SCHOOL OF CHEMICAL ENGINEERING

NONLINEAR CONTROL TECHNIQUES IN ALUMINA REFINERIES

LAP CHI TO, B. ENG. (HONS.)

THIS DISSERTATION IS PRESENTED AS

PART OF THE REQUIREMENTS FOR

THE AWARD OF THE DEGREE OF

DOCTOR OF PHILOSOPHY

OF

CURTIN UNIVERSITY OF TECHNOLOGY

JULY 1996

SUMMARY

Nonlinearities exist in all process control systems. The use of linear control techniques is valid only in a narrow range of operation. Therefore, in this thesis, multivariable nonlinear control techniques are considered. The target process is the single effect evaporative process of the liquor burning unit in Alcoa's alumina refinery in Kwinana and the proposed triple effects unit in the Wagerup refinery. Two types of nonlinear control strategies using differential geometry were studied, namely, the input output linearization (Kravaris and Soroush, 1990) and the input state linearization (Hunt et al, 1983a). The research has successfully demonstrated the superiority and simplicity of the nonlinear controller through simulations and plant implementations. An integrated software package using MAPLE V.3 as the computing environment was developed to automate the solution algorithms and to graphically simulate the closed loop dynamics of different processes using the two nonlinear control strategies.

The issue of robustness of the nonlinear controller was addressed by developing a procedure called uncertainty vector adjustment. The effectiveness of the new strategy was successfully demonstrated on the simulated liquor burning process. Furthermore, the stability of the adjustment technique was proved and its theoretical bounds were established using Lyapunov function analysis.

A comparative study of geometric nonlinear filter and extended Kalman filter was conducted to reduce the requirement of full state feedback necessary for nonlinear control using either input output linearization or input state linearization. The simulation of the single effect evaporation unit of the liquor burning process showed that the geometric nonlinear filter is superior to the extended Kalman filter in terms of nonlinear tracking performances.

The plant trials of the input output linearization in Alcoa's Kwinana alumina refinery demonstrated the practicability and feasibility of implementing nonlinear control in an

industrial setting and also fostered a closer gap between academia and industry. The trials established guidelines for implementing a global linearizing controller on site, including conversion of the relevant constraints and the output of an industrial proportional and integral controller to the equivalent proportional and integral action required by the nonlinear controller. The results showed that the performance of the nonlinear controller was better than the current linear controller on site in terms of responsiveness and resistance to disturbances. Hence, the nonlinear control strategy enables the process to settle faster.

All in all, efforts have been made in this thesis to minimise the use of abstract mathematical language and, in some cases, simplify the language so that nonlinear control theory can be understood by a wider range of audience, especially industrial practitioners. It is hoped that the insights provided in the dissertation will encourage more industrial implementations of nonlinear controllers and forge more interaction to close the widening gap between academic and industrial practice in process control.

Keywords: *nonlinear control, differential geometry, symbolic algebra, evaporator process, uncertainty vector adjustment, geometric nonlinear filter*

BRIEF BIOGRAPHY OF THE AUTHOR

Lap Chi (Benjamin) TO received the Bachelor of Engineering (Chemical) with First Class Honours from the Curtin University of Technology in 1992 and commenced his work on this PhD thesis in February of the same year. During the course of his undergraduate study, he received several awards, including the Medal for the Best Performance in the Chemical Engineering Course by a Graduating Student in 1992, and scholarships including the Overseas Postgraduate Research Scholarship in 1993. He currently holds the position of a Process Control Engineer with Bristol Digital Systems Australasia Pty Ltd, Perth.

Journal Publications arising from this Thesis

To, L. C., Tadé, M. O., Kraetzl, M. and Le Page, G. P. (1995a) Nonlinear Control of a Simulated Industrial Evaporation Process. *Journal of Process Control*, **5**(3), 173-182.

To, L. C., Tadé, M. O., and Kraetzl, M. (1995b) An Uncertainty Vector Adjustment for Process Modelling Error Compensation. To appear in *Journal of Process Control*.

To, L. C., Tadé, M. O. and Kraetzl, M. (1996a) An Integrated MAPLE Package for Nonlinear Control Studies. *Chemical Engineering Issue of the Institution of Engineers, Singapore*, **36**(3), 49-551.

To, L. C., Tadé, M. O. and Le Page, G. P. (1996b) Nonlinear Control of a Simulated Triple Effects Evaporation Process. In preparation.

