimization of Gluconic Acid Production Using Net Flow

Hayley Halsall-Whitney, David Taylor, and Jules Thibault

Department of Chemical Engineering
University of Ottawa, Ottawa, Ontario,
Canada K1N 6N5
Tel: (613) 562-5920; Fax: (613) 562-5172
thibault@genie.uottawa.ca and taylor@genie.uottawa.ca

Journal Publication: Bioprocess and Biosystems Engineering, Volume 25, Number 5
Publisher: Springer-Verlag
Received: 26th August 2002
Accepted: 20th November 2002
Published Online: 6th February 2003
Published in Print: March 2003

Abstract

The biochemical process industry is often confronted with the challenge of making decisions in an atmosphere of multiple and conflicting objectives. Recent innovations in the field of operations research and systems science have yielded rigorous multicriteria optimization techniques that can be successfully applied to the field of biochemical engineering. These techniques incorporate the expert's experience into the optimization routine and provide valuable information about the zone of possible solutions.

This paper presents a multicriteria optimization strategy that generates a Pareto domain, given a set of conflicting objective criteria, and determines the optimal operating region for the production of gluconic acid using the Net Flow method. The objective criteria include maximizing the productivity and concentration of gluconic acid, while minimizing the residual substrate.

Three optimization strategies are considered. The first two strategies identify the optimal operating region for the process inputs. The results yielded an acceptable compromise between productivity, gluconic acid concentration, and residual substrate concentration. Fixing the process inputs representing the batch time, initial substrate concentration and initial biomass equal to their optimal values, the remaining simulations were used to study the sensitivity of the optimum operating region to changes in the oxygen mass transfer coefficient, K_La, by utilizing a multi-level K_La strategy. The results show that controlling K_La during the reaction reduced the production of biomass, which in turn resulted in increased productivity and concentration of gluconic acid above that of a fixed K_La.

Keywords: Multicriteria Optimization, Pareto domain, Net Flow, Gluconic Acid Production

Nomenclature

C	Dissolved oxygen concentration in the broth, $g.L^{-1}$ *or* Global concordance index
c_k	Individual concordance index
C^*	Concentration of oxygen in liquid in equilibrium with gas phase, $g.L^{-1}$
D_k	Discordance index,
K_La	Volumetric oxygen transfer coefficient, h^{-1}
k_l	Michaelis constant for lactone production, $g.L^{-1}$
k_p	Gluconolactone hydrolysis rate constant, h^{-1}
k_o	Monod rate constant of growth with respect to oxygen, $g.L^{-1}$
k_s	Monod rate constant of growth with respect to glucose, $g.L^{-1}$
l	Gluconolactone concentration, $g.L^{-1}$
p	Gluconic acid concentration, $g.L^{-1}$
p_f	Final Gluconic acid concentration, $g.L^{-1}$
p_f/t_B	Gluconic acid productivity, $g.L^{-1}.h^{-1}$
P_k	Preference threshold
P/V	Power supply for agitation in the reactor, $W.m^{-3}$
Q_k	Indifference threshold
S	Substrate concentration, $g.L^{-1}$
S_o	Initial substrate concentration, $g.L^{-1}$
t_B	Batch time, h
t	Time, h
U_g	Superficial gas velocity of air entering the reactor, $m.s^{-1}$
V_k	Veto threshold
v_l	Velocity constant for lactone production, $mg.UOD^{-1}.h^{-1}$
W_k	Weight, dimensionless
X	Cell concentration, $UOD.ml^{-1}$
X_o	Initial cell concentration, $UOD.ml^{-1}$
Y_o	Yield of growth based on oxygen, $UOD.mg^{-1}$
Y_s	Yield of growth based on glucose, $UOD.mg^{-1}$

Greek Characters

μ_m Maximum specific growth rate, h^{-1}

1 Introduction

In complex processes, determining a suitable set of operating conditions that would provide an optimal set of results is a major challenge, primarily due to the existence of multiple and often conflicting objectives. In a fermentation process, for instance, it is desirable to minimize production costs, the time of fermentation, waste production and energy consumption while maximizing productivity and yield. The solution to this dilemma lies in determining a compromise between all or some of the objectives in order to achieve an acceptable overall result, given the various constraints of the process.

Historically, optimization techniques have dealt with multiple objectives by combining them into one objective function composed of the weighted sum of individual objectives, or by transforming one objective into a single response function while using the others as constraints (Edgar *et al.*, 2001)]. These techniques are based on the assumption that the objective function and the constraints are well behaved, and that there exists some optimal point that will resolve all multiple and conflicting objectives. There are several drawbacks to these approaches. First, combining multiple objectives into a single objective function does not provide information about trade-offs amongst the various objectives, or about alternative operating conditions. Second, an optimization technique that constrains some objectives may bias the final solution, which can result in misleading conclusions. Third, techniques that provide only one optimal operating solution to a given process, rather than an optimal operating region, do not take into account the difficulty associated with controlling the process. Fourth, traditional optimization methodologies do not incorporate the practical experience of the expert in terms of his or her knowledge of the overall behavior of the process.

In recent years, new methodologies have been developed to simultaneously optimize multiple and conflicting objectives. The decision-maker's knowledge about the relationships that exist between the various objective criteria is considered a tool for selecting an optimal solution. A general class of multicriteria algorithms exists that generates a domain of possible solutions (Pareto domain) and then optimizes it using methods that incorporate the preferences expressed by the expert (Thibault *et al*, 2001; Ringuest, 1992)

This paper examines the robustness of one of these techniques, the Net Flow Method, in determining optimum operating conditions for the production of gluconic acid and for studying the sensitivity of this optimal region to changes in the overall mass transfer coefficient, $K_L a$.

2 Theory

2.1 Gluconic Acid Production Model

This investigation simulates the fermentation of glucose to gluconic acid by the micro-organism *Pseudomonas ovalis* in a batch stirred tank reactor. Studies of this reaction concerned with the effect of $K_L a$ on productivity have reported that, at steady state, the rate of oxygen absorption is proportional to the rate of production of gluconic acid without the interference of side reactions (Bull and Kempe, 1970) The overall mechanism can be expressed as follows (Johansen and Foss, 1995):

Cells + Glucose + Oxygen $\longrightarrow$ More cells

Glucose + Oxygen $\longrightarrow$ Gluconolactone

Gluconolactone + Water $\longrightarrow$ Gluconic Acid

The following state-space model has been derived to represent, respectively, the concentration of cells (X), gluconic acid (p), gluconolactone (l), glucose substrate (S), and dissolved oxygen (C) (Ghose and Gosh, 1976):

$$\frac{dX}{dt} = \mu_m \frac{SC}{k_s C + k_o S + SC} X \quad (1)$$

$$\frac{dp}{dt} = k_p l \quad (2)$$

$$\frac{dl}{dt} = v_l \frac{S}{k_l + S} X - 0.91 k_p l \quad (3)$$

$$\frac{dS}{dt} = -\frac{1}{Y_s}\mu_m \frac{SC}{k_sC+k_oS+SC}X - 1.011v_l\frac{S}{k_l+S}X \quad (4)$$

$$\frac{dC}{dt} = K_La\left(C^* - C\right) - \frac{1}{Y_o}\mu_m \frac{SC}{k_sC + k_oS + SC}X - 0.09v_l\frac{S}{k_l+S}X \quad (5)$$

In this study the overall oxygen mass transfer coefficient, K_La, was expressed in terms of the amount of power consumed during the supply and dispersal of oxygen during the reaction. The general form of this correlation is as follows (Bulock and Kristiansen, 1987)

$$K_La = \xi\left(\frac{P}{V}\right)^{\alpha}\left(U_g\right)^{\beta} \quad (6)$$

In other words, K_La is represented as a function of the power input per unit volume (P/V) for agitation and the superficial gas velocity (U_g). This equation has been used to determine K_La in reciprocating plate reactors, bubble columns, air-lift systems, and stirred tank reactors (Gagnon *et al*, 1998; Trevan *et al.*, 1987). Table 1 lists the values of the coefficients used in these equations to model the production of gluconic acid.

Table 1 - Gluconic acid production model parameter values

Parameter	Value	Unit
μ_m	0.39	h^{-1}
k_s	2.50	g/L
k_o	0.00055	g/L
k_p	0.65	h^{-1}
v_l	8.30	*mg/UOD.h*
k_l	12.80	g/L
Y_s	0.38	*UOD/mg*
Y_o	0.89	*UOD/mg*
C^*	0.00685	g/L
ζ	30.0	*dimensionless*
β	0.6	*dimensionless*
α	0.5	*dimensionless*

2.2 Optimization Objectives

While numerous objective criteria could be identified for optimizing the production of gluconic acid, this work focused on maximizing the overall production rate (p_f/t_B) and the final concentration (p_f) of gluconic acid, while minimizing the final substrate concentration (S_f) at the end of the fermentation process.

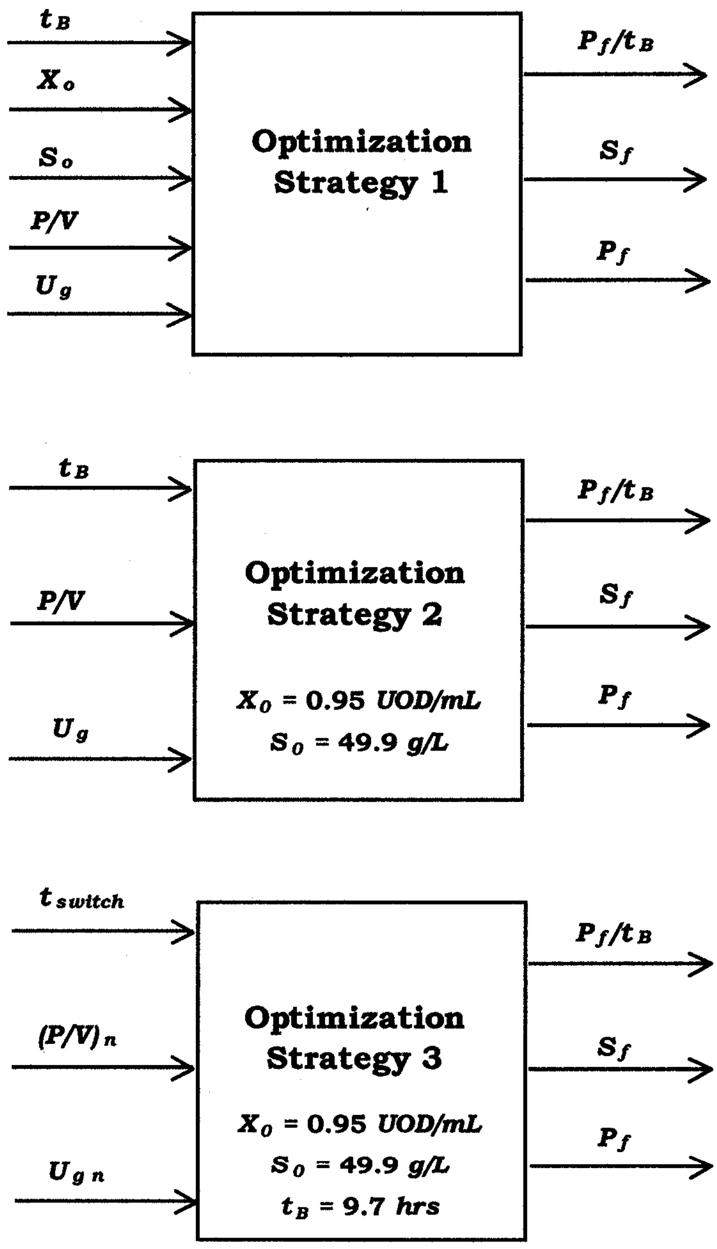

Figure 1 Objectives used during the development of the Pareto domain and optimization of gluconic acid production using Net Flow

Figure 1 presents the three multi-input, multi-output optimization strategies used in this investigation to determine optimum initial inputs for the process, and to study $K_L a$ requirements during the reaction. The initial inputs are the duration of the fermentation process ($t_B \in$ [5-15 *h*]), the initial substrate concentration ($S_0 \in$ [20-50 *g/L*]), the initial biomass concentration ($X_0 \in$ [0.05-1.0 *UOD/mL*]), the power requirement per unit volume for agitation ($P/V \in$ [345-2155 W/m^3]), the superficial gas velocity ($Ug \in$ [0.0025-0.01 *m/s*]), and for the third optimization strategy a switch time ($t_{switch} \in [0\text{-}\ t_B]$) (Thibault *et al.*, 2001).

2.3 Optimization Strategy

The optimization procedure consisted primarily of two steps: (i) the generation of a large number of feasible solutions that approximates the domain of all possible candidates for the optimum, known as the Pareto domain, and (ii) the evaluation of these possible candidate solutions by ranking them using the Net Flow method, which incorporated *a priori* process knowledge supplied by the expert.

2.3.1 Approximating the Pareto domain

The Pareto domain represents a set of operating conditions yielding solutions that may serve as the optimum, depending on the relative importance placed on each of the objectives. These objective criteria, meanwhile, are provided by the expert. To establish this domain the following procedure was used (Thibault *et al.*, 2001)

1. For each input variable, a value was randomly selected from within its predefined range of variation, using a random number generator. The model was then solved for this set of inputs in order to generate values for the objective criteria. Initially, *M* points, specified by the user, were generated to fully span the decision space using a pseudo-

random number generator, which was altered so that the boundaries of the decision space would be taken into consideration.

2. All M points were compared two by two, based on the values for their objective criteria, in order to determine the number of times a point had been dominated by another in the current output space. Each time a point was dominated, its domination score was incremented by one; otherwise it remained unchanged from an initial score of zero. At the end of the comparison process, the data set was sorted in order of domination scores, starting from data points that were never dominated (i.e. having a score of 0) and ending with the data point displaying the highest frequency of domination.
3. All non-dominated data points, N_0, were then selected, along with a fraction of the dominated points having the lowest frequency of domination, to be used in generating the next approximation to the Pareto Domain. This selection process can be described by the following equation:

$$N = N_0 + INT\left[(F_s)(M - N_0)\right] \qquad (7)$$

In this equation, F_S is the survival fraction of the dominated data points and $INT(x)$ yields the integer value of x. In this study a survival fraction of 0.3 was used. This ensures that only the better elements of the population survived to participate in the creation of the new generation of possible solutions.

4. A total of $(M\text{-}N)$ new data points were then generated to replace those points that were eliminated during the selection process. Combining these new points with those points selected from the previous set yielded a new set of M data points. Each new point, $I_{p,k}$, was generated by randomly selecting one non-dominated point, $I_{p,i}$, and one dominated point, $I_{p,j}$, from the set of points retained in the previous generation. The new set of inputs for the new point was then determined according to the following equation.

$$I_{p,k}^n = D_p I_{p,i}^n + (1 - D_p) I_{p,j}^n \qquad (8)$$

In this equation, D_p is a randomly selected number between 0 and 1. A new value for D_p was selected each time a new input, in the vector of inputs representing $I_{p,k}$, was determined.

5. Steps (1)-(4) were repeated until M non-dominated data points, which was specified by the user as the number of points required to adequately define the Pareto domain, were generated. By eliminating in successive generations all data points that were not Pareto-efficient, it was assumed that this final set of M possible solutions approximates the Pareto domain.

The search for a judicious compromise solution, subject to conflicting objectives, could now be performed on a significantly reduced, unbiased solution space using a multicriteria optimization algorithm.

2.3.2 Multicriteria Optimization Algorithm

The Pareto domain was established from a domination perspective without the bias of *a priori* knowledge concerning the relative importance of the various criteria. However, to classify each point in the domain the expert needs to incorporate his/her knowledge of the process into the optimization routine. The Net Flow method (NFM), which developed as a result of modifications made to the ELECTRE III method (Roy, B, 1978; Brans *et al.*, 1984; Derot *et al.*, 1997), was used here to optimize the Pareto domain.

In the NFM, *a priori* knowledge of the process, expressed by the decision maker, is incorporated into the optimization routine using four sets of parameters to rank all the data points of the Pareto domain. These parameters are described as follows:

1. The first parameter gives the relative importance of each criterion, expressed as a weight W_k. In this algorithm, the weights are normalized:

$$\sum_{k=1}^{n} W_k = 1 \quad \text{......(9)}$$

2. The second parameter refers to the indifference threshold (Q_k), which defines the range of variation of each criterion for which it is not possible for the decision-maker to favor the criterion of one point over the corresponding criterion of another.
3. The third parameter refers to the preference threshold (P_k). If the difference between two values for a given criterion exceeds this threshold, a preference was given to the better criterion. If the objective was to maximize the criterion then the better point was the point with the larger value.
4. The fourth parameter refers to the veto threshold (V_k), which serves to ban a point relative to the other if the difference between the values of a criterion was too high to be tolerated. A point was banned if the veto threshold was violated for at least one of the criteria even if the other criteria were acceptable.

These three thresholds are defined for each criterion such that the following relationship holds:

$$0 \le Q_k \le P_k \le V_k \quad \text{......(10)}$$

They represent a reference range established by the decision-maker to assess the values of the objective criteria for each alternative in the Pareto domain (Roy, 1978). The NFM algorithm is described as follows.

1. First, for each combination of points in the Pareto domain, the difference between the values F_k of each criterion k was calculated by comparing point *i* with point *j* using the following equation.

$$\Delta_k[i, j] = F_k(i) - F_k(j) \begin{cases} i \in [1, M] \\ j \in [1, M], j \neq i \\ k \in [1, n] \end{cases} \quad \text{......(11)}$$

In subsequent equations, minimizing a criterion required using $\Delta_k[i,j]$, while maximizing a criterion required using $-\Delta_k[i,j]$.

2. Using the values of $\Delta_k[i,j]$, the concordance index $c_k[i,j]$ for each criterion was determined for all n criteria and for each pair of data points using the following relationships.

$$c_k[i,j] = \begin{cases} 1 & \text{if } \Delta_k[i,j] \leq Q_k \\ \dfrac{P_k - \Delta_k[i,j]}{P_k - Q_k} & \text{if } Q_k < \Delta_k[i,j] \leq P_k \\ 0 & \text{if } \Delta_k[i,j] > P_k \end{cases} \qquad (12)$$

The concordance index measures the strength of the argument that when comparing point i to point j for a given criterion k the value of '*F_k (i) is at least as good as F_k (j)*' when compared to values specified by the decision-maker in the reference range for a given criteria (Roy, 1978). Figure 2a illustrates how the concordance index was determined using the values of the calculated differences, the indifference threshold, and the preference threshold. For a difference smaller than the indifference threshold, the corresponding concordance index is 1. Between the indifference and preference thresholds, it varies linearly from 1 to 0. For a difference larger than the preference threshold, the concordance index was set to 0.

3. The weighted sum of individual concordance indices was calculated to determine the global concordance index.

$$C[i,j] = \sum_{k=1}^{n} W_k c_k[i,j] \quad \begin{cases} i \in [1,M] \\ j \in [1,M] \end{cases} \qquad (13)$$

4. A discordance index $D_k[i,j]$ was calculated for each criterion k using the preference and veto thresholds.

$$D_k[i,j] = \begin{cases} 0 & \text{if } \Delta_k[i,j] \le P_k \\ \dfrac{\Delta_k[i,j] - P_k}{V_k - P_k} & \text{if } P_k < \Delta_k[i,j] \le V_k \\ 1 & \text{if } \Delta_k[i,j] > V_k \end{cases} \qquad (14)$$

The discordance index measures the strength of the argument that when comparing point *i* to point *j* for a given criterion k the value of '*F_k (i) is significantly worse than F_k (j)*' when compared to values specified by the decision-maker in the reference range for a given criteria (Roy, 1978). Figure 2b illustrates how the discordance index was determined using the preference and veto thresholds. For a difference smaller than the preference threshold, the discordance index is 0. Between the preference and veto thresholds, it varies linearly from 0 to 1, and for a difference larger than the veto threshold, the concordance index is set to 1.

5. Using the concordance and discordance indices, the relative performance of each pair of domain points was evaluated by calculating each element of the outranking matrix $\sigma[i, j]$ using the following equation.

$$\sigma[i,j] = C[i,j]\left(\prod_{k=1}^{n} 1 - (D_k[i,j])^3 \right) \begin{cases} i \in [1, M] \\ j \in [1, M] \end{cases} \qquad (15)$$

Each element $\sigma[i, j]$ measures of the quality of point *i* relative to point *j* in terms of the three criteria. An element $\sigma[i, j]$ close to 0 indicates that point *j* outranks point *i*. If the value is near 1, then data point *i* may outrank point *j* or simply be located in the vicinity of point *j*.

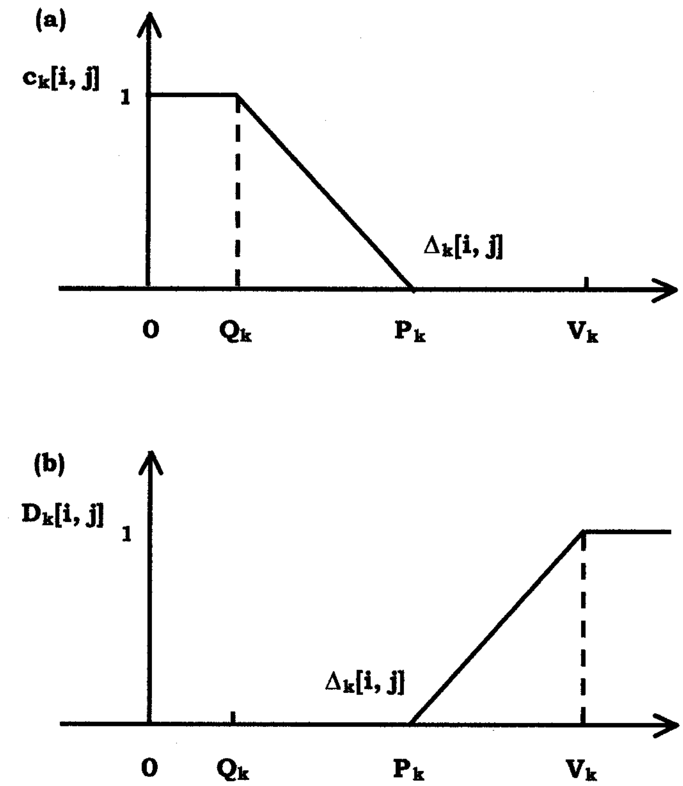

Figure 2 -(a) Concordance index, and (b) discordance index calculations used in the Net Flow algorithm to determine ranking scores for the Pareto domain points

6. The final ranking score for each data point in the Pareto domain was obtained by summing individual outranking elements associated with each domain point as follows.

$$\sigma_i = \sum_{j=1}^{M} \sigma[i, j] - \sum_{j=1}^{M} \sigma[j, i] \qquad (16)$$

The first term evaluates the extent to which element i performs relative to all the other points in the Pareto domain, while the second term evaluates the performance of all the other points relative to point i. The points were then sorted from highest to lowest according to ranking score. The data point with the highest ranking was the one that best satisfies the set of preferences provided by the decision-maker.

Finally, the results of the Net Flow method was used to divide the Pareto domain into zones containing the high-ranked, mid-ranked, and low-ranked domain points in order to identify graphically where the optimal region was located.

3 Results and Discussion

Three optimization strategies represented in Figure 1 were used to determine optimum operating conditions for the production of gluconic acid. The results of each strategy included a ranked output space, the Pareto domain, and its corresponding input space. These results were used to identify optimal operating conditions for a given strategy and to establish the degrees of freedom assumed in subsequent strategies. In each case the objective criteria involved maximizing the productivity and concentration of gluconic acid, while minimizing the residual substrate concentration.

The Pareto domain generated for each strategy consisted of 4000 to 8000 unique points; the generation of a large number of points ensured an evenly distributed representation of the domain. These points were then ranked using the Net Flow method parameters listed in Table 2. Criteria 1, 2 and 3 represent p_f/t_B, S_o and p_f, respectively.

Table 2 Net Flow method parameter values selected by the expert for optimizing the production of gluconic acid

Criterion	NFM Parameters			
k	W_k	Q_k	P_k	V_k
1	0.4	0.1	0.3	0.6
2	0.2	0.01	0.1	0.2
3	0.4	0.5	1.0	1.5

3.1 Optimization Strategy 1

The first optimization strategy was used to identify the ranked Pareto domain and its corresponding input space for the production of gluconic acid, the latter consisting of the batch time, initial biomass and substrate concentrations, the power required for agitation, and the superficial gas velocity. Figures 3 and 4 present the Pareto domain and its corresponding input space respectively, which consisted of 8000 non-dominated points. The 3-D output space of the Pareto domain (i.e. the gluconic acid productivity, final gluconic acid concentration and final substrate concentration) is adequately represented by a 2-D space because the optimization resulted in nearly complete substrate utilization in all cases. The data presented in this and subsequent figures have been divided into three sets representing the 500 top-ranked, 3000 mid-ranked and 4500 lowest-ranked points. Inspection of the Pareto domain confirms that simultaneously attempting to maximize both the productivity and concentration of gluconic acid led to conflicting objectives, since maximizing either objective could only be accomplished at the expense of the other.

The advantage of considering a ranked Pareto domain over traditional optimization techniques that yield only a single optimal solution lies in the amount of immediate information the ranked domain can provide regarding process operation. In this case, the values for the productivity, final substrate and gluconic acid concentration for points contained within the Pareto domain ranged from 2.98 $g/\ L.h$ – 6.55 $g/\ L.h$, 2.0 × 10-7 g/L – 0.417 g/L, and 35.72 g/L – 51.58 g/L respectively. Meanwhile, the results for the highest ranked point were 5.3 $g/L.h$ and 46.8 g/L for the productivity and gluconic acid concentration respectively, with almost total utilization of the substrate during reaction. The results for the corresponding input space also show that, in general, the initial biomass and substrate concentrations were confined to the upper limit of their predetermined range of variation. With respect to the highest ranked point, the values the initial biomass and substrate concentrations were 0.95 UOD/mL and 49.9 g/L respectively.

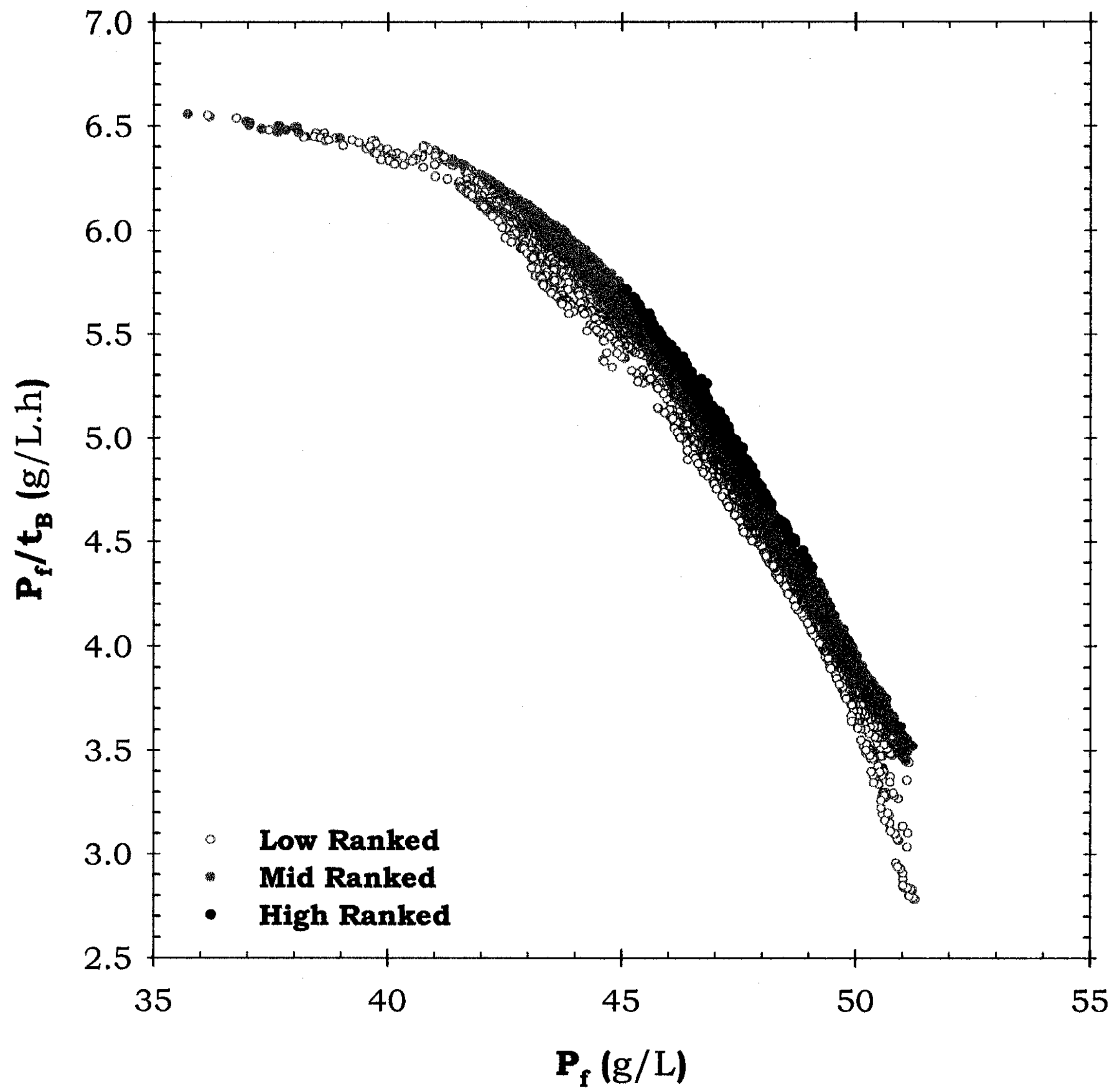

Figure 3 - Optimized Pareto domain for the production of gluconic acid.

However, the optimization did not reduce the predetermined range of variation for either the batch time or the overall mass transfer coefficient. It is also worth noting that the highest ranked points represent a compromise between the individual objectives, while the mid and low ranked points tended towards maximizing one of the objectives at the expense of the others.

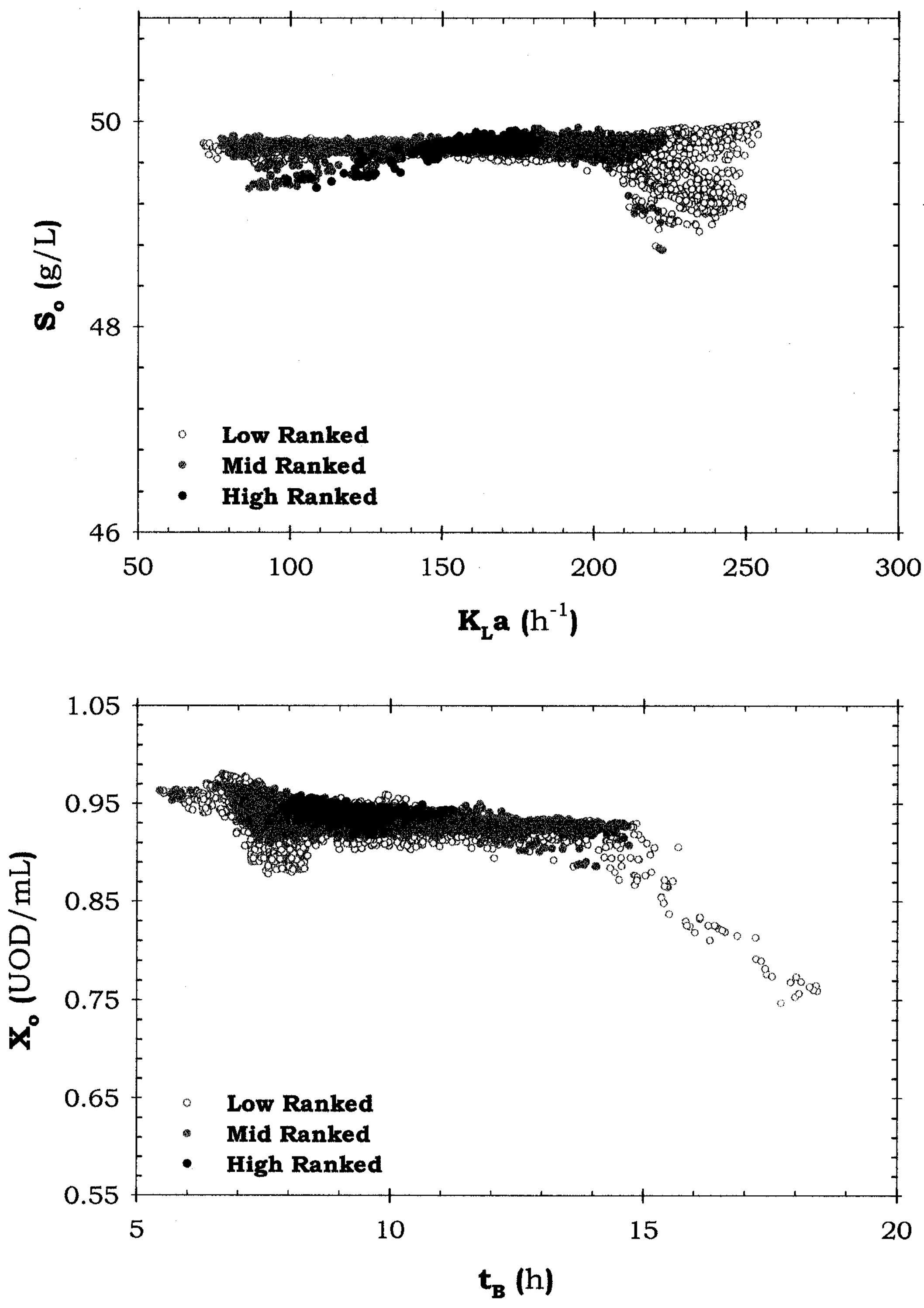

Figure 4 - Process input space corresponding to the Pareto domain for the production of gluconic acid

Figure 5 provides a closer examination of three individual cases selected from the high, mid, and low ranked region of the domain, each case having approximately the same initial biomass and substrate concentrations.

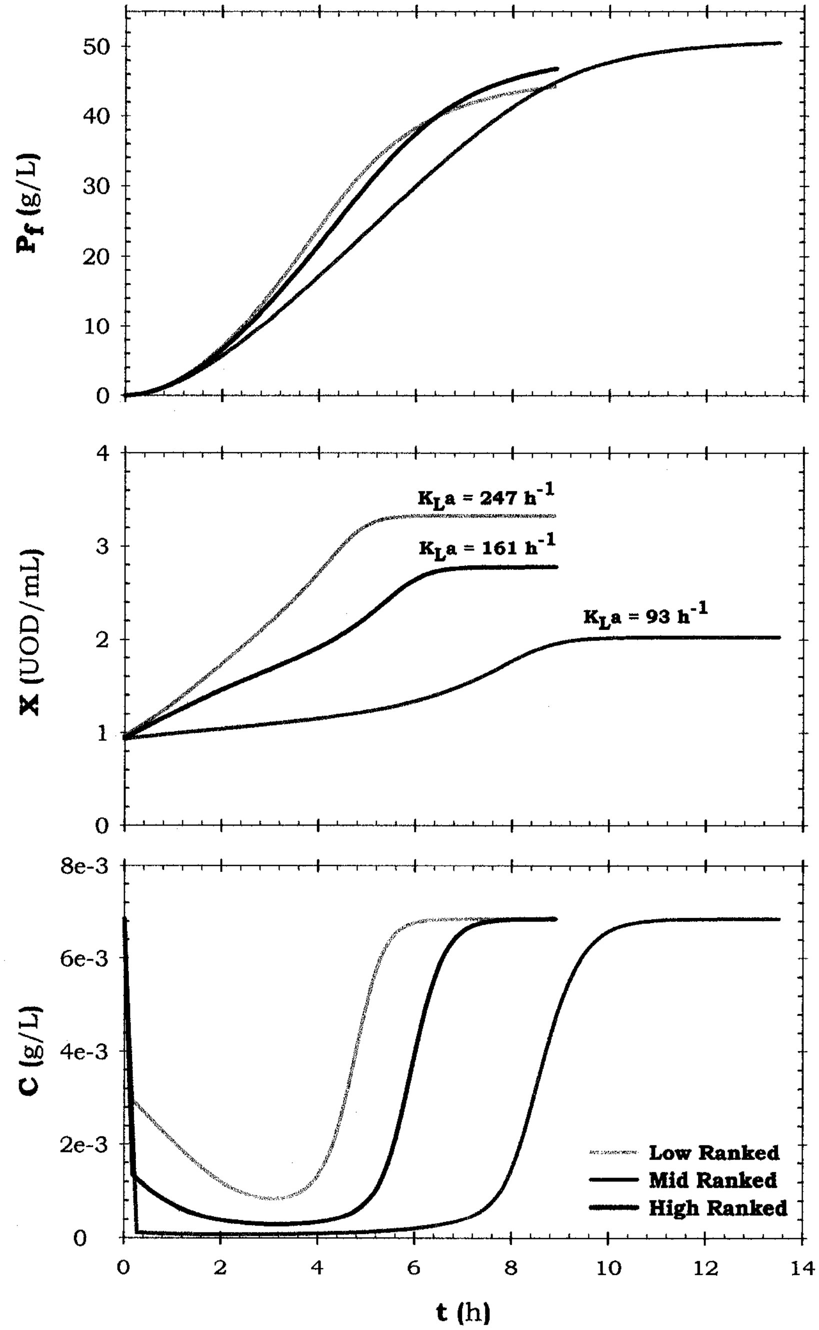

Figure 5 Gluconic acid concentration (Pf), biomass (X) and oxygen utilization rates for high-, mid- and low- ranked points selected from the optimized Pareto domain

The points were analyzed according to their biomass, overall mass transfer coefficient, oxygen utilization, and gluconic acid concentration during the reaction. In this case the productivity, final biomass concentration and gluconic acid concentration were, respectively, 2.77 g/L, 5.3 g/L.h, and 46.8 g/L for the high-ranked point, 2.02 g/L, 3.7 g/L.h, and 50.5 g/L for the mid-ranked point, and 3.33 g/L, 5 g/L.h, and 44.2 g/L for the low-ranked point respectively. Nearly complete substrate utilization was achieved in each case.

The K_La values for the high, mid, and low-ranked points selected were 161 h^{-1}, 93 h^{-1}, and 247 h^{-1} respectively. The simulations show that the high overall mass transfer coefficient attributed to the low-ranked point created an environment for producing biomass at the expense of gluconolactone production, the reaction intermediate required for the production of gluconic acid. This resulted in a lower final concentration of gluconic acid when compared to the high and mid- ranked points. The mid-ranked point had the lowest K_La value, which significantly reduced the production of biomass during the reaction and resulted in an increased production of gluconic acid. However, this increase in gluconic acid concentration required a batch time that was 4 *h* longer than that of the highest ranked point. Meanwhile, the high-ranked point represented a compromise between these two extremes. When compared to the other two simulations, the lower K_La value assumed in the high ranked case sufficiently reduced the production of biomass, which increased the production of gluconolactone used to produce gluconic acid, without increasing significantly the batch time.

3.2 Optimization Strategy 2

The results from the previous study indicate that variations in the initial biomass and substrate concentrations across the Pareto domain are minimal. It was therefore decided to refine the optimization by repeating the process, this time holding the initial substrate and biomass concentrations at the optimal values determined from the previous analysis. This effectively reduced the degrees of freedom with respect to the input space to three: the batch time, the power required for agitation, and the superficial gas velocity.

Figure 6 shows the Pareto domain, approximated here by 8000 non-dominated points, and the corresponding input space, which consisted of the power and superficial gas velocity, in terms of K_La and the batch time for this new scenario.