To, L. C., Tadé, M. O. and Le Page, G. P. (1996c) Plant Trial of Input Output Linearization. In preparation.

Refereed Conference Publications arising from this Thesis

To, L. C., Tadé, M. O., Kraetzl, M., and Le Page, G. P. (1994) A Differential Geometric Approach to Control an Evaporation Process. *Proc. of Chemeca '94*, Perth, Australia, September, **2**, 735-742.

To, L. C., and Tadé, M. O. (1996d) An Uncertainty Vector Description for Process Modelling Error: A Lyapunov Function Analysis. *Proc. of APCChE 1996 Congress*, Taipei, Taiwan, March, **2**, 637-641.

To, L. C., Tadé, M. O., and Kraetzl, M. (1996e) Process Modelling Error Description Using an Uncertainty Vector. *Proc. of 1996 IChemE Research Event / Second European Conference for Young Researchers*, Leeds, UK, April, **2**, 729-731.

To, L. C. and Tadé, M. O. (1996f) A Chemical Engineering Application of Geometric Nonlinear Filter. *Proc. of Chemeca '96*, Sydney, Australia, September.

ACKNOWLEDGMENTS

The preparation of this thesis was surprisingly enjoyable because of the flashes of pleasure in seeing many results fall into place and the thought of the opportunity I would have at some stage to formally acknowledge the many people who made this endeavour possible.

I would like to express my sincerest gratitude to Dr Moses O. Tadé for his support and guidance throughout the course of this work. I was honoured and fortunate enough to be one of his PhD students. Any instances of originality, creativity and clarity in my research and exposition are but pale imitations of his shining example. I wish to acknowledge Dr Miro Kraetzl, as my co-supervisor, for his assistance in understanding modern algebra, in using MAPLE V.3 and in revising the text. In addition, special thanks are due to Mr Graham P. Le Page for his professional assistance and contribution during the plant trials.

The productive environment at Curtin University of Technology has fostered the success of this research. I am delighted to thank the School of Chemical Engineering for the consistent financial supports and all the staff in the School.

Most importantly I wish to thank my parents for their steady supports and encouragement . Finally, I dedicate this thesis to Linda who has been the best thing that ever happened to me. I could not have done it without her.