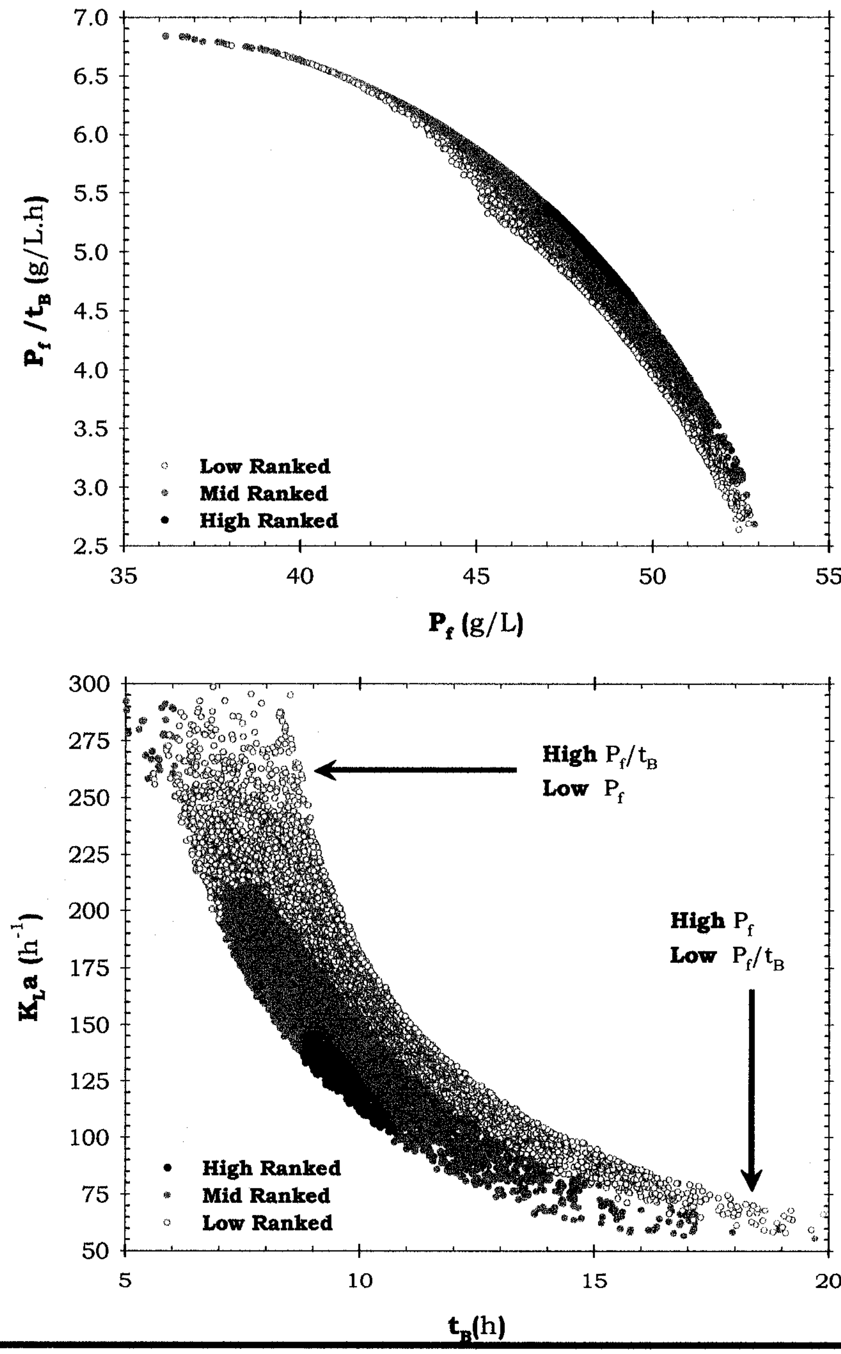

Figure 6 Required K_La values for a given batch time. Three scenarios have been identified: (a) high productivity and low gluconic acid concentration, (b) high gluconic acid concentration and low productivity, and (c) a compromise solution

The shape of the Pareto domain is similar to that generated in the previous optimization strategy, though somewhat more defined. Once again the results indicate almost total substrate utilization by the end of the reaction. Furthermore, the concentration of gluconic acid for the highest ranked point increased from 46.8 *g/L* to 48 *g/L* when compared to the previous simulation. However, this increase came at the expense of the productivity, which was reduced from 5.3 *g/L.h* to 4.9 *g/L.h*. Meanwhile, the batch time and K_La values for the optimal point in the Pareto domain were 121 h^{-1} and 9.8 *h* respectively. The corresponding input space also showed a distinct correlation between the overall mass transfer coefficient and the batch time: more specifically, the K_La requirements for operating within the Pareto domain decreased as the batch time increased. The input space also showed that shorter batch times led to productivity increases at the expense of gluconic acid production. This is because high K_La values encourage the production of biomass rather than gluconolactone, which is required for producing gluconic acid.

3.3 Optimization Strategy 3

This optimization strategy was developed to determine whether further improvements could be made with respect to productivity, final substrate concentration and gluconic acid concentration by controlling the overall mass transfer coefficient during the reaction. In particular, the simulations assumed that the K_La value could be adjusted periodically in a step fashion throughout the course of the reaction. Four additional simulations were therefore conducted, each with differing numbers of adjustable K_La values and K_La switch times, t_{switch} as process inputs. Meanwhile, the optimal X_0, S_0 and t_B of 0.95 *UOD/mL*, 49.90 *g/L*, and 9.7 *h* identified in the previous optimization strategies were used as fixed inputs in these simulations.

The first of the four simulations considered here used a two-level K_La strategy. Figure 7 presents the results of the optimization in terms of the input space, which shows that all the points in the Pareto domain retained high initial K_La values at the beginning of the fermentation process up to an average switch time of approximately 3*h*, after which the K_La values were reduced by approximately half of the original values. Furthermore, incorporating two K_La values into the optimization routine increased the productivity from 4.9 *g/L.h* to 5.1 *g/L.h* and gluconic acid production from 48 *g/L* to 49.6 *g/L*.

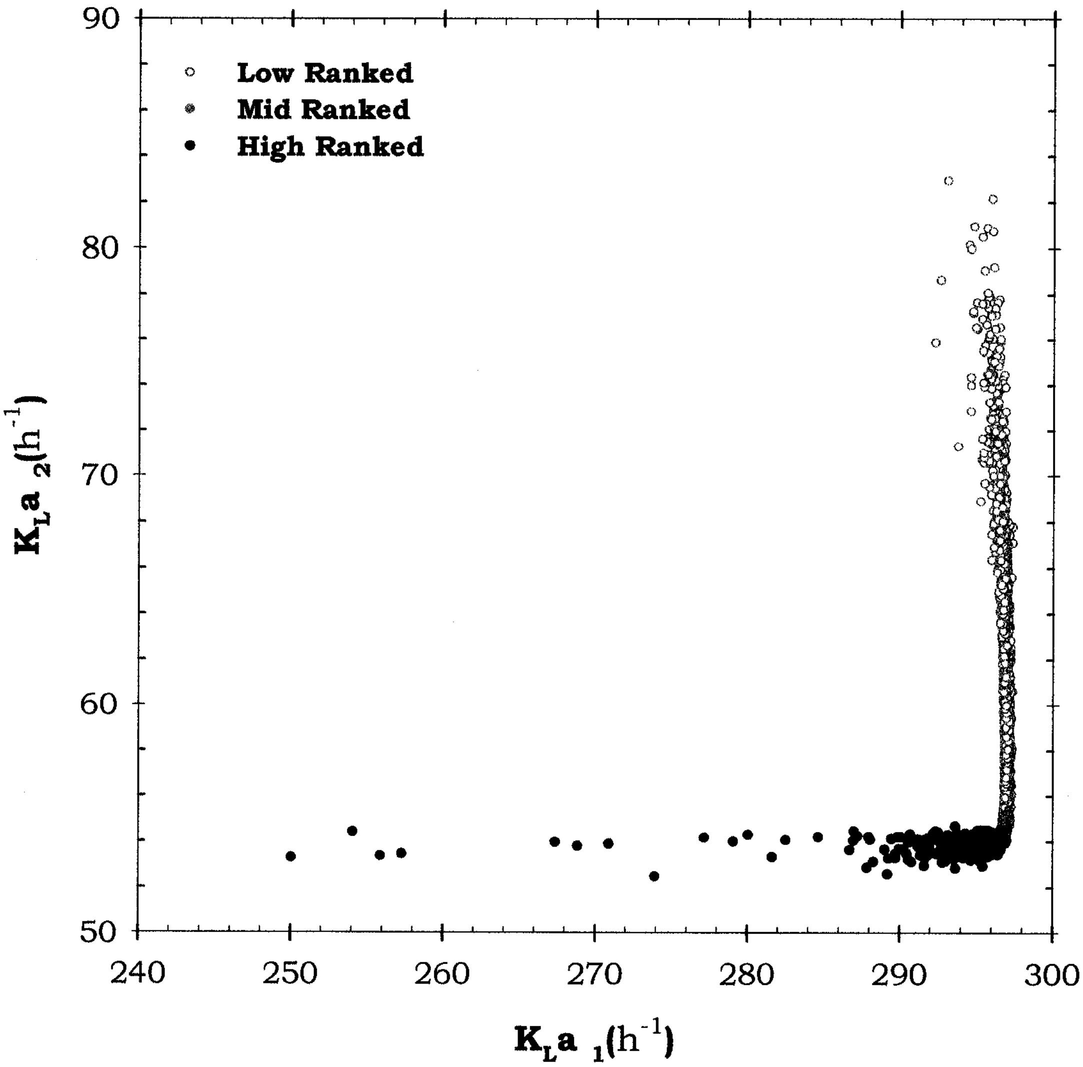

Figure 7 Optimal K_La values for each Pareto domain point for an optimal batch time of 9.7 *h*, biomass concentration of 0.95 *UOD/mL* and substrate concentration of 49.9 *g/L*

The three additional simulations incorporated a three, four and eight-level K_La strategy into the optimization process. Comparing the results of the highest ranked points in the Pareto domain generated by these simulations showed only minimal improvements in productivity and gluconic acid concentrations. However, comparing the K_La dynamics for the optimal point generated during each of these four simulations provides additional information for improving the overall production process, while reducing some of the costs of production. Figure 8 presents this comparison. The general trend in the K_La dynamics is similar for each optimal point. A high K_La value is initially required to produce the biomass required by the reaction. After approximately 3 *h*, the K_La was reduced to less than half its initial value in order to promote the production of gluconolactone, which is required for the production of gluconic acid.

Further analysis of the K_La dynamics for the three-level optimization strategy showed how the costs attributed to the power required for agitation and the superficial gas velocity could be reduced below that of the two-level optimization strategy. During the three-level optimization strategy a K_La value of 187 h^{-1} was used for the first 1.3 *h*, after which time it was increased to 265 h^{-1}, a value similar to the initial K_La value used in the two-level optimization strategy. After a total of 3 *h* this value was reduced to 70 h^{-1}. This strategy also reduced the rate of production of biomass at the beginning of the process, which in turn increased the production of gluconolactone, causing an increase in the final concentration of gluconic acid. This strategy also reduced the amount of biomass at the end of the reaction from 2.4 *UOD/mL* to 1.8 *UOD/mL*.

As can be seen in Figure 8 the results for the four-level and eight-level K_La strategy also imposed an increase in K_La at the end of the simulated reaction. This is a consequence of the to minimize the final substrate concentration: in order to reduce the amount of substrate remaining at the end of the reaction, the optimization process sought to increase the production of biomass by increasing the value of K_La, even though this increase had no benefit with respect to either the gluconic acid productivity or its final concentration.

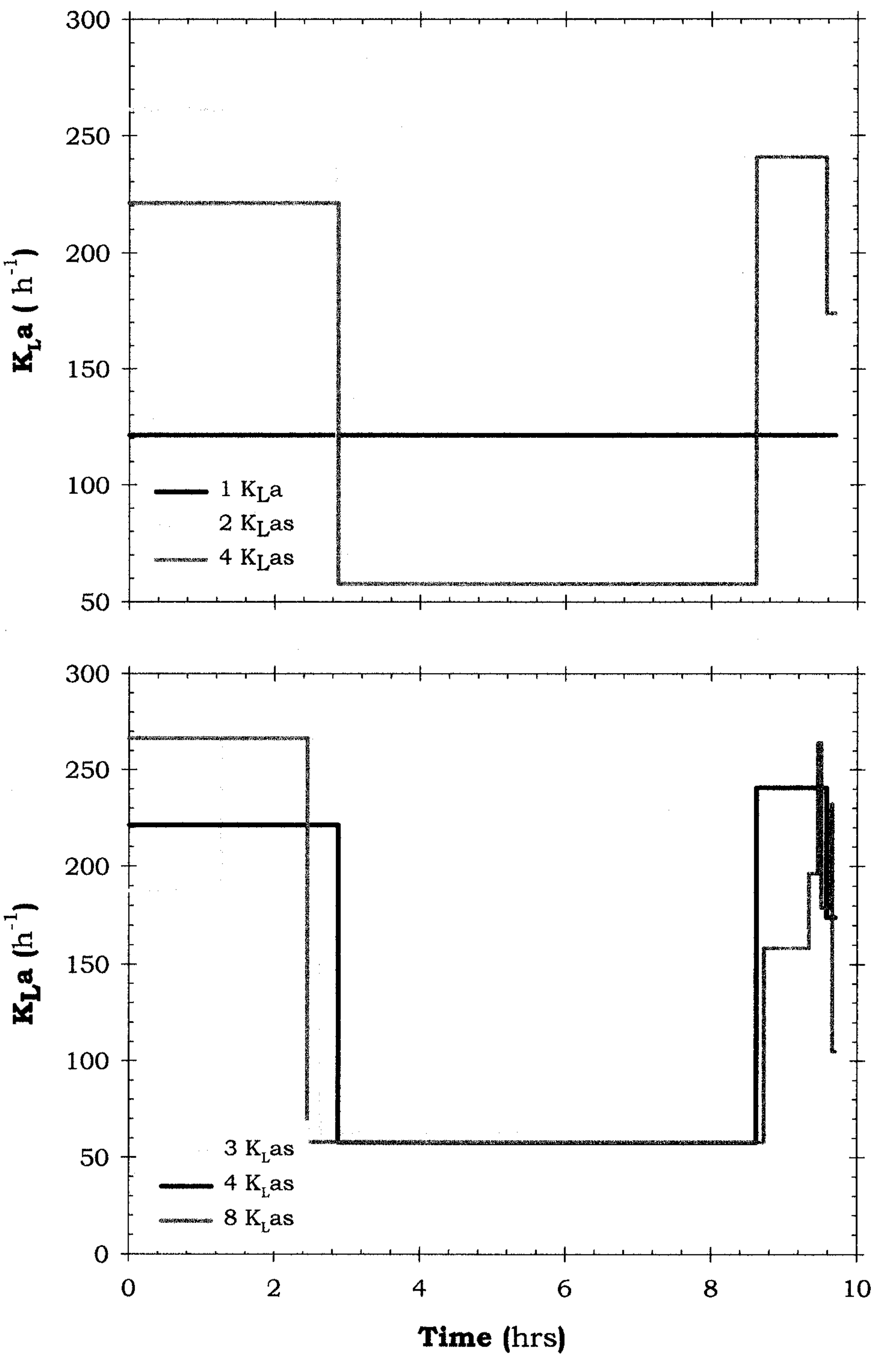

Figure 8 - Optimal $K_L a$ values for different stages of gluconic acid production. The graphs represent the highest ranked Pareto domain point for each of the five simulations studied using an optimal batch time of 9.7 *h*, biomass concentration of 0.95 *UOD/mL* and substrate concentration of 49.9 *g/L*

4 Conclusions

This main objective of this paper was to present a strategy for optimizing the production of gluconic acid in the presence of multiple and conflicting objectives. Three optimization strategies were used to determine the optimum operating region, and to study the effect of controlling the overall mass transfer coefficient on productivity, final substrate and gluconic acid concentrations. Each strategy involved generating the Pareto domain and ranking it using the Net Flow method.

The results show that the multicriteria-optimization technique described here was capable of identifying optimal operating conditions, which resulted in an acceptable compromise between the conflicting objectives. Further, the Pareto domain provided the expert with more immediate information about the relationship between the process inputs, the objective criteria, and the zone of possible solutions than that of traditional optimization algorithms.

References

Brans, J., P., Mareschal, B., and Vincke, P. H., PROMETHEE: A New Family of Outranking Methods in Multicriteria Decision Making, *Operations Research*, Vol. 84, pgs. 477-498, 1984

Bull, D. N., and Kempe, L. L., Kinetics of the Conversion of Glucose to Gluconic Acid by Pseudomonas Ovalis. *Biotechnology and Bioengineering*, Vol. 12, pgs. 273-290, 1970

Bulock, J., and Kristiansen, B., (eds) Basic Biotechnology, Academic Press, pgs. 149-172, 1987

Derot B., Gareau J., Kiss L. N., and Martel J. M., The Solver of Volvox Multicriteria Table. Multicriteria Analysis, Climaio J., (ed) Springer-Verlag, pgs. 113-126, 1997

Edgar, T. F., Himmelblau, D. M., and Lasdon, L. S., Optimization of Chemical Processes. 2nd ed., McGraw-Hill, 2001

Gagnon, H., Lounes, M., and Thibault, J., Power Consumption and Mass Transfer in Agitated Gas-Liquid Columns: A Comparative Study, *Canadian Journal of Chemical Engineering*, Vol. 76, pgs. 379-389, 1998

Ghose, T. K., and Gosh, P., Kinetic Analysis of Gluconic Acid Production by *Pseudomonas ovalis*, *Journal of Applied Chemical Biotechnology*, Vol. 26, pgs. 768-777, 1976

Johansen, T. A., and Foss, B. A., Semi-Empirical Modelling of Non-Linear Dynamic Systems Through Identification of Operating Regimes and Locals Models, In Hunt K, Irwin G, Warwick K (eds) *Neural Network Engineering in Control Systems*, Springer-Verlag, pp 105-126, 1995

Perrin, E., Mandrille, A., Oumoun, M., Fonteix, C., and Marc I., Optimisation Globale par Stratégie d'Évolution: Technique Utilisant la Génétique des Individus Diploïdes. *RAIRO-Recherche Opérationnelle*, Vol. 31, pgs.161-201, 1997

Ringuest, J. L., Multiobjective Optimization: Behavioral and Computational Considerations, Kluwer Academic Publishers, 1992

Roy, B., ELECTRE III: Algorithme de Classement Basé sur une résentation Floue des Préférences en presence de Critères Multiples. Cahiers du CERO, Vol. 20, pgs.3-24, 1978

Thibault, J., Taylor, D., and Fonteix, C., Multicriteria Optimization for the Production of Gluconic Acid, *8th International Conference on Computer Applications in Biotechnology: Modeling and Control of Biological Processes*, Chateau Frontenac, Quebec City, pgs. 24-27, 2001

Trevan, M. D., Boffey, S., Goulding, K. H., and Stranbury, P., Biotechnology: The Biological Principles, *Open University Press Biotechnology Series*, Taylor and Francis Publishing, 1987

Triantaphyllou, E., Multicriteria Decision Making Methods: A Comparative Study. *Applied Optimization Series*, Kluwer Academic Publishers, 2000

Chapter 4

Paper 3

Multi-Objective PID Controller Design for First-Order Processes with Dead time

Hayley Halsall-Whitney, David Taylor, and Jules Thibault

Department of Chemical Engineering
University of Ottawa, Ottawa, Ontario,
Canada K1N 6N5
Tel: (613) 562-5920; Fax: (613) 562-5172
thibault@genie.uottawa.ca

Journal Publication: The Journal of Process Control
Publisher: Elsevier
Publication Status: In review/To be submitted

Abstract

Designing controllers in order to satisfy various conflicting objectives can present a major challenge not easily resolved using single-objective optimization techniques or existing tuning correlations. The design of robust controllers seldom requires the optimization of a single performance criterion, but instead involves competing objectives that should be optimized simultaneously. The development of rigorous multi-objective optimization techniques in the field of systems science and operations research can be successfully applied to process control. These techniques incorporate the experience of an expert into the optimization process and provide valuable information about zone of feasible solutions.

This paper presents a multi-objective optimization strategy that generates a set of feasible solutions, called the Pareto domain, given a set of conflicting objective criteria, and classifies it using Net Flow, a technique that incorporates the knowledge of an expert into the optimization process. This strategy was used to develop PI and PID controller tuning correlations, for processes that can be approximated using a first-order-plus-dead-time model, by simultaneously minimizing three performance measures, namely, ITAE, ISDU and settling time. The correlations developed were compared to Lopez and Rovira's tuning relations for minimizing the ITAE and Ciancone correlations.

The results show that the proposed multi-objective optimization strategy was capable of identifying robust PI and PID controller tuning parameters in the presence of model uncertainty and measurement noise for disturbance inputs and set point changes.

Keywords: Multi-objective optimization, Multi-objective controller design, Pareto domain, Net Flow, PID controllers

Nomenclature

C	Global concordance index
c_k	Individual concordance index
CV_p	Controlled variable response
D	Disturbance
D_k	Discordance index of criterion k,
IAE	Integral absolute error
ISDU	Integral of the squares of the differences in the manipulated variable U
ITAE	Integral time absolute error
K_c	Controller gain
K_p	Process gain
MV	Manipulated variable
P_k	Preference threshold
Q_k	Indifference threshold
SP	Set point
T_d	Derivative time, *min*
t_f	Closed loop simulation time
T_I	Integral time, *min*
V_k	Veto threshold
W_k	Weight
Δt	Integration time step

Greek Characters

τ_p	Process time constant, *min*
θ	Dead time, *min*
Γ	Delay

1 Introduction

In process control, selecting optimal PID controller tuning parameters to satisfy multiple conflicting design objectives, can present a major challenge for the control engineer that cannot be easily resolved using established tuning correlations or traditional optimization methods. For instance, minimizing both excessive variations in the manipulated variable and settling time of the controlled variable are conflicting objectives since restricting manipulated variable movements, given a disturbance, would result in the process requiring a longer time to reach the desired steady state. The solution lies in determining a compromise between all the objectives in order to achieve acceptable controller performance.