TABLE OF CONTENTS

Summary i

Brief Biography of The Author iii

Acknowledgments v

Table of Contents vi

List of Figures xv

List of Tables xviii

Nomenclature xxi

CHAPTER ONE INTRODUCTION

1.1 Why Nonlinear Control Theory 1-1

1.2 Brief History Of Nonlinear Control Theory 1-2

1.3 Motivations For This Study 1-3

1.4 Objectives And Contributions 1-4

1.5 Thesis Overview 1-4

CHAPTER TWO FEEDBACK LINEARIZATION AND SYMBOLIC COMPUTATIONS

2.1 Introduction To Feedback Linearization Theory 2-1

2.1.1 Full State Feedback Linearization 2-1

2.1.2 Partial State Feedback Linearization 2-2

2.2 Literature Research 2-3

2.2.1 Input State Linearization (I/S) 2-3

2.2.2 Input Output Linearization (I/O) 2-4

2.3 Mathematical Preliminaries For Differential Geometry 2-6

2.3.1 Vector Fields and Scalar Fields 2-6

2.3.2 Lie Derivative 2-7

2.3.3 Lie Bracket 2-7

2.4 Defining The Nonlinear Control System 2-8

2.4.1 Su-Hunt-Meyer (SHM) Transformation 2-9

2.4.1.1 Algorithm for Computing Kronecker Indices *2-10*

2.4.1.2 Criteria for Transformation *2-12*

2.4.1.3 Transformation of Variables *2-13*

2.4.1.4 Proposed Special Case of The SHM Transformation *2-16*

2.4.1.5 Summary of the Algorithm for The SHM Transformation *2-17*

2.4.1.6 Feedback Control by The SHM Transformation *2-17*

2.4.2 Input Output Linearization (I/O) 2-18

2.4.2.1 MIMO Globally Linearizing Control (GLC) Structure *2-20*

2.4.2.2 A Special Case of The Input Output Linearization: Generic Model Control *2-21*

2.4.2.3 Design Procedures of MIMOGLC Structure *2-22*

2.5 Symbolic Computations 2-23

2.5.1 What are Symbolic Computations? 2-23

2.5.2 MAPLE: The Introduction 2-24

2.5.3 Applications of Symbolic Computations in Process Control 2-25

2.6 Conclusions 2-27

CHAPTER THREE NONLINEAR CONTROL OF A SIMULATED SINGLE EFFECT EVAPORATOR PROCESS

3.1 Liquor Burning Process (LBP) 3-1

3.1.1 Process Description 3-1

3.1.2 Dynamic Modelling 3-2

3.1.2.1 Assumptions *3-3*

3.1.2.2 Dynamic Equations *3-4*

3.1.2.3 Selection of Input and Output Variables *3-5*

3.2 Simulation Results **3-6**

3.2.1 Open Loop Simulation Results 3-8

3.2.2 Closed Loop Simulation Results 3-8

3.2.2.1 Input Output Linearization versus Su-Hunt-Meyer Transformation *3-8*

3.2.2.2 Input Output Linearization versus Generic Model Control *3-10*

3.3 Discussions **3-13**

3.3.1 Local Linearization 3-13

3.3.2 Input Output Linearization and Generic Model Control 3-13

3.3.3 Su-Hunt-Meyer Transformation 3-14

3.4 Conclusions **3-15**

CHAPTER FOUR ROBUSTNESS STUDY OF NONLINEAR CONTROL THEORIES

4.1 Introduction **4-1**

4.2 Simulation Results On Robustness Property **4-2**

4.2.1 Input Output Linearization and Generic Model Control 4-2

4.2.2 Su-Hunt-Meyer Transformation 4-7

4.3 Discussions On Simulated Robustness Property **4-10**

4.3.1 Input Output Linearization, Generic Model Control and Linear Control 4-10

4.3.2 Su-Hunt-Meyer Transformation 4-10

4.4 Conclusions On Simulated Robustness Property **4-12**

CHAPTER FIVE ROBUST CONTROLLER SYNTHESIS USING UNCERTAINTY VECTOR ADJUSTMENT

5.1 Introduction **5-1**

5.2 Brief Literature Review On Modelling Errors **5-1**

5.3 Theoretical Development Of Uncertainty Vector Adjustment **5-3**

5.4 Theoretical Analysis Of Uncertainty Vector Adjustment **5-4**

5.4.1 Instability caused by the Derivative Estimation 5-4

5.4.2 Simulation of the Nominal Process 5-7

5.5 Algorithm For Uncertainty Vector Adjustment 5-8

5.5.1 Uncertainty Vector Adjustment and Input Output Linearization 5-9

5.5.2 Uncertainty Vector Adjustment and Su-Hunt-Meyer Transformation 5-12

5.5.3 Selection of the Uncertainty Vector Parameters 5-14

5.6 Simulation Results Using Uncertainty Vector Adjustment 5-15