PID controllers are the most commonly used controllers found in industry, and for which there exists a number of tuning methods, especially for processes that can be approximated using a first-order-plus-dead-time model. Traditional methods used for determining tuning parameters for PI and PID controllers include Ziegler-Nichols, Cohen and Coon, Lopez, and Rovira's tuning relations for minimizing the ITAE, to name a few (Rovira, 1981; Lopez *et al.*, 1967; Cohen and Coon, 1953; Ziegler and Nichols, 1942). Although these tuning methods have been successfully used in determining initial tuning parameters for PI and PID controllers, there are several drawbacks to their use.

1. Many traditional tuning methods were developed without considering plant-model mismatch, measurement noise, or were designed for disturbance response and not set point response. The use of these techniques could lead to inadequate controller performance if the plant operated at a wide range of operating conditions or where the process measurements contain noise.
2. Many traditional tuning methods differ in the performance objective used. The selection of the appropriate method is important since different performance criteria usually lead to different controller designs. For example, the Ziegler-Nichols tuning formulas were developed based on a 4:1 decay ratio in the response of the controlled variable, while the Lopez tuning formulas were developed based on the IAE, the ISE

or the ITAE performance measure for disturbance inputs. Therefore, the appropriate tuning method depends on the desired controller objectives.

3. Traditional tuning methods were developed based on the optimization of one objective criterion and do not address controller design problems involving multiple conflicting objectives. Therefore their use may not result in the desired controller response.

Recent development of new tuning methods, such as the Ciancone correlations, for PI and PID controllers, sought to address the issue of model parameter variation and measurement noise. The Ciancone correlations were developed using a single objective optimization technique that minimized the IAE performance measure subject to constraints on the variation of the manipulated variable (Marlin, 2000). It has been demonstrated in the literature that controllers designed using the Ciancone correlations were more robust to model parameter variations and noise when compared to traditional methods (Ciancone and Marlin, 1992).

Although the use of single objective optimization techniques can be an effective tool in determining optimal tuning parameters there are drawbacks to their use in designing controllers to satisfy conflicting objectives.

1. Combining conflicting objectives into a single objective function does not provide the decision-maker with information about trade-offs amongst the various objectives, or about alternative solutions for achieving the desired controller performance.
2. Transforming a multi-objective problem into a single objective function composed of the weighted sum of the conflicting objectives relies only, and sometimes too heavily, on the selection of the weights, which is not a trivial task. The application of different weights leads to a variety of solutions that may be optimal but do not provide the desired controller performance.

3. An optimization technique that constrains some of the objectives may bias the final solution, which can result in misleading conclusions. The decision in regards to which objective function to optimize may not be a simple one especially if all the objectives are equally important. The solution is to identify the trade-offs present in the problem, which is possible using an optimization technique that uses a population-based search.
4. Single objective optimization techniques provide only one optimal solution even if there are other possible solutions. These techniques are often plagued with the problem of finding the global optimum or multiple global optima, and could miss possible solutions if the functions are non-convex, multi-modal or discontinuous. They also require information about function derivatives and an initial estimate of the solution, which may not be readily available.
5. Traditional optimization techniques do not incorporate the practical experience and knowledge of the decision-maker in regard to the overall behavior of the process and the required controller performance.

In general, the application of single objective optimization techniques has been appropriate due to the well-behaved nature of the design problem when a single criterion or non-conflicting criteria are used. However, the growing diversity and complexity of controllers have led to an increased interest in the development and application of multi-objective optimization techniques to controller design in the literature (Liu et al., 2003; El-Kady *et al.*, 2003; Herreros *et al.*, 2002; Visioli, 2001; Kang and Bien, 2000; Hu *et al.*, 1998; Rangan and Poolla, 1997; Man *et al.*, 1997; Marrison and Stengel, 1997; Fonseca and Fleming, 1996, Kim et al., 1995, Krishnakumar and Goldberg, 1992; Oliveira *et al.*, 1991), especially for determining tuning parameters for PID controllers (Vlachos *et al.*, 1999; Chen *et al.*, 1995; Wang and Kwok, 1992; Porter and Jones, 1992). Multi-objective optimization offers the control engineer the possibility of simultaneously optimizing multiple conflicting objectives that reflect the true nature of the design problem without resorting to single objective optimization techniques. The result is a set of feasible solutions representing optimal controllers. These solutions are optimal in the sense that no one solution is better than any other in the domain when compared on all criteria (Deb, 2001, Caello, 1999). The decision-maker's experience and knowledge is then incorporated into the optimization procedure in order to classify the available alternatives in terms of his or her preference (Doumpos and Zopounidis, 2002; Derot *et al*, Brane *et al.*, 1984).

This paper presents a robust multicriteria optimization strategy for developing tuning correlations for PI and PID controllers that simultaneously minimizes three objective criteria, namely, the ITAE, the ISDU, and the settling time, while taking into consideration model parameter variation and measurement noise. This optimization strategy generates a set of feasible solutions, called the Pareto domain, by simultaneously minimizing multiple conflicting objectives, and classifies the entire domain using Net Flow, a technique that incorporates the knowledge of an expert into the optimization process. The results were compared to the Minimum Error Integral Tuning formulas for set point changes developed by Rovira, the Minimum Error Integral Tuning formulas for disturbance inputs developed by Lopez, and the Ciancone correlations (Ciancone and Marlin 1992; Rovira, 1981; Lopez *et al.*, 1967).

2 Theory

2.1 Multi-objective PID Controller Design

Figure 1 presents the block diagram of the single loop feedback control system used in the development of PI and PID tuning correlations using a first-order-plus-dead-time model subject to set point changes or unmeasured disturbances, plant-model mismatch and measurement noise. Four sets of correlations were developed using the following cases studies for set point and for disturbance changes.

1. Case Study 1: Nominal plant represented by an accurate model.
2. Case Study 2: Nominal plant represented by a model with uncertainty in the parameter estimates. To evaluate the impact of model uncertainty, a series of 350 simulations were performed by randomly changing the nominal model parameter values using a Gaussian perturbation with $\mu = 0$ and $\sigma = 0.25$.

3. Case Study 3: Nominal plant represented by model with uncertainty in the parameter estimates and subjected to measurement noise. Two levels of measurement noise were considered. They were modeled using a Gaussian distribution with $\mu = 0$ and $\sigma = 0.01$, and $\mu = 0$ and $\sigma = 0.05$.

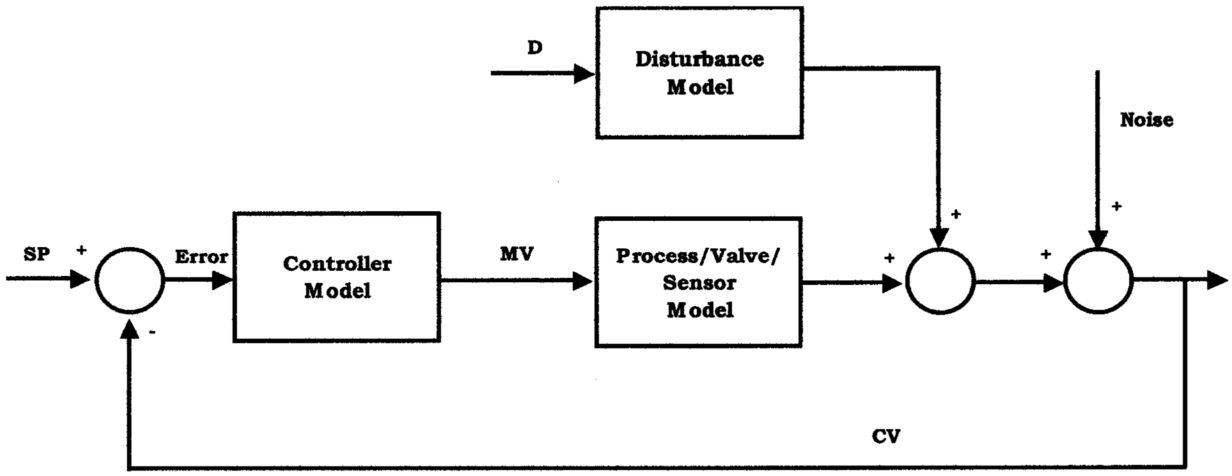

Figure 1 - Block diagram of a single-loop feedback control system.

2.1.1 First-Order-Plus-Dead-Time Model

The first-order-plus-dead-time model, which was used to represent the nominal plant, was approximated using the following algebraic equation.

$$CV_{p_n} = \frac{K_p \Delta t}{\tau_p} MV_{(n-\Gamma)} + \left(1 - \frac{\Delta t}{\tau_p}\right) CV_{P(n-1)} \quad \text{.......} (1)$$

In this equation the dead time was simulated by a delay, Γ, where $\Gamma = INT(\theta/\Delta t)$(Marlin, 2000). The values used for the model parameters K_p, and τ_p, were 1 and 5, respectively. The models representing the valve and sensor were equated to 1 and the values for θ were varied from 0-45 during a series of multi-objective optimization simulations. The values for Δt and the t_f were calculated using the dead time, θ, and process time constant, τ_p, where $\Delta t = 0.01(\theta + \tau_p)$, and $t_f = 6(\theta + \tau_p)$.

The disturbance model was approximated using the following first-order model. The disturbance model parameters for K_d and τ_d were 1 and 5, respectively.

$$CV_{d_n} = \frac{K_d \Delta t}{\tau_d} D_n + \left(1 - \frac{\Delta t}{\tau_d}\right) CV_{d(n-1)} \qquad (2)$$

2.1.2 PID Controller

The velocity form of the PID controller with derivative action taken on the measurement, rather than on the error, was used during the simulation and is described as follows.

$$MV_n = MV_{n-1} + K_c\left[(SP_n - CV_n) - (SP_{n-1} - CV_{n-1})\right] + \left[\frac{K_c \Delta t}{T_I}(SP_n - CV_n)\right] + \left[\frac{K_c T_d}{\Delta t}(-CV_n + 2CV_{n-1} - CV_{n-2})\right] \qquad (3)$$

In this equation K_c is the controller gain, T_I integral time constant, and T_d is the derivative time constant. Optimal values for these parameters were found using a multi-objective optimization strategy for a given dead time.

During the simulation, bounds were set on the manipulated variable in order to prevent the simulation from experiencing values of infinity for the manipulated variable. The bounds were set so that they would not interfere with the results of the simulation. The simulation was carried out using deviation variables. Therefore, the controlled variable was initially at a steady state value of 0 before experiencing a step change of 1 at time ($2\Delta t$). The final steady state value for the manipulated variable is $\Delta SP/K_p$, given a unit step change in the set point, or - $\Delta DV/K_p$, given a unit step change in the disturbance input. For example, for a unit step change in the set point, if K_p is set to 1, then the final steady state value is 1 and the bounds on the manipulated variable would be 1 ± 1000. Data points experiencing such excessive variations in the manipulated variable were not present in the Pareto domain.

2.1.3 Optimization Objectives

Although there are a number of objective criteria that can be used to determine optimum tuning parameters for the PID family of controllers, this work selected to minimize the ITAE, the ISDU and the settling time. These measures are examples of some of the most commonly used criteria used in the literature and in industry for analyzing PID controller performance. Figure 2 represents the multi-input, multi-output optimization strategy used for developing optimal tuning parameters for the PI and PID controllers. The initial ranges used for the dimensionless inputs were $K_c K_p \in [0.01, 10]$, $T_I/(\theta+\tau) \in [0.001, 1]$, and $T_d/(\theta+\tau) \in [0.001, 1]$ (Marlin, 2000).

The ITAE measures the cumulative deviation of the controlled variable from the set point. It penalizes deviations that endure for long periods of time, and is described using the following equation.

$$ITAE = \int_0^{\infty} t\left|SP - CV\right| dt \approx \sum_{n=1}^{M} t\left|SP_n - CV_n\right| \Delta t \quad \text{.......... (4)}$$

The ISDU is a performance measure for the manipulated variable movements. Its objective is to minimize the amplitude of manipulated variable moves, and is described using the following equation.

$$ISDU = \sum_{n=1}^{M} \left(MV_n - MV_{n-1}\right)^2 \quad \text{.......... (5)}$$

The settling time is the amount of time that is required for the process to reach within ±5% of its new steady state value after experiencing a unit set point change and ±1% of its new steady state value after experiencing a unit disturbance change. In both cases a short settling time is desired.

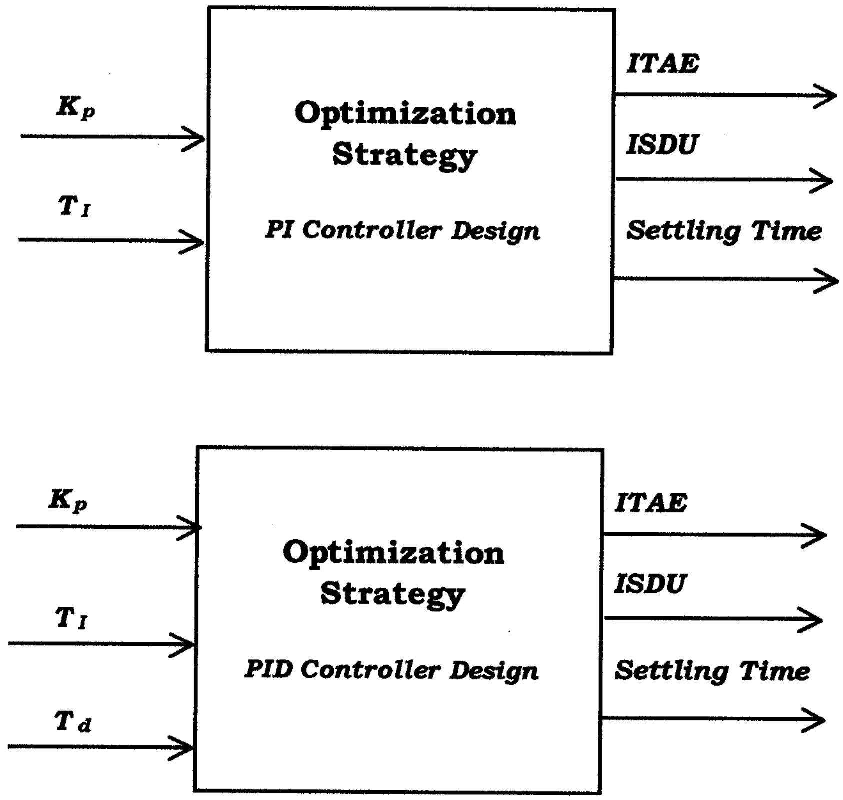

Figure 2- Optimization strategy for determining tuning parameters for PI and PID Controllers.

2.2 Principles of Multi-objective Optimization

A multi-objective optimization problem involves the simultaneous optimization of multiple objectives. These objectives can be either maximized, minimized or a combination of both, and can be subjected to constraints that the feasible solutions must satisfy. The following expression describes the formulation of a multi-objective problem.

$$Min/Max/Combination \quad F(x) = \left[f_1(x_1..x_n), f_2(x_1..x_n), ...f_n(x_1..x_n)\right]$$

$$\begin{aligned} \text{subject to} \quad & G_{i=1...n}(x) \geq 0, \\ & H_{i=1...n}(x) = 0, \quad i = 1,2,3...n \end{aligned} \qquad \text{..................} (6)$$

$$\text{where} \quad x_{i=1...n}^{(Lower\ Bound)} \leq x_{i=1...n} \leq x_{i=1...n}^{(Upper\ Bound)}$$

In this expression, the input space, referred to as the *Decision Variable Space*, is predefined by the ranges associated with the independent variables $\boldsymbol{X} = (x_1, x_2, x_3...x_n)^T$. While the output or solution space, expressed by $\boldsymbol{F(x)}$, is called the *Pareto domain* (Deb, 2001,Coello, 1999). There are two common principles used in most multi-objective optimization techniques, namely, the concept of domination and Pareto-optimality (Deb, 2001).

Concept of Domination

The concept of dominance is an approach used for determining a fitness score for each member in a population. It was first introduced as a method of determining a point's fitness within the current population of data by a technique called Multi-objective Genetic Algorithm (MOGA) (Fonseca and Fleming, 1993). The concept of domination is defined in this paper as follows. Given two points, $\boldsymbol{X_1} = (x_1, x_2, x_3...x_n)^T$ and $\boldsymbol{X_2} = (x_1, x_2, x_3...x_n)^T$ within a population's decision variable space, the following properties are defined.

1. $\boldsymbol{X_1}$ dominates $\boldsymbol{X_2}$ if the values of all objective criteria for $\boldsymbol{X_1}$ are better than the values for the corresponding objective criteria for $\boldsymbol{X_2}$.
2. $\boldsymbol{X_1}$ does not dominate $\boldsymbol{X_2}$ if the value of at least one of the objective criteria for $\boldsymbol{X_1}$ is equal or worse than the value of the corresponding objective criterion for $\boldsymbol{X_2}$.

If X_1 dominates X_2 then X_2 is called a dominated point, and if X_1 does not dominate X_2 and X_2 does not dominate X_1 then both are called non-dominated points with respect to one another.

Pareto Optimality

Pareto optimal solutions, called the *Pareto domain*, are a set of non-dominated solutions where no one solution is better than any other in the domain when compared on all objective criteria. The Pareto domain is generated by simultaneously optimizing multiple criteria and represents feasible alternatives to the multi-objective problem. It provides the decision-maker with information about the trade-offs arising from the use of multiple conflicting objectives, and on the importance of each objective in the set of feasible solutions.

2.3 Optimization Strategy

The multi-objective optimization strategy proposed in this work consists primarily of two steps: (1) the generation of the Pareto domain using the Dual Population Evolutionary Algorithm, and (2) the classification of the domain by ranking all the data points approximating the Pareto domain using the Net Flow method, which incorporates *a priori* process knowledge supplied by the decision-maker.

2.3.1 Approximating the Pareto domain

Dual Population Evolutionary Algorithm

The Dual Population Evolutionary Algorithm incorporates the concept of domination when generating the Pareto domain. The general approach is described as follows (Halsall-Whitney *et al.*, 2003; Thibault *et. al.*, 2001; Perrin *et al.*, 1997).

1. For each input variable, a value was randomly selected from within its predefined range of variation, using a random number generator. The model was then solved for

this set of inputs in order to generate values for the objective criteria. Initially, M points were generated to fully span the decision space using a pseudo-random number generator, which was altered so that the boundaries of the decision space would be taken into consideration. The number of points M is specified by the user and must be large enough to adequately approximate the Pareto domain.

2. All M points were compared two by two, based on the values for their objective criteria, in order to determine the number of times a point had been dominated by another in the current output space. Each time a point was dominated, its domination score was incremented by 1; otherwise it remained unchanged from an initial score of 0. At the end of the comparison process, the data set was sorted in order of domination scores, starting from data points that were never dominated (i.e. having a score of 0) and ending with the data point displaying the highest frequency of domination.
3. All non-dominated data points, N_0, were then selected, along with a fraction of the dominated points having the lowest frequency of domination, to be used in generating the next approximation to the Pareto Domain. This selection process can be described by the following equation:

$$N = N_0 + INT\left[(F_s)(M - N_0)\right] \quad (7)$$

In this equation, F_S is the survival fraction of the dominated data points and $INT(x)$ yields the integer value of x. In this study a survival fraction of 0.3 was used. This ensures that only the better elements of the population survived to participate in the creation of the new generation of possible solutions.

4. A total of $(M\text{-}N)$ new data points were then generated to replace those points that were eliminated during the selection process. Combining these new points with those points selected from the previous set yielded a new set of M data points. Each new point, $I_{p,k}$, was generated by randomly selecting one non-dominated point, $I_{p,i}$, and one dominated point, $I_{p,j}$, from the set of points retained in the previous

generation. The new set of inputs for the new point was then determined according to the following equation.

$$I^n_{p,k} = D_p I^n_{p,i} + (1 - D_p) I^n_{p,j} \quad \text{(8)}$$

In this equation, D_p is a randomly selected number between 0 and 1. A new value for D_p was selected each time a new input, in the vector of inputs representing $I_{p,k}$, was determined.

5. Steps (1)-(4) were repeated until M non-dominated data points, which was specified by the user as the number of points required to adequately define the Pareto domain, were generated. By eliminating in successive generations all data points that were not Pareto-efficient, it was assumed that this final set of M possible solutions adequately approximates the Pareto domain.

2.3.2 Algorithm for Classifying the Pareto Domain

The optimization strategy used in this work incorporates a classification technique, called Net Flow (NFM). It developed as a result of modifications made to the ELECTRE III method, and was used to rank the Pareto domain (Doumpos and Zopounidis, 2002; Scarelli and Narula, 2002; Triantaphyllou, 2000; Derot *et al.*, 1997; Brans *et al.*, 1984; Roy, 1968). In the NFM, *a priori* knowledge of the process, expressed by an expert, is incorporated into the optimization routine using four sets of parameters to classify all the data points of the Pareto domain. These parameters are described as follows:

1. The first parameter gives the relative importance of each criterion, expressed as a weight W_k. In this algorithm, the weights are normalized:

$$\sum_{k=1}^{n} W_k = 1 \quad \text{(9)}$$

2. The second parameter refers to the indifference threshold (Q_k), which defines the range of variation of each criterion for which it is not possible for the decision-maker to favor the criterion of one point over the corresponding criterion of another.
3. The third parameter refers to the preference threshold (P_k). If the difference between two values for a given criterion exceeds this threshold, a preference was given to the better criterion.
4. The fourth parameter refers to the veto threshold (V_k), which serves to ban a point relative to the other if the difference between the values of a criterion was too high to be tolerated. A point was banned if the veto threshold was violated for at least one of the criteria even if the other criteria were acceptable.

These three thresholds are defined for each criterion such that the following relationship holds:

$$0 \leq Q_k \leq P_k \leq V_k \quad (10)$$

They represent a reference range established by the decision-maker to assess the values of the objective criteria for each alternative in the Pareto domain (Roy, 1978). Although this method makes use of weights it does not only rely on them, but instead uses them in combination with the other parameters, which incorporate the preferences of the decision-maker. The NFM algorithm is described as follows.

1. First, for each combination of points in the Pareto domain, the difference between the values F_k of each criterion k was calculated by comparing point i with point j using the following equation.

$$\Delta_k[i, j] = F_k(i) - F_k(j) \begin{cases} i \in [1, M] \\ j \in [1, M] \ , j \neq i \\ k \in [1, n] \end{cases} \quad (11)$$

In subsequent equations, minimizing a criterion required using $\Delta_k[i,j]$, while maximizing a criterion required using $-\Delta_k[i,j]$.

2. Using the values of $\Delta_k[i,j]$, the concordance index $c_k[i,j]$ for each criterion was determined for all n criteria and for each pair of data points using the following relationships.

$$c_k[i,j] = \begin{cases} 1 & \text{if } \Delta_k[i,j] \le Q_k \\ \dfrac{P_k - \Delta_k[i,j]}{P_k - Q_k} & \text{if } Q_k < \Delta_k[i,j] \le P_k \\ 0 & \text{if } \Delta_k[i,j] > P_k \end{cases} \quad (12)$$

The concordance index measures the strength of the argument that when comparing point *i* to point *j* for a given criterion k the value of 'F_k *(i) is at least as good as* F_k *(j)*' when compared to values specified by the decision-maker in the reference range for a given criteria (Roy, 1978). Figure 3a illustrates how the concordance index was determined using the values of the calculated differences, the indifference threshold, and the preference threshold. For a difference smaller than the indifference threshold, the corresponding concordance index is 1. Between the indifference and preference thresholds, it varies linearly from 1 to 0. For a difference larger than the preference threshold, the concordance index was set to 0.

3. The weighted sum of individual concordance indices was calculated to determine the global concordance index.

$$C[i,j] = \sum_{k=1}^{n} W_k c_k[i,j] \quad \begin{cases} i \in [1,M] \\ j \in [1,M] \end{cases} \quad (13)$$

4. A discordance index $D_k[i,j]$ was calculated for each criterion k using the preference and veto thresholds.

$$D_k[i,j]=\begin{cases}0 & if\ \Delta_k[i,j]\le P_k\\ \dfrac{\Delta_k[i,j]-P_k}{V_k-P_k} & if\ P_k<\Delta_k[i,j]\le V_k\\ 1 & if\ \Delta_k[i,j]>V_k\end{cases} \qquad (14)$$

The discordance index measures the strength of the argument that when comparing point *i* to point *j* for a given criterion k the value of *'F_k (i) is significantly worse than F_k (j)'* when compared to values specified by the decision-maker in the reference range for a given criteria (Roy, 1978). Figure 3b illustrates how the discordance index was determined using the preference and veto thresholds. For a difference smaller than the preference threshold, the discordance index is 0. Between the preference and veto thresholds, it varies linearly from 0 to 1, and for a difference larger than the veto threshold, the concordance index is set to 1.

5. Using the concordance and discordance indices, the relative performance of each pair of domain points was evaluated by calculating each element of the outranking matrix $\sigma[i,j]$ using the following equation.

$$\sigma[i,j]=C[i,j]\left(\prod_{k=1}^{n}1-(D_k[i,j])^3\right)\begin{cases}i\in[1,M]\\ j\in[1,M]\end{cases} \qquad (15)$$

Each element $\sigma[i,j]$ measures of the quality of point *i* relative to point *j* in terms of the three criteria. An element $\sigma[i,j]$ close to 0 indicates that point *j* outranks point *i*. If the value is near 1, then data point *i* may outrank point *j* or simply be located in the vicinity of point *j*.

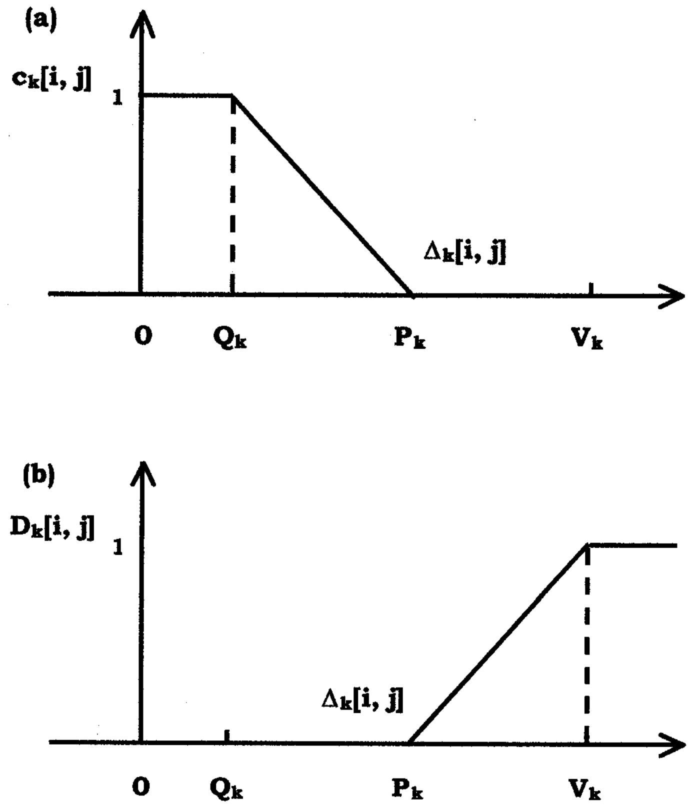

Figure 3 -(a) Concordance index, and (b) discordance index calculations used in the Net Flow algorithm to determine ranking scores for the Pareto domain points

6. The final ranking score for each data point in the Pareto domain was obtained by summing individual outranking elements associated with each domain point as follows.

$$\sigma_i = \sum_{j=1}^{M} \sigma[i, j] - \sum_{j=1}^{M} \sigma[j, i] \quad \text{.......... (16)}$$

The first term evaluates the extent to which element i performs relative to all the other points in the Pareto domain, while the second term evaluates the performance of all the other points relative to point i. The points were then sorted from highest to lowest according to ranking score. The data point with the highest ranking was the one that best satisfies the set of preferences provided by the decision-maker.

The results of the Net Flow method were used to divide the Pareto domain into zones containing the high-ranked, mid-ranked, and low-ranked domain points in order to identify graphically the location of the optimal region.

3 Results and Discussion

The case studies resulted in the development of four sets of PI and PID tuning correlations for set point change, and four sets of PID tuning correlations for disturbance response using the proposed multi-objective optimization strategy. Each set of correlations generated ten Pareto domains representing fractional dead times, $\theta/(\theta+\tau_p)$, 0 - 0.9 in increments of 0.1, by simultaneously minimizing ITAE, ISDU, and settling time. The resulting set of feasible solutions was classified using Net Flow. Table 1 gives the values of the Net Flow parameters used in developing the correlations for set point change. Table 2 is an example of the Net Flow parameters used to classifying some of the Pareto domains used to determine the correlations for the disturbance response. Criteria 1, 2 and 3 represent the ITAE, ISDU and settling time performance measures, respectively.

Table 1 – Net Flow parameter values used for classifying the Pareto domains generated in the development of tuning correlations for PI and PID controllers for set point change.

Criterion	**NFM Parameters**			
k	W_k	Q_k	P_k	V_k
1	0.35	5	10	15.0
2	0.35	0.01	0.05	0.1
3	0.3	1	3.0	6.0

Table 2 – Net Flow parameter values used for classifying the Pareto domains generated in the development of tuning correlations for the PID controller for disturbance response.

Criterion	NFM Parameters			
k	W_k	Q_k	P_k	V_k
1	0.35	5	10	15.0
2	0.35	0.01	0.05	0.1
3	0.3	0.5	0.1	2.0

For each case study, the correlations developed related the fractional dead time, $\theta/(\theta+\tau_p)$, to the dimensionless gain, $K_c K_p$, integral time, $T_I/(\theta+\tau_p)$, and derivative time, $T_d/(\theta+\tau_p)$. The optimal point for each Pareto domain generated, which was the point that ranked the highest by Net Flow, was used in the development of the tuning correlations. These correlations were compared to the Minimum Error Integral Tuning formulas for set point changes developed by Rovira, the Minimum Error Integral Tuning formulas for disturbance inputs developed by Lopez, and the Ciancone correlations in order to demonstrate the robustness of this multi-objective optimization strategy (Rovira, 1981; Ciancone and Marlin 1992).

Case Study 1: Nominal Plant

In this case study, given a step change in the set point, a first-order-plus-dead-time model represented the nominal plant with known parameters without measurement noise. Figure 4 presents the decision space and Pareto domain associated with a fractional dead time of 0.5 for a PID controller considering a unit step change in the value of the set point. It represents 15000 non-dominated points. Therefore, no one point in the domain is better on all three objective criteria than any other point in the domain. In this case, the value of θ for the first-order-plus-dead-time model was 5.

The results show the trade-off between the ITAE, ISDU and settling time, which is due to the simultaneous minimization of all three objective criteria. For instance, minimizing the

ITAE and the settling time maximizes the ISDU. Therefore, the solution will be a compromise between these three objectives.

The difference between the high-ranked, mid-ranked and low-ranked points lies in the excessive movement of the manipulated variable in an attempt to reduce overshoot in the controlled variable. The high-ranked points experienced a rapid smooth response in the controlled variable to a set point change without excessive control action. While the mid-ranked to low-ranked points experience overshoot in the manipulated variable of upwards of 30% of the final steady state value in order secure a smooth response in the controlled variable with minimal overshoot.

The ranges for the optimal controller inputs, represented by the decision space, were reduced to $K_c \in [0.3, 1.5]$, $T_I \in [3.5, 8.5]$, and $T_d \in [0.01, 2]$ from initial values of $K_c \in [0.01, 10]$, $T_I \in [0.01, 10]$, and $T_d \in [0.01, 10]$. The ranges for the top 1000 points were $K_c \in [0.614 - 0.669]$, $T_I \in [5.8 - 6.17]$, and $T_d \in [0.792 - 0.398]$, which correspond to ranges for the ITAE, ISDU and settling time of [51-55], [0.39-0.47] and [14.8-15.8], respectively. The values for the controller parameters of the optimal point were 0.638, 5.97, and 0.58 for K_c, T_I, and T_d, respectively. Figure 5 presents the controlled and manipulated variable responses to a step change in the set point using these tuning parameters. The results show a smooth response of the controlled variable without excessive manipulated variable movements.

In comparison, the Rovira tuning relations for minimizing the ITAE provides for a more aggressive controller, with manipulated variable action similar to that experienced by the low-ranked points in the Pareto domain, in order to achieve a smooth response in the controlled variable with minimal overshoot. The Rovira tuning parameters for a fractional dead time of 0.5 were 0.965, 7.7, and 1.54 for K_c, T_I, and T_d, respectively. The corresponding values for the performance measures were, 37, 1.0, and 11.5 for the ITAE, ISDU and the settling time respectively.

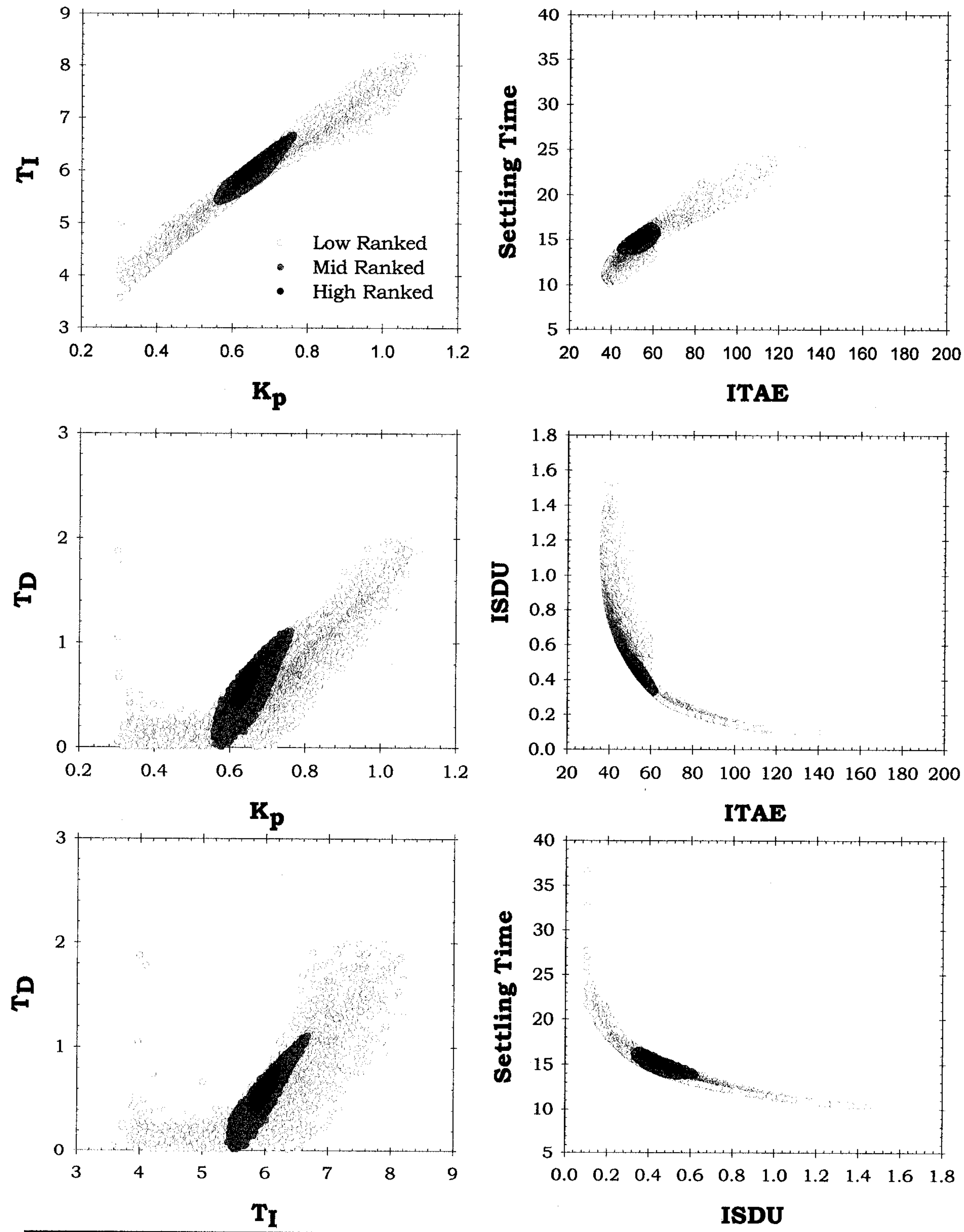

Figure 4 – Decision space and Pareto domain for a PID controller for the nominal plant.

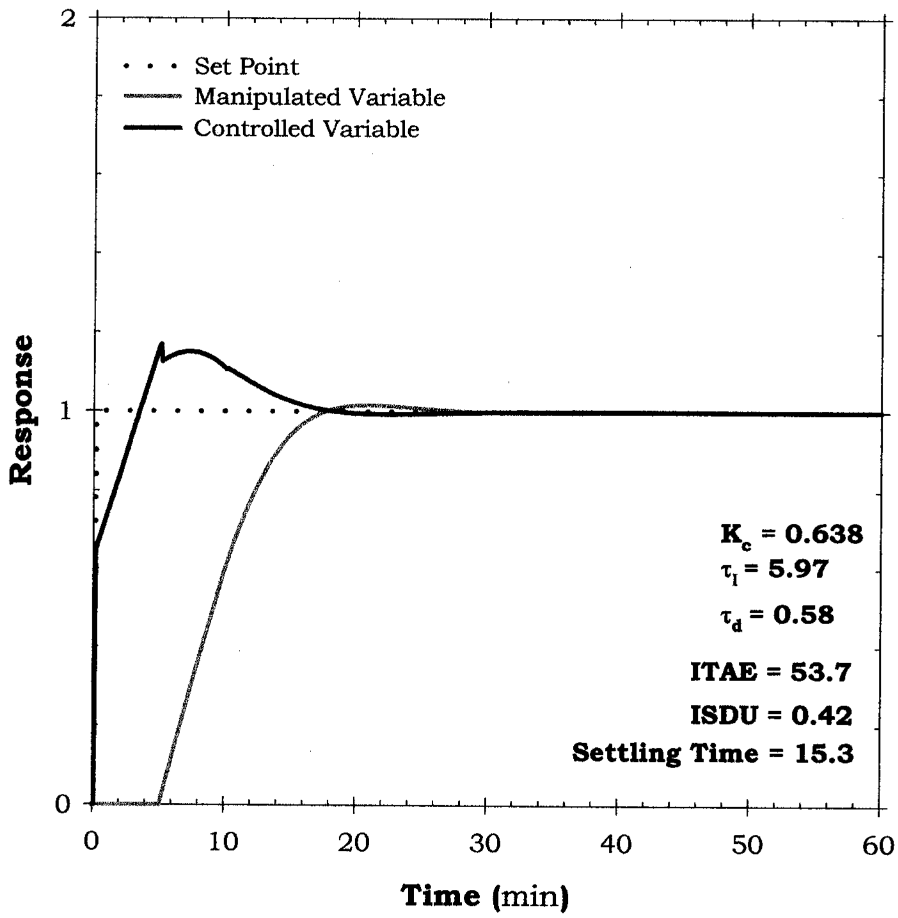

Figure 5 – Controlled and manipulated variable response to a step change in the set point for the optimal point in the Pareto domain identified using the Net Flow method, for a PID Controller.

Case Study 2: Nominal Plant with Parameter Variation

In this case study, given a step change in the set point, a first-order-plus-dead-time model represented the nominal plant with known variation in the model parameters but without measurement noise. A Monte Carlo simulation, consisting of 350 runs, was used to simulate parameter variation. Therefore, for a given set of controller tuning parameters, the performance measures were calculated by running the closed loop simulation 350 times each time changing the parameters of the model from their nominal values and averaging the individual performance measures. The parameter values for the altered plants were selected at random using a Gaussian distribution with standard deviation $\sigma = 0.25$ in the parameter values. The resulting distributions in the model parameters and the performance measures were approximately normal. A Monte Carlo simulation using between 400 and 10000 runs did not change the values of the averaged performance measures.

Figure 6 represents the Pareto domain and the corresponding input space for this case study considering a unit step change in the value of the set point. It represents 2500 non-dominated points. The results show that the feasible set of tuning parameters were further restricted because of the inclusion of parameter variation into the simulation from those obtained for the nominal process in Case Study 1. Therefore, the number of feasible solutions available to the control engineer has been reduced. However the values for the tuning parameters of the optimal point were similar to those obtained for the nominal plant. This suggests that the tuning parameters identified for the optimal point for the nominal plant were robust to variations in the model parameters for the first-order-plus-dead-time model used. This can be explained by the fact that three objective criteria were simultaneously optimized.