5.6.1 Input Output Linearization 5-16

5.6.2 Su-Hunt-Meyer Transformation 5-25

5.7 Discussions On Robust Controller Synthesis Using Uncertainty Vector Adjustment 5-34

5.7.1 Case 1: An Increase in Temperature Factor for Input Output Linearization and Su-Hunt-Meyer Transformation 5-34

5.7.2 Case 2: An Increase in Temperature Factor and a Decrease in Feed Density 5-36

5.7.2.1 Input Output Linearization *5-36*

5.7.2.2 Su-Hunt-Meyer Transformation *5-37*

5.8 Conclusions On Robust Controller Synthesis Using Uncertainty Vector Adjustment 5-37

CHAPTER SIX LYAPUNOV FUNCTION ANALYSIS OF THE UNCERTAINTY VECTOR ADJUSTMENT

6.1 Introduction 6-1

6.2 Background 6-1

6.3 Brief Literature Review 6-2

6.3.1 Construction of a Lyapunov Function 6-2

6.3.2 Robust Controller Design by Lyapunov Functions 6-3

6.4 Mathematical Preliminary 6-4

6.4.1 Concepts of Stability 6-4

6.4.1.1 Local Asymptotic Stability *6-4*

6.4.1.2 Global Asymptotic Stability *6-4*

6.4.2 Lyapunov's Direct Method 6-5

6.4.2.1 Lyapunov Function *6-5*

6.5 Construction Of Lyapunov Functions Through Transformed Linear System 6-7

6.5.1 Proof of Theorem 6.3 for Input Output Linearization 6-8

6.5.2 Proof of Theorem 6.3 for Su-Hunt-Meyer Transformation 6-11

6.6 Theoretical Bounds For Uncertainty Vector And Modelling Error 6-13

6.6.1 Theoretical Bounds for the Uncertainty Vector Using Input Output Linearization 6-13

6.6.2 Theoretical Bounds for the Uncertainty Vector Using Su-Hunt-Meyer Transformation 6-17

6.6.3 Existence of the Bound for the Uncertainty Vector 6-20

6.6.4 Theoretical Bound for Modelling Errors 6-20

6.6.5 Theoretical Bound for Uncertainty Vector Parameters 6-21

6.6.6 Remarks 6-21

6.7 Simulation Results And Discussions 6-22

6.7.1 Construction of Lyapunov Functions 6-22

6.7.2 Determination of the Bound for the Uncertainty Vector 6-25

6.8 Conclusions 6-25

CHAPTER SEVEN SIMULATION AND CONTROL OF A TRIPLE EFFECTS EVAPORATOR

7.1 Introduction 7-1

7.2 Triple Effects Evaporator Design 7-2

7.2.1 Process Description 7-2

7.2.2 Dynamic Modelling 7-3

7.2.2.1 Assumptions *7-8*

7.2.2.2 Dynamic Equations *7-8*

7.2.3 Selection of Control Strategies 7-12

7.2.4 Proposed Input Output Linearization for a General Class of Process 7-12

7.2.4.1 Relative Order 7-13

7.2.4.2 Design Method 7-13

7.3 Simulation Results And Discussions 7-14

7.3.1 Specifications for the Simulation 7-14

7.3.2 Study of the Nominal Situation 7-17

7.3.3 Study of Robustness 7-25

7.3.3.1 Discussions on Robustness Study without Uncertainty Vector Adjustment 7-39

7.3.3.2 Discussions on Robustness Study Using Uncertainty Vector Adjustment 7-39

7.4 Further Test For Uncertainty Vector Adjustment 7-40

7.5 Conclusions 7-41

CHAPTER EIGHT USE OF STATE OBSERVERS IN NONLINEAR CONTROL

8.1 Introduction 8-1

8.2 Brief Literature Review On Geometric Nonlinear Filters 8-1

8.3 Geometric Nonlinear Filter 8-2

8.3.1 Transformation of Nonlinear Systems into Observer Form 8-3

8.3.2 Observer Design 8-5

8.3.3 Alternative Observer Design 8-6

8.4 Extended Kalman Filter 8-7

8.5 Simulation Results 8-9

8.5.1 Determination of Actual Plant States 8-9

8.5.2 Implementation of the Geometric Nonlinear Filter 8-10

8.5.3 Implementation of the Extended Kalman Filter 8-12

8.6 Discussions 8-12

8.7 Conclusions 8-15

CHAPTER NINE PLANT IMPLEMENTATION OF INPUT OUTPUT LINEARIZATION ON THE LIQUOR BURNING PROCESS

9.1 Introduction 9-1

9.2 Results and Discussions 9-2

9.2.1 PI Action in the Nonlinear Controller and Industrial PI Controller 9-2

9.2.2 Anti-reset Windup for PI Controller 9-4

9.2.3 Initialisation of PI Controller 9-5

9.2.4 Tuning of PI Controller 9-5

9.2.5 Nonlinear Controller Implementation Tests 9-7

9.2.5.1 Step Change in Product Flowrate *9-7*

9.2.5.2 Set Point Change in Density Control *9-10*

9.2.5.3 Discussions *9-13*

9.3 Conclusions 9-14

CHAPTER 10 CONCLUSIONS AND RECOMMENDATIONS

10.1 Conclusions 10-1

10.2 Recommendations 10-3

REFERENCES

References R-1

APPENDIX A A MAPLE PACKAGE FOR NONLINEAR CONTROL STUDY

A.1 Introduction **A-1**

A.2 Mathematical Preliminary **A-1**

A.3 Algorithms for Maple Procedures **A-2**

A.4 User Guide for Maple Procedures **A-5**

A.4.1 Descriptions of Procedures A-5

A.4.1.1 L - Lie Derivative *A-5*

A.4.1.2 io and ioloop - Input Output Linearization *A-5*

A.4.1.3 shm and shmloop - Su-Hunt-Meyer Transformation *A-8*

A.4.2 Other Options and Settings in Procedures ioloop and shmloop A-10

A.4.3 Other Conditions Required by Procedure ioloop and shmloop A-10

A.5 Illustrative Example to Use the Maple Procedures **A-10**

A.6 Discussions **A-12**

A.6.1 Single Effect Evaporator A-12

A.6.2 Triple Effects Evaporator A-12

A.7 Conclusions on Developed Maple Procedures **A-14**

APPENDIX B CALLING SEQUENCES FOR MAPLE PROCEDURES

B.1 Calling Sequences for Procedures io and ioloop **B-1**

APPENDIX C SAMPLE OUTPUTS FOR MAPLE PROCEDURES

C.1 Sample Output of Procedures io and ioloop **C-1**

C.2 Sample Output of Procedures shm and shmloop **C-4**

APPENDIX D MODEL EQUATIONS FOR TRIPLE EFFECTS EVAPORATOR

D.1 Model Equations for Triple Effects Evaporator D-1

APPENDIX E SIMULATION PROGRAM TRIIOUV AND SAMPLE OUTPUTS

E.1 Program *triiouv* for Simulation of Triple Effects Evaporator E-1

E.2 Sample Outputs of Program *triiouv* E-14

APPENDIX F LINEARIZED MODEL FOR TRIPLE EFFECTS EVAPORATOR

F.1 Linearized Model For Triple Effects Evaporator F-1

APPENDIX G LINEAR CONTROL STRATEGIES FOR LIQUOR BURNING PROCESS

LIST OF FIGURES

CHAPTER TWO FEEDBACK LINEARIZATION AND SYMBOLIC COMPUTATIONS

Figure 2.1: The Fundamental Structure of Full State Linearization 2-2

Figure 2.2: The Fundamental Structure of Partial State Feedback Linearization..... 2-2

Figure 2.3: Feedback Control by SHM Transformation 2-18

Figure 2.4: Multi-Input Multi-Output Globally Linearizing Control Structure..... 2-21

CHAPTER THREE NONLINEAR CONTROL OF A SIMULATED SINGLE EFFECT EVAPORATOR PROCESS

Figure 3.1: Flowsheet of Liquor Burning Process..... 3-2

Figure 3.2: Open Loop Dynamics of 2×2 LBP by Local Linearization and SHM 3-9

Figure 3.3: Closed Loop Dynamics of 2×2 LBP (Linear, I/O and SHM)..... 3-11

Figure 3.4: Closed Loop Dynamics of 2×2 LBP (I/O and GMC) 3-12

CHAPTER FOUR ROBUSTNESS STUDY OF NONLINEAR CONTROL THEORIES

Figure 4.1: Closed Loop Dynamics of 2×2 LBP using Local Linearization 4-3

Figure 4.2: Closed Loop Dynamics of 2×2 LBP using GMC 4-4

Figure 4.3: Closed Loop Dynamics of 2×2 LBP using I/O..... 4-5

Figure 4.4: Closed Loop Dynamics of 3×3 LBP using SHM Transformation..... 4-11

CHAPTER FIVE ROBUST CONTROLLER SYNTHESIS USING UNCERTAINTY VECTOR ADJUSTMENT

Figure 5.1: Application of UVA to I/O 5-10

Figure 5.2: Application of UVA to SHM 5-14

Figure 5.3: Closed Loop Dynamics for Case 1 using I/O with $\delta p = (0, 1, 0)$..... 5-19

Figure 5.4: Closed Loop Dynamics for Case 1 using I/O with $\delta_p = (0, 2, 0)$..... 5-20

Figure 5.5: Closed Loop Dynamics for Case 1 using I/O with $\delta_p = (1, 1, 0)$..... 5-21

Figure 5.6: Closed Loop Dynamics for Case 2 using I/O with $\delta_p = (0, 1, 0)$..... 5-22

Figure 5.7: Closed Loop Dynamics for Case 2 using I/O with δp = (0, 2, 0)..... 5-23

Figure 5.8: Closed Loop Dynamics for Case 2 using I/O with δp = (1, 1, 0)..... 5-24

Figure 5.9: Closed Loop Dynamics for Case 1 using SHM with δp = (0, 1, 0)..... 5-28