The ranges for the optimal controller inputs, represented by the decision space, were reduced to $K_c \in [0.3 - 0.98]$, $T_I \in [3.3 - 6.9]$, and $T_d \in [0.01 - 1.9]$ from values of $K_c \in [0.3 - 1.5]$, $T_I \in [3.5 - 8.5]$, and $T_d \in [0.01 - 2]$ obtained in Case Study 1 for the nominal plant. The tuning parameters for the optimal point obtained for this case study for a fractional dead time of 0.5 was 0.645, 5.67, and 0.77 for K_c, T_I, and T_d, respectively. The corresponding values for the performance measures were, 58, 0.44, and 14.6 for the ITAE, ISDU and the settling time respectively. Figure 7 shows the response of the controlled and manipulated variables using the tuning parameters for the optimal point in the Pareto domain.

In comparison to Case Study 1, the increase in the ITAE was the result of the controlled variable response experiencing a larger overshoot. However, there was only a marginal increase of 0.02 in the ISDU, and the settling time was not adversely affected.

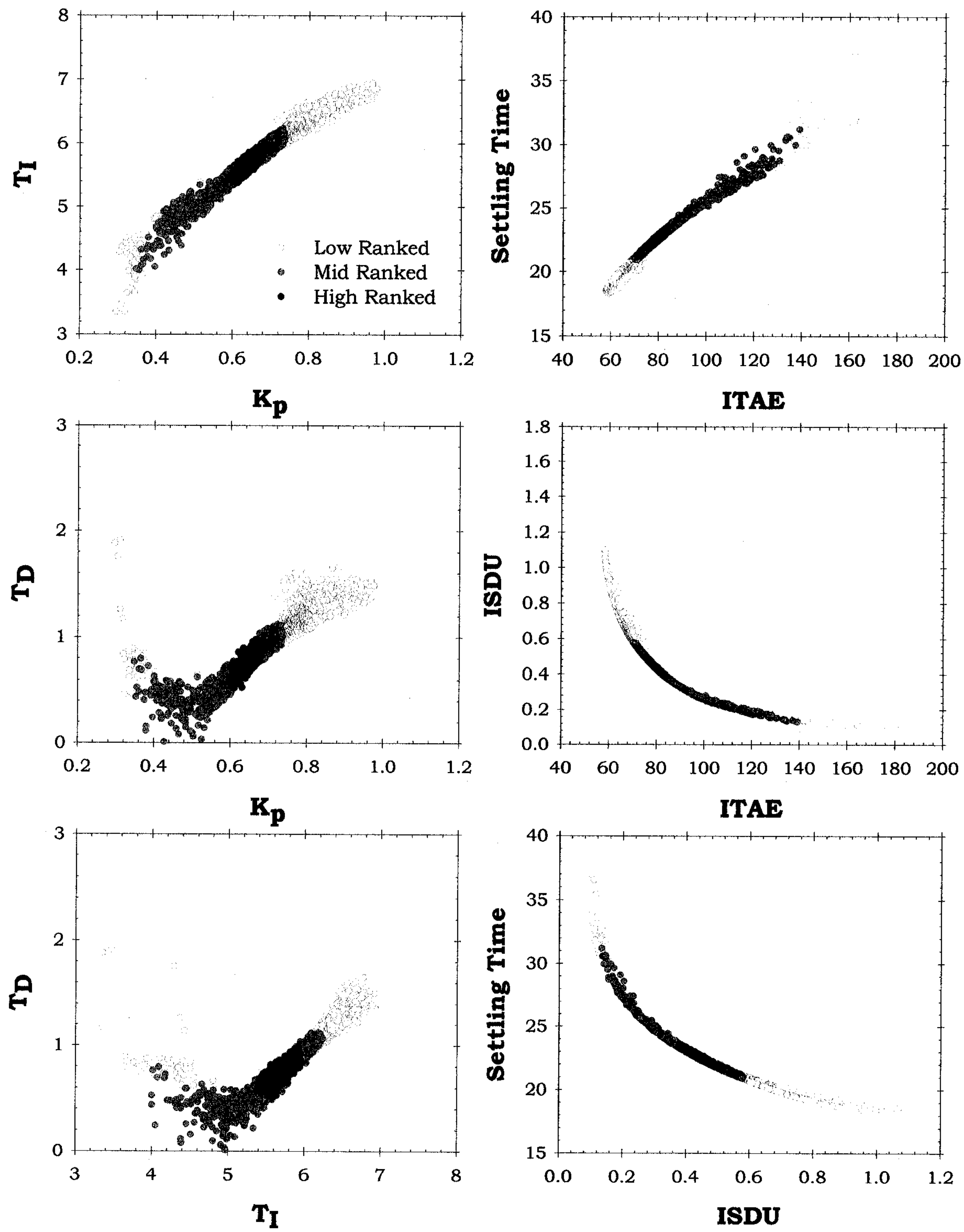

Figure 6 – Pareto domain for a PID controller showing the effect of variance in the process model parameters and measurement noise on controller design.

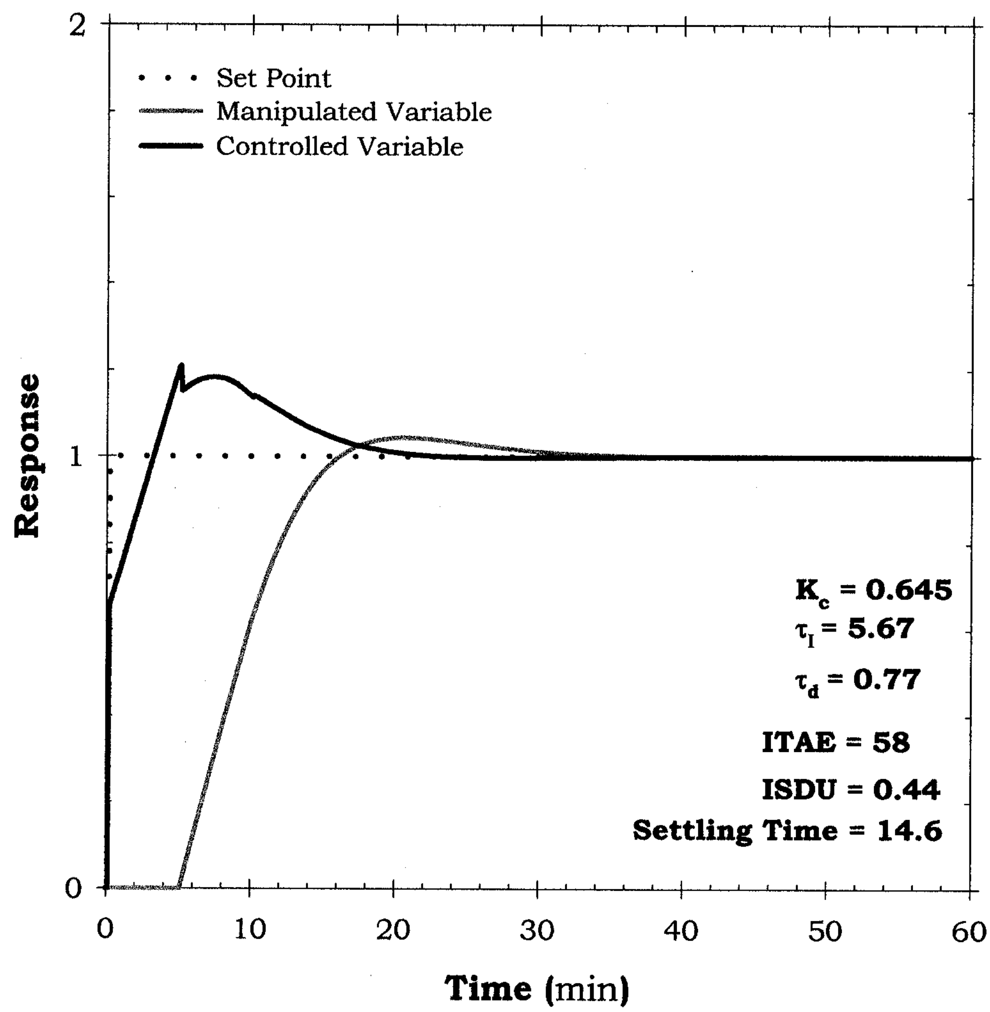

Figure 7 – Controlled and manipulated variable responses to a step change in the set point for the optimal point in the Pareto domain identified using the Net Flow method, and considering parameter variation in the nominal model.

Case Study 3: Nominal Plant with Parameter Variation and Noise

In this case study, given a step change in the set point, a first-order-plus-dead-time model represented the nominal plant with known variation in the model parameters and measurement noise. Two levels of measurement noise were considered and were modeled using a Gaussian distribution, with mean $\mu = 0$ and standard deviation $\sigma = 0.01$ and $\sigma = 0.05$.

Figures 8 and 9 show the effect of including parameter variation and noise on the shape of the decision space and the Pareto domain using a fractional dead time of 0.5 for a PID controller for noise frequencies with standard deviations $\sigma = 0.01$ and $\sigma = 0.05$, respectively, considering a unit step change in the value of the set point. The Pareto domain consists of 2500 non-dominated points. The results show that the shape of the Pareto domain has been reduced from the range determined for the nominal plant obtained in Case Studies 1 and 2. The results show that the ranges for the controller inputs, represented by the decision space, were $K_c \in [0.3 - 1.0]$, $T_I \in [3.5 - 7]$, and $T_d \in [0.01\text{-}2]$ and $K_c \in [0.01 - 1.3]$, $T_I \in [0.05 - 7.5]$, and $T_d \in [0.01\text{–} 2]$ for parameter variation and measurement noise, with $\sigma = 0.01$ and $\sigma = 0.05$, respectively, compared to the nominal plant, which were $K_c \in [0.3 - 1.5]$, $T_I \in [3.5 - 8.5]$, and $T_d \in [0.01 - 2]$. However, the ability of the optimization strategy to identify optimal tuning parameters was not affected The values for the optimal tuning parameters for a fractional dead time of 0.5 were 0.73, 5.98, and 0.51 using $\sigma = 0.01$ and 0.68, 5.59, and 0.18 using $\sigma = 0.05$ for K_c, T_I, and T_d, respectively. The corresponding values for the performance measures were 54.5, 0.56, and 12.9 using $\sigma = 0.01$, and 62, 0.49 and 13.1 using $\sigma = 0.05$ for ITAE, ISDU and the settling time respectively. These results meet expectations since it is well known that as the level of noise in the measured value of the controlled variable increases the controller is detuned. In comparison to Case Study 2, the noise levels considered did not affect the robustness of the controller tuning parameters. However, a filter should be used to reduce the noise before the measurement is sent to the controller.

Figure 10 show the controlled and measured variable responses to a step change in the set point using the optimal tuning parameters obtained for both noise levels. The results show that the multi-objective optimization strategy used was capable of identifying tuning parameters that provided a rapid smooth response in the controlled variable without excessive control action in the presence of both parameter variation and noise.

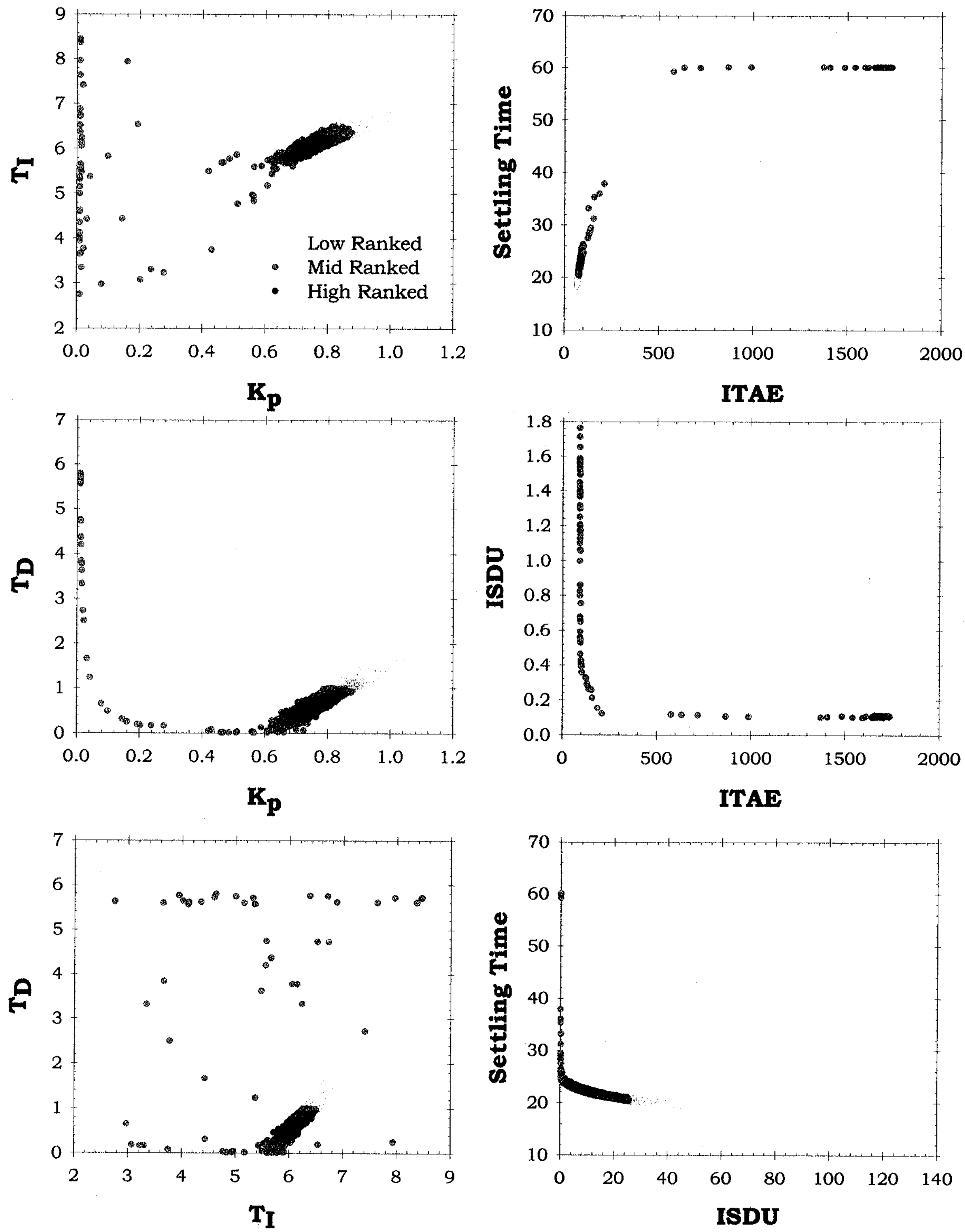

Figure 8 – Decision space and Pareto domain developed for a PID controller showing the effect of variance in the process model parameters and measurement noise with $\sigma = 0.01$ on controller design.

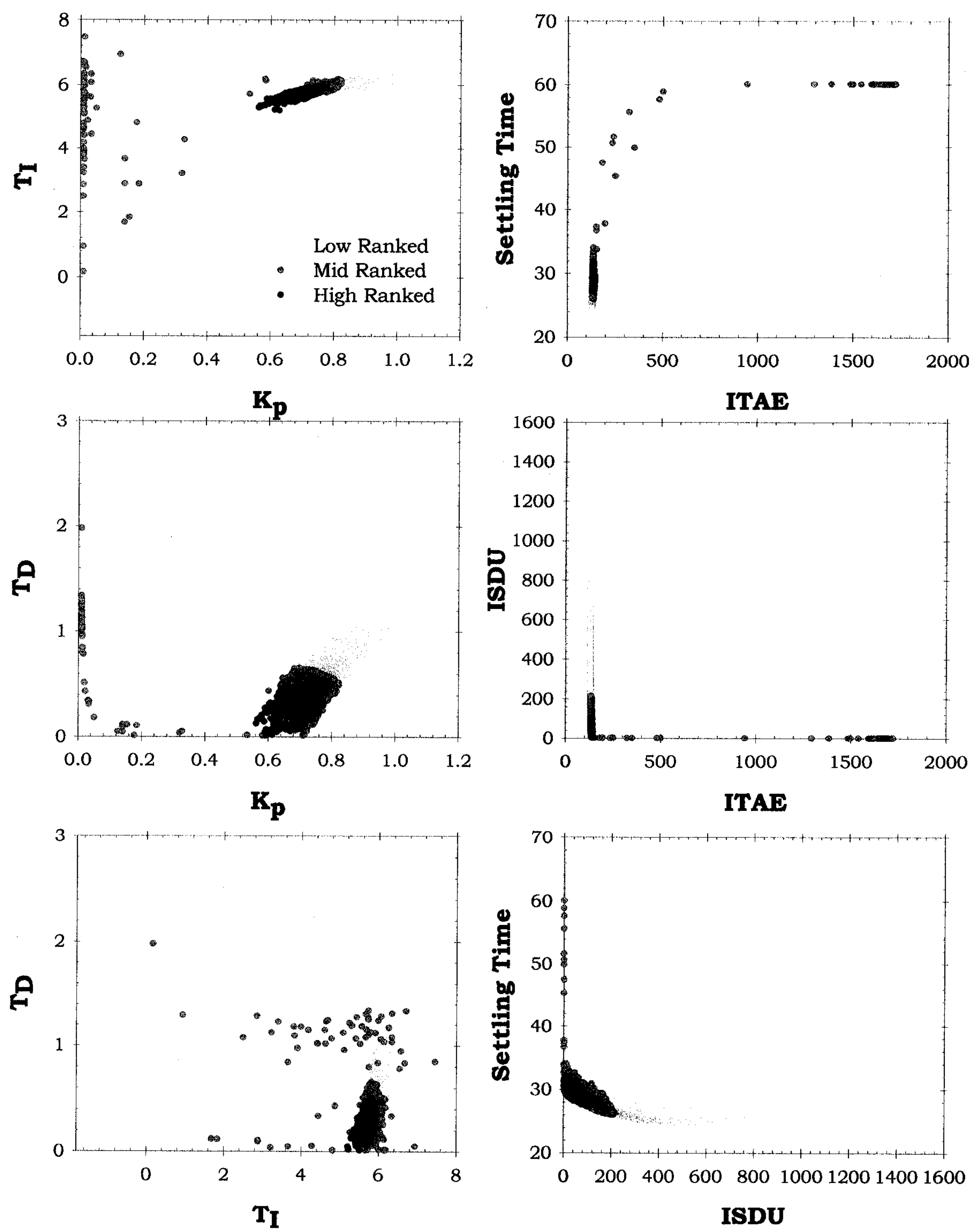

Figure 9 – Decision space and Pareto domain developed for a PID controller showing the effect of variance in the process model parameters and measurement noise with $\sigma = 0.05$ on controller design.

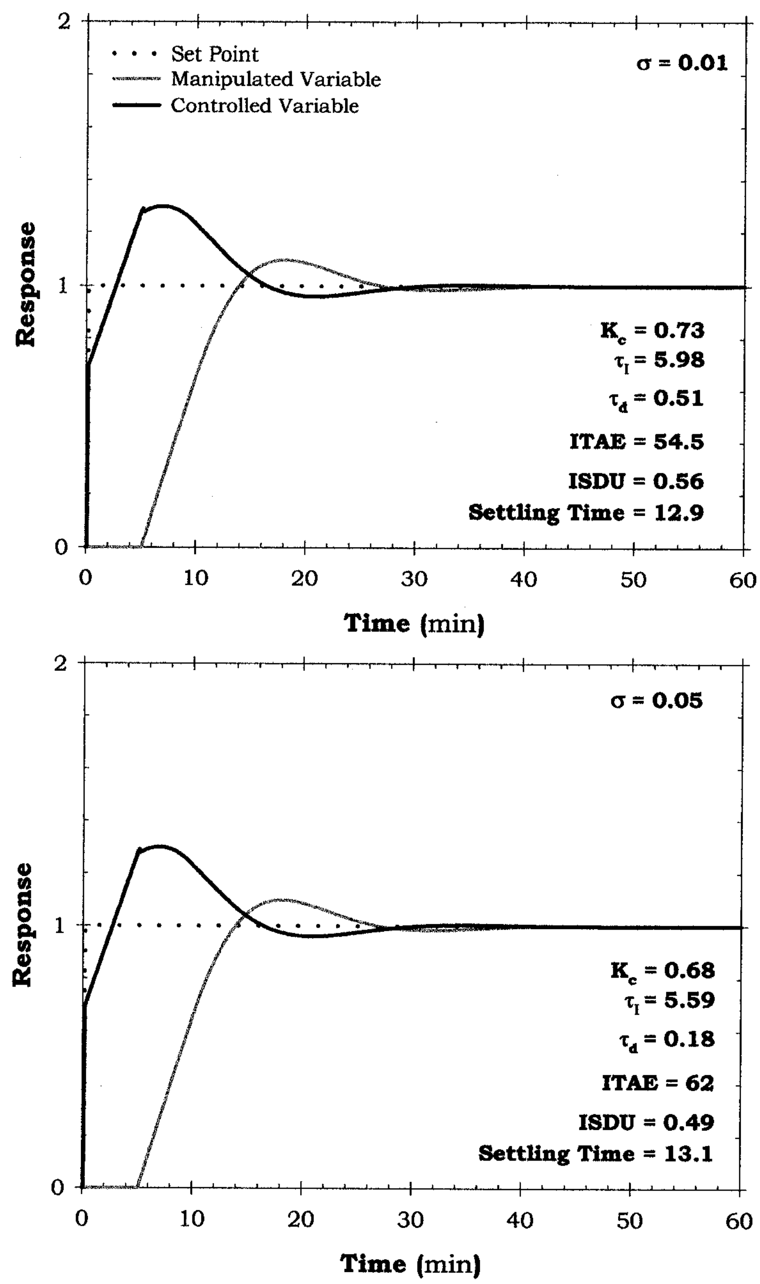

Figure 10 - Controlled and manipulated variable responses to a step change in the set point for the optimal point in the Pareto domains generated using measurement noise levels with $\sigma = 0.01$ and $\sigma = 0.05$, respectively.