Figure 5.10: Closed Loop Dynamics for Case 1 using SHM with δp = (0, 2, 0)... 5-29

Figure 5.11: Closed Loop Dynamics for Case 1 using SHM with δp = (1, 1, 0)... 5-30

Figure 5.12: Closed Loop Dynamics for Case 2 using SHM with δp = (0, 1, 0)... 5-31

Figure 5.13: Closed Loop Dynamics for Case 2 using SHM with δ_p = (0, 2, 0).... 5-32

Figure 5.14: Closed Loop Dynamics for Case 2 using SHM with δp = (1, 1, 0)... 5-33

CHAPTER SIX LYAPUNOV FUNCTION ANALYSIS OF UNCERTAINTY VECTOR ADJUSTMENT

Figure 6.1: V versus dx_2 with $dx_1 = dx_3 = 0$ for SHM..... 6-23

Figure 6.2: Level Curve of $\dot{V}$ at $dx_2 = 0$ for I/O..... 6-24

Figure 6.3: Level Curve of $\dot{V}$ at $dx_2 = 0$ for SHM..... 6-24

CHAPTER SEVEN SIMULATION AND CONTROL OF A TRIPLE EFFECTS EVAPORATOR

Figure 7.1: Flowsheet of Triple Effects Evaporator Design 7-3

Figure 7.2: Closed Loop Dynamics of Triple Effects Evaporator using Linear Control (Nominal Case) 7-20

Figure 7.3: Closed Loop Dynamics of Triple Effects Evaporator using I/O Control (Nominal Case) 7-24

Figure 7.4: Closed Loop Dynamics of Triple Effects Evaporator using Linear Control (Robustness Study) 7-29

Figure 7.5: Closed Loop Dynamics of Triple Effects Evaporator using I/O Control (Robustness Study without UVA) 7-33

Figure 7.6: Closed Loop Dynamics of Triple Effects Evaporator using I/O Control (Robustness Study with UVA) 7-37

Figure 7.7: Closed Loop Responses of T_1 and h_1 of Triple Effects Evaporator..... 7-38

CHAPTER EIGHT USE OF STATE OBSERVERS IN NONLINEAR CONTROL

Figure 8.1: State Estimation of LBP using GNF 8-11
Figure 8.2: State Estimation of LBP using EKF 8-13

CHAPTER NINE PLANT IMPLEMENTATION OF INPUT OUTPUT LINEARIZATION ON THE LIQUOR BURNING PROCESS

Figure 9.1: PI-Industrial Controller Action 9-3
Figure 9.2: Conversion of PI-Industrial to PI-GLC 9-3
Figure 9.3: PI Controller Tuning for I/O Implementation 9-6
Figure 9.4: Product Flow Step Test for Feedforward Ratio Control 9-8
Figure 9.5: Product Flow Step Test for I/O Implementation 9-9
Figure 9.6: Density Set Point Test for Feedforward Ratio Control 9-11
Figure 9.7: Density Set Point Test for I/O Implementation 9-12

APPENDIX A A MAPLE PACKAGE FOR NONLINEAR CONTROL STUDY

Figure A.1: Flowchart of Procedure *io* A-3
Figure A.2: Flowchart of Procedure *ioloop* A-4
Figure A.3: Sample Plot of Procedure *ioloop* A-11
Figure A.4: Sample Plot of Procedure *shmloop* A-11

LIST OF TABLES

CHAPTER THREE NONLINEAR CONTROL OF A SIMULATED SINGLE EFFECT EVAPORATOR PROCESS

Table 3.1: List of State, Input, and Output Variables for 2×2 LBP 3-5

Table 3.2: Potential Linear Control Strategies for 2×2 LBP 3-5

Table 3.3: PI Tuning Parameters and Design Parameters $\hat{\beta}_{ik}$ for 2×2 LBP..... 3-10

Table 3.4: ITAE of 2×2 LBP using Local Linearization, I/O and SHM..... 3-10

Table 3.5: Design Parameters for GMC and I/O in 2×2 LBP 3-10

Table 3.6: ITAE of 2×2 LBP using GMC and I/O..... 3-10

Table 3.7: Parameters of Height Control Loop by I/O 3-14

Table 3.8: Parameters of Density Control Loop by I/O..... 3-14

Table 3.9: Parameters of Both Height and Density Control Loop by GMC..... 3-14

CHAPTER FOUR ROBUSTNESS STUDY OF NONLINEAR CONTROL THEORIES

Table 4.1: ITAE for Local Linearization with $Tf = 21$ and $Tf = 30$..... 4-3

Table 4.2: ITAE for GMC with $Tf = 21$ and $Tf = 30$ 4-3

Table 4.3: ITAE for I/O with $Tf = 21$ and $Tf = 30$..... 4-3

Table 4.4: List of State, Input, and Output Variables for 3×3 LPB 4-7

Table 4.5: PI Controller Tuning Parameters for SHM transformation..... 4-9

Table 4.6: ITAE for SHM Transformation with $Tf = 21$ and $Tf = 30$ 4-9

CHAPTER FIVE ROBUST CONTROLLER SYNTHESIS USING UNCERTAINTY VECTOR ADJUSTMENT

Table 5.1: Design Parameters $\hat{\beta}_{ik}$, PI Controller Tuning Parameters and Uncertainty Vector Parameters for I/O 5-17

Table 5.2: Nominal ITAE for Closed Loop Dynamics with No Modelling Error using I/O 5-17

Table 5.3: ITAE for Closed Loop Dynamics with Temperature Factor as Modelling Error using I/O (Case 1) 5-18

Table 5.4: ITAE for Closed Loop Dynamics with Temperature Factor and Feed Density as Modelling Errors using I/O (Case 2) 5-18

Table 5.5: PI Controller Tuning Parameters and Uncertainty Vector Parameters for SHM 5-26

Table 5.6: Nominal ITAE for Closed Loop Dynamics with no Modelling Error using SHM 5-26

Table 5.7: ITAE for Closed Loop Dynamics with Temperature Factor as Modelling Error using SHM (Case 1) 5-27

Table 5.8: ITAE for Closed Loop Dynamics with Temperature Factor and Feed Density as Modelling Errors using SHM (Case 2) 5-27

CHAPTER SEVEN SIMULATION AND CONTROL OF A TRIPLE EFFECTS EVAPORATOR

Table 7.1: Steady State Values of Process Variables for Evaporator Stage 1 7-4

Table 7.2: Steady State Values of Process Variables for Evaporator Stage 2 7-5

Table 7.3: Steady State Values of Process Variables for Evaporator Stage 3 7-6

Table 7.4: Steady State Values of Product Stream and Utilities 7-7

Table 7.5: List of Variables for the Triple Effects Evaporator 7-11

Table 7.6: PI Controller Tuning Parameters and Design Parameters $\hat{\beta}_{ik}$ 7-15

Table 7.7: ITAE for Triple Effects Evaporator (Nominal Case) 7-17

Table 7.8: ITAE for Robustness Study of Triple Effects Evaporator with Two Modelling Errors 7-38

Table 7.9: ITAE for Robustness Study of Triple Effects Evaporator with Five Modelling Errors 7-41

CHAPTER EIGHT USE OF STATE OBSERVERS IN NONLINEAR CONTROL

Table 8.1: ITAE for GNF and EKF 8-10

Table 8.2: ITAE for GNF and EKF with Covariance Matrix = 3 I 8-15

CHAPTER NINE PLANT IMPLEMENTATION OF INPUT OUTPUT LINEARIZATION ON THE LIQUOR BURNING PROCESS

Table 9.1: Steady State Values and Ranges of Height and Density 9-4

NOMENCLATURE

Abbreviations

EKF	Extended Kalman Filter
GLC	Global Linearizing Control
GMC	Generic Model Control
GNF	Geometric Nonlinear Filter
I/O	Input Output (Linearization)
I/S	Input State (Linearization)
ITAE	Integral of Time-Weighted Absolute Error
LBP	Liquor Burning Process
MIMO	Multiple-Input Multiple-Output
PI	Proportional and Integral
sg	Specific Gravity
SHM	Su-Hunt-Meyer (Transformation)
SISO	Single-Input Single-Output
UVA	Uncertainty Vector Adjustment

Symbols

A	Flash Tank Cross Sectional Area (m^2)
C_P	Specific Heat Capacity (J/kg/°C)
H	Enthalpy ($kJ/m^3/°C$)
h	Flash Tank Level (m)
K	Kronecker Indices
K_C	Proportional Gain of Controller (%/%)
K_I	Integral Gain of Controller (%/%)
l	Observability Index
m	Number of Manipulated Inputs
n	Number of States
Q	Flowrate (m^3/hr)