The tuning parameters obtained using the Ciancone correlations for a fractional dead time of 0.5 were 0.687, 6.72, and 0.87 for K_c, T_I, and T_d, respectively. While the corresponding values for the performance measures were 60.5, 0.49, and 16.6 for the ITAE, ISDU and the settling time respectively. Comparing the results obtained for a noise level of $\sigma = 0.01$ to the Ciancone correlations, which were developed using a uniform distribution with a standard deviation of ±2.2% of scale (Marlin, 2000), the ITAE was higher using the Ciancone tuning parameters because of the persistent error experienced by the controlled variable, which accounts for the longer settling time. However, there was no overshoot in the controlled variable.

Developing Tuning Correlations Using Multi-objective Optimization

Figures 11and 12 present the tuning correlations developed for the PI and PID controllers using the proposed multi-objective optimization strategy for set point change. The results have been compared to the Rovira tuning relations for minimizing the ITAE and the Ciancone correlations, which minimizes the IAE. The results include the tuning correlations for the nominal plant, the nominal plant with variation in the model parameters, and the nominal plant with both model parameter variation and measurement noise with standard deviations $\sigma = 0.01$ and $\sigma = 0.05$. The formulas developed for the ITAE are empirical and were developed for a θ/τ range $\in [0.1\text{-}1]$ (Smith and Corripio, 1985; Rovira, 1981). However, fraction dead time values past 0.7 were not considered since the function for calculating T_I tends to infinity at 0.862. The Ciancone correlations were developed using a single objective optimization technique that minimized the IAE subject to constraints on the manipulated variable with a ±25% correlated change in the process model parameters and measurement noise, which was modeled using a uniform distribution with a standard deviation of ±2.2% of scale (Marlin, 2000).

Figures 11 and 12 compare the tuning correlations for both the PI and PID controllers developed using the proposed multi-objective optimization strategy with the Rovira ITAE tuning relations and the Ciancone correlations. The results show that both the Rovira ITAE tuning relations and the Ciancone correlations produced more aggressive controllers than those developed using the multi-objective optimization strategy presented for a unit step change in the value of the set point. In terms of the PI controller, the values for the tuning parameters using Rovira tuning formulas and the proposed multi-objective optimization strategy tended to converge as the dead time increased, while the Ciancone correlations diverged.

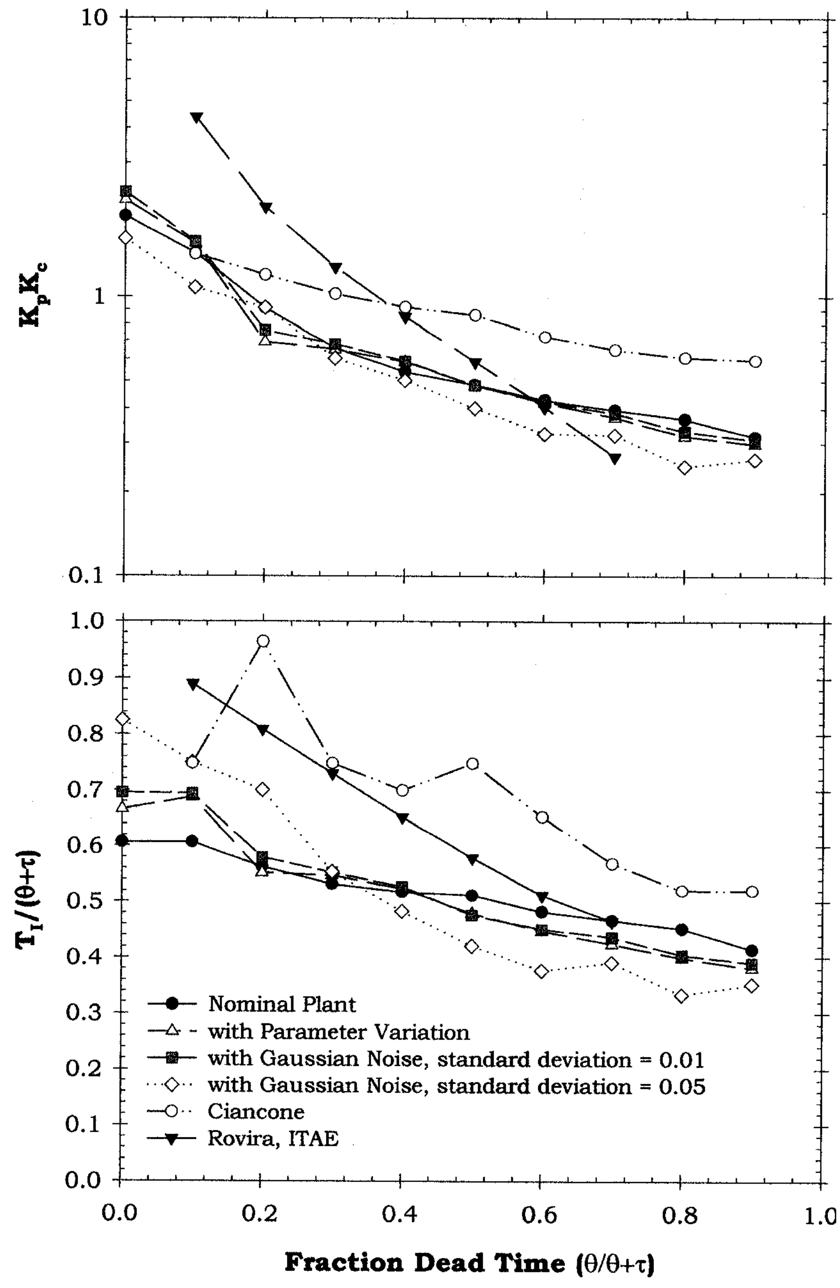

Figure 11 – Correlations developed using a multi-objective optimization strategy for dimensionless tuning constants in comparison with the Rovira ITAE tuning formulas and the Ciancone correlations for a PI controller subject to set point changes.

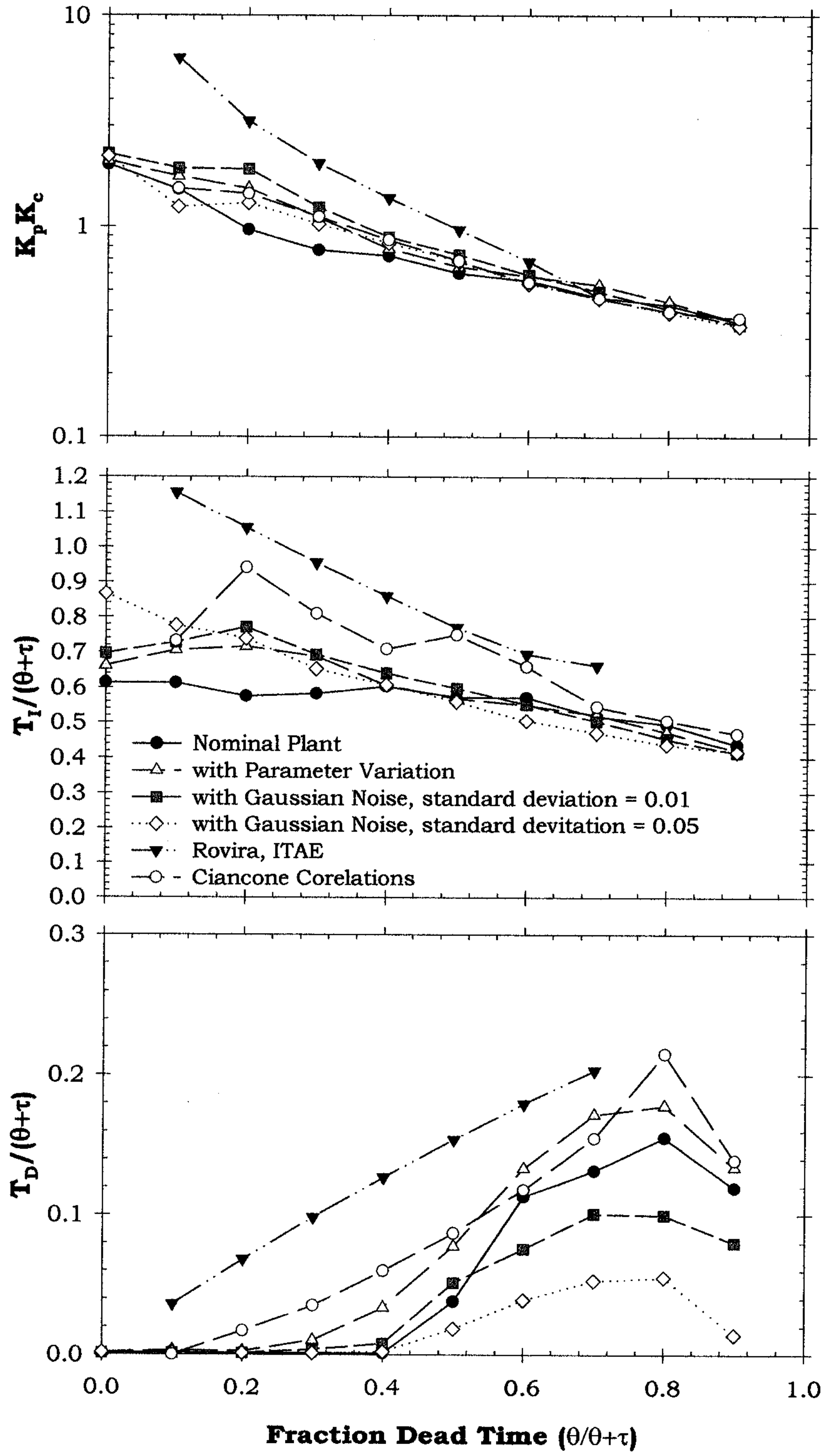

Figure 12 – Correlations developed using a multi-objective optimization strategy for dimensionless tuning constants in comparison with the Rovira ITAE tuning formulas and the Ciancone correlations for a PID controller subject to set point changes.

In terms of the PID controller, the values for the tuning parameters using all three techniques for K_p and T_I tended to converge as the dead time increased, while the values for T_d each followed the same trend.

Figure 13 present the tuning correlations developed for the PID controller using the proposed multi-objective optimization strategy for the disturbance response. The results include the tuning correlations for the nominal plant, the nominal plant with variation in the model parameters, and the nominal plant with model parameter variation and measurement noise using a standard deviation of $\sigma = 0.01$. The tuning correlations developed have been compared to the Lopez tuning relations for minimizing the ITAE for the disturbance response and the Ciancone correlations. The correlations developed using the proposed multi-objective optimization strategy for the nominal plant in Case Study 1were robust to variations in the model parameters for the first-order-plus-dead-time model when compared to the tuning parameters developed considering model parameter variations in Case Study 2. The correlations developed in both cases were similar for small dead times and converged as the fractional dead time increased. The addition of measurement noise caused the controller to be detuned, as expected. A comparison of the tuning correlations developed with the Lopez ITAE tuning formulas and the Ciancone correlations shows that the Lopez tuning formulas produced a more aggressive controller, while the Ciancone correlations produced a less aggressive controller. However, in each case the values for K_p and T_I tended to converge as the dead time increased, while the values for T_d followed the same trend.

The multi-objective optimization correlations demonstrate that for small measurement noise the resulting correlations were similar to the correlations where only model parameter variations were considered. As the level of noise increases the controller was detuned as expected. The results also show that tuning correlations developed using this optimization strategy were robust to model parameter variation and measurement noise. The trends in the correlations also followed expectations and, as such, are comparable to tuning correlations developed using traditional techniques. For example, K_c decreased as the dead time increased due to longer settling time for both the PI and PID controllers.

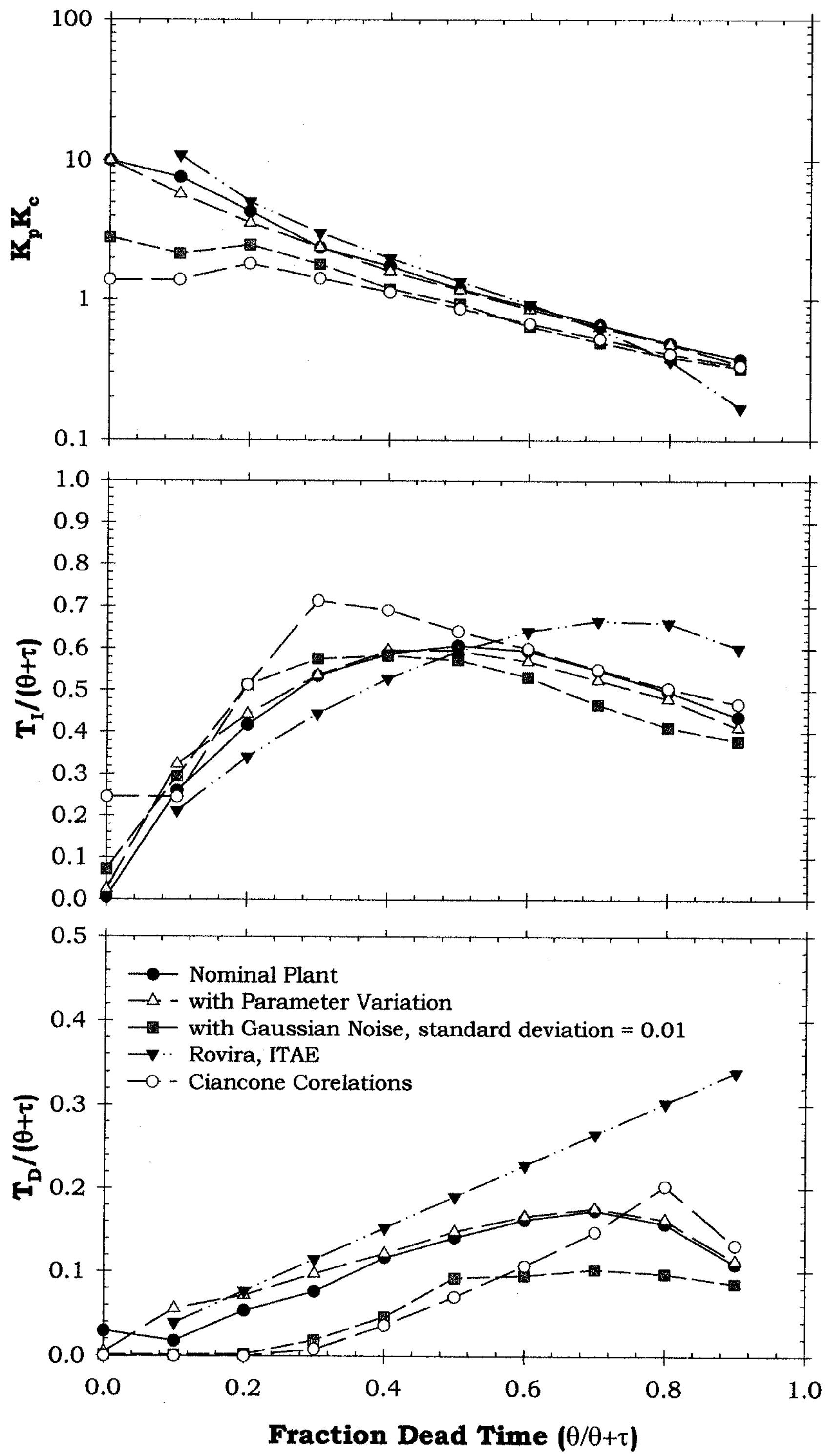

Figure 13 – Correlations developed using a multi-objective optimization strategy for dimensionless tuning constants in comparison with the Lopez ITAE tuning formulas and the Ciancone correlations for a PID controller subject to disturbance changes.

Conclusions

The focus of this paper was to demonstrate the robustness of a multi-objective optimization strategy by demonstrating its ability to develop robust PI and PID controller tuning correlations for both set point changes and disturbance response. This optimization strategy generated the required Pareto domains, by simultaneously minimizing three controller performance criteria, namely, the ITAE, the ISDU and the settling time, and classified the domain using Net Flow, a method that incorporates the decision-maker into the optimization routine.

The correlations developed using this optimization strategy were compared to the Rovira ITAE tuning formulas for set point changes, the Lopez ITAE tuning formulas for disturbance changes and the Ciancone correlations in order to demonstrate its robustness in developing tuning parameters for PI and PID controllers. Although tuning correlations only provide initial values for the controller tuning parameters the results show that the Dual Population Evolutionary Algorithm coupled with the Net Flow method was successful at identifying tuning parameters for the PI and PID controllers, by simultaneously optimizing multiple conflicting design objectives, that were conservative but robust to model parameter variation and measurement noise, and provide an alternative to traditional tuning methods.

References

Brans, J., P., Mareschal, B., Vincke, P. H., PROMETHEE: A New Family of Outranking Methods in Multicriteria Decision Making, *Operations Research*, Vol. 84, pgs. 477-498, 1984

Chen, B., Cheng, Y., and lee, C., A Genetic Approach to Mixed H_2/H_∞ Optimal PID Control, *Control Systems*, Vol. 15, No. 5, pgs. 51-60, 1995

Ciancone, R., and Marlin, T., Tune Controllers to Meet Plant Objectives, *Control*, Vol. 5, pgs. 50-57, 1992

Coello Coello, C. A., A Comprehensive Survey of Evolutionary-Based Multi-objective Optimization Techniques, *Knowledge and Information Systems*, Vol. 1, pgs. 269-308, 1999

Deb, K., Multi-objective Optimization Using Evolutionary Algorithms, Wiley, NY, 2001

Derot B., Gareau J., Kiss L. N., Martel J. M., The Solver of Volvox Multicriteria Table. Multicriteria Analysis, Climaio J., (ed) Springer-Verlag, pgs. 113-126, 1997

Doumpos, M., and Zopounidis, Multicriteria Decision Aid Classification Methods, *Applied Optimization*, Vol. 3, Kluwer Academic, 2002.

El-Kady, M.,M., Salim, M. S., and El-Sagheer, Numerical Treatment of Multiobjective Optimal Control Problems, *Automatica*, Vol. 39, pgs. 47-55, 2003

Fonseca, C. M., and Fleming, P. J., On Performance Assessment and Comparison of Stoichastic Multiobjective optimizers, *Proceedings of Parrallel Problem Solving from Nature IV (PPSN-IV)*, pgs. 584-59, 1996

Halsall-Whitney, H., Taylor, D., Thibault, J., Multicriteria Optimization of Gluconic Acid Production Using Net Flow, *Bioprocess and Biosystems Engineering*, Vol. 25, No. 5, pgs. 299-307, 2003

Herreros, A., Baeyens, E., Peran, J. R., MRCD: A Genetic Algorithm for Multi-objective Robust Control Design, *Engineering Applications of Artificial Intelligence*, Vol. 15, pgs. 285-301, 2002

Hu, A., Salcudean, S. E., and Loewen, P. D., A Numerical Solution of Multiobjective Control Problems, *Optimal Control Applications and Methods*, Vol. 19, pgs. 411-422, 1998

Kang, D., and Bien, Z., Design of Multiobjective Satisfactory Fuzzy Logic Controller UsingReinforcement Learning, *International Journal of Fuzzy Systems*, Vol. 2, No. 2, 2000

Kim, J., Moon, Y., and Zeigler, B. P., Designing Fuzzy Net Controllers Using Genetic Algorithms, *IEEE Control Systems Magazine*, Vol. 15, Iss. 3, pgs 62-72, 1995

Krishnakumar, K., and Goldberg, D. E., Control System Optimization Using Genetic Algorithms, Journal of Guidance, Control and Dynamics, Vol. 15, No. 3, pgs 735-740, 1992

Kristinsson, K., and Dumont, G. A., System Identification and Control Using Genetic Algorithms, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 22, No. 5, pgs 1033-1046, 1992

Liu, G.P., Yang, J. B., Whidborne, J. F., Multiobjective Optimisation and Control, *Research Studies Press*, England, 2003

Lopez, A. M., Murrill, P. W., Smith, C. L., Controller Tuning Relationships Based on Integral Performance Criteria, *Instrumentation Technology*, Vol. 14, pg. 57

Man, K., Tang, K. S., and Halang, W. A., Genetic Algorithms for Control and Signal Processing, Springer, Berlin, 1997

Marrison, C. I., and Stengel, R. F., Robust Control System Design Using Random Search and Genetic Algorithms, *IEEE Transactions on Automatic Control*, Vol. 42, No. 6, pgs. 835-839, 1997

Marlin, T. E., Process Control: Designing Processes and Control Systems for Dynamic Performance, 2nd ed., McGraw Hill, NY, 2000

Oliveira, P., Sequeira, J., and Sentieiro, J., Selection of Controller Parameters Using genetic Algorithms, *Engineering Systems with Intelligence: Concepts, Tools, and Applications*, Kluwer Academic Publishers, pgs. 431-438, 1991

Perrin, E., Mandrille, A., Oumoun, M., Fonteix, C., Marc I., Optimisation Globale par Stratégie d'Évolution: Technique Utilisant la Génétique des Individus Diploïdes. *RAIRO-Recherche Opérationnelle*, Vol. 31, pgs.161-201, 1997

Porter, B., and Jones, A. H., Genetic Tuning of PID Controllers, *Electronics Letters*, Vol., 28, No. 9, pgs. 843-844, 1992

Rangan, S., and Poolla, K., Weighted Optimization for Multi-objective Full-information Control Problems, *Systems and Letters*, vol. 31, pgs. 207-213, 1997

Rovira, D. E., Marari, M., and Skogestad, S., Internal Model Control for PID Controller Design, *I&EC Process Design and Development*, Vol. 25, pg. 252, 1981

Roy, B., ELECTRE III: Algorithme de Classement Basé sur une résentation Floue des Préférences en presence de Critères Multiples. Cahiers du CERO, Vol. 20, pgs.3-24, 1978

Scarelli, A., and Narula, C. N., A Multicriteria Assignment Problem, *Journal of Multi-Criteria Decision Analysis*, Vol. 11, pgs 65-74, 2002

Schubert, W. M., and Stengel, R. F., Parallel Synthesis of Robust Control Systems, *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 6, pgs. 701-706, 1998

Smith, C. A., and Corripio, A. B., Principles and Practice of Automatic Process Control, 2nd ed., Wiley, 1997

Thibault, J., Taylor, D., Fonteix, C., Multicriteria Optimization for the Production of Gluconic Acid, *Proceedings of the 8th International Conference on Computer Applications in Biotechnology: Modeling and Control of Biological Processes*, IFAC, Quebec, pgs. 24-27 2001;

Triantaphyllou, E., Multicriteria Decision Making Methods: A Comparative Study. *Applied Optimization Series*, Kluwer Academic Publishers, 2000

Visioli, A., Optimal Tuning of PID Controllers of Integral and Unstable Processes, *IEE Process Control Theory and Applications*, Vol 48, pge. 180, 2001

Vlachos, C., Williams, D., and Gomm, J. B., Genetic Approach to Decentralized PI Controller Tuning for Multivariable Processes, *IEE Proceedings – Control Theory and Applications*, Vol. 146, No. 1, pgs 58-64, 1999

Wang, P. and Kwok, D. P., Auto-tuning of Classical PID Controllers Using an Advanced Genetic Algorithm, *International Conference on Industrial Electronics, Control, Instrumentation and Automation*, IECON 92, Vol. 3, pgs. 1224-1229, 1992

Ziegler, J. G., and Nichols, N. B., Optimum Settings for Automatic Controllers, *Transactions ASME*, Vol. 64, pg. 759, 1942

Chapter 5

Conclusions

Conclusions

This research focused on the development and application of a multi-objective optimization strategy for selecting optimum operating conditions for chemical processes and for determining optimum tuning parameters for controllers.

In Paper 1, Multi-objective Optimization for Chemical Processes and Controller Design: Approximating and Optimizing the Pareto Domain, the robustness of three methods for approximating the Pareto domain was studied using twelve test cases and two engineering problems. Three of these test cases were presented along with the engineering problems. A multi-objective optimization strategy that generates a Pareto domain, using the Dual Population Evolutionary Algorithm, and classifies the entire domain using Net Flow, a technique that incorporates the knowledge of an expert into the optimization routine, was recommended. The strength of this optimization strategy lies in its ability to simultaneously optimize multiple conflicting objectives, and provide valuable information about the zone of possible solutions. This strategy was used in subsequent papers.

In Paper 2, Multicriteria Optimization of Gluconic Acid Production Using Net Flow, the optimization strategy proposed in Paper 1 was used to determine optimal operating conditions for the production of gluconic acid, which resulted in an acceptable compromise between the conflicting objectives. Further, the Pareto domain provided the expert with more immediate information about the relationship between the process inputs, the objective criteria, and the zone of possible solutions than that of traditional optimization algorithms.

In Paper 3, Multi-objective PID Controller Design for First Order Processes with Dead Time, the Dual Population Evolutionary Algorithm coupled with the Net Flow method was successful in identifying initial tuning parameters for PI and PID controllers, by simultaneously optimizing multiple conflicting design objectives, which is common in controller design, that were robust to model parameter variation and measurement noise, and comparable to traditional tuning methods.

Chapter 6

Recommendations

Recommendations

The development and application of multi-objective optimization techniques is one of the most promising and least explored area of optimization in the literature. Current research has demonstrated its ability to extend the current scope of optimization in engineering by addressing the shortcomings of traditional single objective criterion techniques. In terms of this research, the following two of recommendations are made.

1. First, one of the issues facing the evolutionary algorithms studied in this work is the need to specify the number of data points required to describe the Pareto domain. Since some simulations could take days, therefore, the ability for the algorithms to determine on their own when the Pareto domain has been defined, and then end the simulation, would be an asset.

2. Papers 1 and 2 proved that the proposed multi-objective optimization technique is robust in theory. The next step would be to prove it in reality.

Appendices

Appendix A

A.1 Standard Test Cases

The following are twelve standard test cases identified in the literature were used to evaluate the performance and efficiency of the multi-objective optimization algorithms studied in this work. The results of test case models 3, 6, 7, 9, and 11 were presented in Paper 1.

Test Case Model 1

Test case model 1 is a 2×2 multi-objective optimization problem that has a convex shaped Pareto domain with a non-uniform and dispersed decision space. The ranges for the inputs are $X_1 \in [-50 - 50]$ and $X_2 \in [-50 - 50]$ (Deb, 2001).

$$\begin{aligned} &Min\ f_1(X_1, X_2) = (X_1)^2 + (X_2)^2 \\ &Min\ f_2(X_1, X_2) = (X_1 + 2)^2 + (X_2)^2 \end{aligned} \quad \text{.......... (1)}$$

Test Case Model 2

Test case model 2 is a 2×2 multi-objective optimization problem that has a convex shaped Pareto domain with a relatively uniform decision space. The ranges for the inputs are $X_1 \in [0.1 - 1]$ and $X_2 \in [0 - 5]$ (Deb, 2001).

$$\begin{aligned} &Min\ f_1(X_1, X_2) = X_1 \\ &Min\ f_2(X_1, X_2) = \frac{1 + X_2}{X_1} \end{aligned} \quad \text{.......... (2)}$$

Test Case Model 3

Test case model 3 is a 2×2 multi-objective optimization problem that has a concave shaped Pareto domain with a relatively uniform decision space. The ranges for the inputs are $X_1 \in [0.1 - 1]$ and $X_2 \in [0 - 5]$ (Deb, 2001).

$$Max\ f_1(X_1,X_2)=1.1-X_1$$
$$Max\ f_2(X_1,X_2)=60-\frac{1+X_2}{X_1} \quad (3)$$

Test Case Model 4

Test case model 4 is a 2×2 multi-objective optimization problem that has a convex shaped Pareto domain with a uniform and well defined decision space. The ranges for the inputs are $X_1 \in [0-5]$ and $X_2 \in [0-3]$ (Deb, 2001).

$$Min\ f_1(X_1,X_2)=4(X_1)^2+4(X_2)^2$$
$$Min\ f_2(X_1,X_2)=(X_1-5)^2+(X_2-5)^2 \quad (4)$$

Test Case Model 5

Test case model 5 is a complex 3×2 multi-objective optimization problem that has a non-convex outputs space containing three distinct disconnected Pareto domains with a dispersed decision space. The ranges for the inputs are $X_1 \in [-5-5]$, $X_2 \in [-5-5]$ and $X_3 \in [-5-5]$ (Kursawe, 1990).

$$Min\ f_1(X_1,X_2,X_3)=\sum_{i=1}^{3}\left[-10\exp(-2.0\sqrt{(X_i)^2+(X_{i+1})^2}\right]$$
$$Min\ f_2(X_1,X_2,X_3)=\sum_{i=1}^{3}\left[|X_i|^{0.8}+5\sin(X_i)^3\right] \quad (5)$$

Test Case Model 6

Test case model 6 is a complex 2×2 multi-objective optimization problem that has a non-convex output space containing two distinct disconnected Pareto domains with a disconnected but well defined decision space. The ranges for the inputs are $X_1 \in [-\pi - \pi]$ and $X_2 \in [-\pi - \pi]$ (Poloni, 2000).

$$Min\ f_1(X_1, X_2) = 1 + (A_1 - B_1)^2 + (A_2 - B_2)^2$$
$$Min\ f_2(X_1, X_2) = (X_1 + 3)^2 + (X_2 + 1)^2$$

where .... (6)

$$A_1 = 0.5sin(1) - 2cos(1) + sin(2) - 1.5cos(2)$$
$$A_2 = 1.5sin(1) - cos(1) + 2sin(2) - 0.5cos(2)$$
$$B_1 = 0.5sin(X_1) - 2cos(X_1) + sin(X_2) - 1.5cos(X_2)$$
$$B_2 = 1.5sin(X_1) - cos(X_1) + 2sin(X_2) - 0.5cos(X_2)$$

Test Case Model 7

Test case model 7 is a complex 2×3 multi-objective optimization problem that has a well-defined Pareto domain with a disconnected but well defined decision space. The ranges for the inputs are $X_1 \in [-3 - 3]$, $X_2 \in [-3 - 3]$ (Viennet, 1995).

$$Min\ f_1(X_1, X_2) = 0.5\left[(X_1)^2 + (X_2)^2\right] + \sin\left[(X_1)^2 + (X_2)^2\right]$$
$$Min\ f_2(X_1, X_2) = \frac{(3X_1 - 2X_2 + 4)^2}{8} + \frac{(X_1 - X_2 - 1)^2}{27} + 15 \quad \text{........................ (7)}$$
$$Min\ f_2(X_1, X_2) = \frac{1}{(X_1)^2 + (X_2)^2 + 1} - 1.1\exp\left[-\left((X_1)^2 + (X_2)^2\right)\right]$$

Test Case Model 8

Test case model 8 is a 2×2 multi-objective optimization problem that has a non-convex shaped, non-concave shaped and discontinuous Pareto domain with a dispersed decision space. The functions are multi-modal; therefore there are multiple global optima. The ranges for the inputs are $X_1 \in [-5 - 5]$ and $X_2 \in [-5 - 5]$ (Silva and Biscaia, 2003).

$$
\begin{aligned}
Max\ f_1(X_1,X_2) &= 20+(X_1)^2-10\cos(2\pi X_1)+(X_2)^2 \\
&\quad -10\cos(2\pi X_2) \\
Max\ f_2(X_1,X_2) &= \exp\left\{-0.3\left[(X_1-2)^2+(X_2-2)^2\right]\right\} \\
&\quad +2\exp\left\{-0.3\left[(X_1+1)^2+(X_2-2)^2\right]\right\}
\end{aligned} \qquad \text{.......... (8)}
$$

Test Case Model 9

Test case model 9 is a 2×2 multi-objective optimization problem that has a non-convex shaped, non-concave shaped and discontinuous Pareto domain with a well-defined decision space. The functions are multi-modal; therefore there are multiple global optima. The ranges for the inputs are $X_1 \in [-5 - 5]$ and $X_2 \in [-5 - 5]$ (Silva and Biscaia, 2003).

$$
\begin{aligned}
Max\ f_1(X_1,X_2) &= 20+(X_1)^2-10\cos(2\pi X_1)+(X_2)^2 \\
&\quad -10\cos(2\pi X_2) \\
Max\ f_2(X_1,X_2) &= -100\left[(X_1)^2-(X_2)^2\right]-\left[1-(X_2)^2\right]^2
\end{aligned} \qquad \text{.......... (9)}
$$

Test Case Model 10

Test case model 10 is a 2×2 multi-objective optimization problem that has a concave shaped Pareto domain with a well-defined decision space. The functions are multi-modal; therefore there are multiple global optima. The ranges for the inputs are $X_1 \in [-5 - 5]$ and $X_2 \in [-5 - 5]$ (Silva and Biscaia, 2003).

$$
\begin{aligned}
Max\ f_1(X_1,X_2) &= 20+(X_1)^2-10\cos(2\pi X_1)+(X_2)^2 \\
&\quad -10\cos(2\pi X_2) \\
Max\ f_2(X_1,X_2) &= \sin\left(\frac{X_1}{\pi}\right)^2 \sin\left(\frac{X_2}{\pi}\right)^2 \left[|X_1|+|X_2|+0.1X_1+0.1X_2\right] \\
&\quad -0.03(X_1)^2-0.03(X_2)^2
\end{aligned} \qquad \text{.......... (10)}
$$

Test Case Model 11

Test case model 11 is a 3×3 multi-objective optimization problem that has a convex shaped Pareto domain with a uniform and well defined decision space. The ranges for the inputs are $X_1 \in [-2 - 2]$, $X_2 \in [-2 - 2]$ and $X_3 \in [-2 - 2]$ (Viennet *et al.*, 1996).

$$\begin{aligned} &Min\ f_1(X_1, X_2, X_3) = (X_1)^2 + (X_2 - 1)^2 \\ &Min\ f_2(X_1, X_2, X_3) = (X_1)^2 + (X_2 + 1) + 1 \\ &Min\ f_3(X_1, X_2, X_3) = (X_1 - 1)^2 + (X_2)^2 + 2 \end{aligned} \quad \text{.........(11)}$$

Test Case Model 12

Test case model 12 is a 2×2 multi-objective optimization problem that has a convex shaped Pareto domain with a uniform and well defined decision space. The ranges for the inputs are $X_1 \in [-100 - 100]$ and $X_2 \in [-100 - 100]$ (Shim and Suh, 2001).

$$\begin{aligned} &Min\ f_1(X_1, X_2) = (X_1)^2 + (X_2)^2 \\ &Min\ f_2(X_1, X_2) = (X_1 - 1)^2 + (X_2 - 2)^2 \end{aligned} \quad \text{.........(12)}$$

A.2 Optimization Objectives Studied

Figure 1 represents the multi-input, multi-output optimization strategies used for developing the Pareto domain and the decision space for each of the test cases described previously. The results were compared to the literature in order to evaluate the performance of the multi-objective optimization techniques studied. The simulated results are contained on the companion disk provided in an MS Access database file called <u>TestCases.</u>

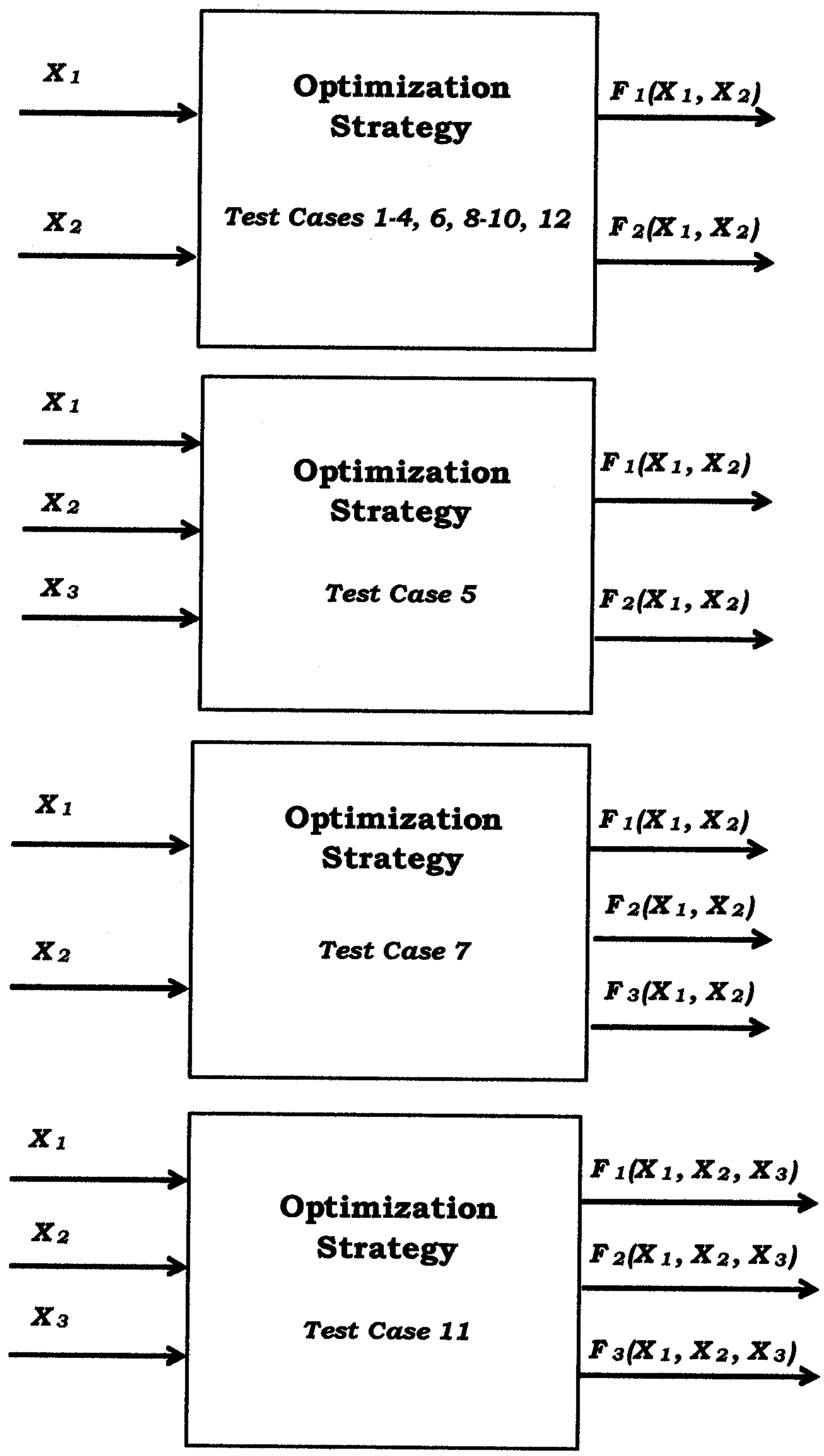

Figure 1 - Objectives used during the development of the Pareto domain and optimization of Test Case 1-12.

References

Deb, K., Multi-objective Optimization Using Evolutionary Algorithms, Wiley, NY, 2001

Kursawe, F., A Variant of Evolution Strategies for Vector Optimization, *Parallel Problem Solving From Nature I (PPSN-I)*, pgs 193-197, 1990

Poloni, C., Giurgevich, A., Onesti, L., and Pediroda, V., Hybridization of a Multi-objective Genetic Algorithm, a Neural Network, and a Classical Optimizer for Complex Design problem in Fluid Dynamics, *Computer Methods in Applied Mechanics and Engineering*, Vol. 186, (2-4), pgs. 403-420, 2000

Shim, M., Suh, M., Furukawa, T., Yagawa, G., and Yoshimura, S., Pareto-based Continuous Evolutionary Algorithms for Multi-objective Optimization, *Engineering Computations*, Vol. 19, No. 1, pgs. 22-48, 2002

Silva, C. M., and Biscaia, E. C., Genetic Algorithm Development for Multi-objective Optimization of Batch Free-radical Polymerization Reactors, *Computers and Chemical Engineering*, Vol 27, pgs 1329-1344, 2003

Viennet, R., Fonteix, C., Marc, I., Multicriteria Optimization Using a Genetic Algorithm for Determining a Pareto Set, *International Journal of Systems Science*, Vol. 27, No. 2, pgs. 255-260, 1995