$\mathbf{Q}$, $\mathbf{R}$	Covariance Matrices of White Noise
r	relative order
$\mathbf{R}$	Set of Real Numbers
T	Temperature (°C)
t	Time (hr)
t_s	Sampling Time (hr)
$\mathbf{u}$	Manipulated Input Vector
V	Lyapunov Function
w	Number of Outputs
$\mathbf{x}$	State Vector
$\mathbf{y}$	Output Vector
$\| \ \|$	Euclidean norm on $\mathbf{R}^n$
$\dot{m}$	Mass Flowrate (kg/hr)
$\overline{\mathbf{O}}$	Observability Matrix
$\hat{\beta}_{ik}$	Scalar Parameters for Input Output Linearization
-	Steady State Value or Mean Value
ρ	Density (kg/m^3)
ν	External Input Vector of Input Output Linearization
τ_I	Integral Time of Controller (min)
δ	Uncertainty Vector
ω, υ	Gaussian White Noise
δ_p	Uncertainty Vector Parameter
ad or $[\ ,\ ]$	Lie Bracket
L or $\langle\ ,\ \rangle$	Lie Derivative

General Subscripts

a	Hot Well Temperature Approach
CW	Cooling Water
D	Flash Tank Liquid Discharge
e	Boiling Point Elevation of Liquor

F	Feed
HD	Heater Discharge
HF	Heater Feed
HW	Hot Well Water
P	Product
p	Plant
R	Recycle Flow
S	Steam
V	Flash Tank Vapour Discharge

General Superscript

sp	Set Point

CHAPTER ONE

INTRODUCTION

1.1 WHY NONLINEAR CONTROL THEORY

A chemical process modelled linearly is merely a mathematical abstraction that could never be encountered in a real world. In practice, nonlinearities exist in essentially every process control system. They arise from either the nonlinear character of the physical laws governing plant dynamics and the steady state behaviour, or man-made control elements with nonlinear characteristics.

Models formulated from nonlinear differential equations are customarily linearized around a steady state condition and are then applied using linear control theories to synthesis linear controllers (Kailath, 1980 and Chen, 1970). Using linear theories as design and analytical tools, the advantage is in the existence of analytical solutions leading to generally more rigorous stability and performance proofs. In addition, the computational requirements for the simulation and implementation of the linear system are lower than for the nonlinear case. However, the use of linear control techniques is limited where global behaviour is important. As described in the literature, linear theories neglect the nonlinear aspects of the system and are valid only in the close vicinity of steady state operations. In spite of this knowledge, most chemical control systems in industries are based on conventional linear control strategies because nonlinear control engineering is generally regarded as a difficult, and more often than not, a confusing endeavour. This remark is somehow justifiable in view of the abstract nature and the algebra involved in the theory making the results difficult to understand and to apply by the majority of control engineers. However, in the past decade, significant development has occurred in the nonlinear control field, especially the differential geometric approach. Combined with the advances in control system hardware and software, the formulation of the inherently difficult nonlinear

problems is facilitated and the practical applications of nonlinear control strategies are now real. Many successful simulations and implementations of nonlinear control schemes can be found in the literature, for example, Lee (1993), Hunt et al (1983a), Rouchon (1992), and Bequette (1991). Therefore, in recent time, the focus of research is shifting to this new and exciting area of process control.

Alcoa of Australia, as a leader in alumina refinery operations, recognises the potential of nonlinear control and its benefits over linear control. The refinery uses Bayer process to produce alumina from bauxite and current control schemes are mainly linear based. There are situations when the conventional control fails making the performance either unsatisfactory or not up to some stringent requirements. The probable consequences are off-specification products, inefficient use of resources and hence, higher production costs. Ultimately, the company may become less competitive in the world market. A process unit in one of the company's refineries was used as a test bed in this thesis for simulating and implementing the nonlinear control techniques examined.

1.2 Brief History Of Nonlinear Control Theory

Many approaches have been developed for nonlinear system theory since 1940s. Firstly, there are relatively simple techniques, such as phase-plane analysis (Graham and McRuer, 1961), which are graphical in nature and thus of limited generality. Then there are techniques based on describing function (Graham and McRuer, 1961; and Billing et al, 1984), Volterra/Wiener Series (Rugh, 1981), operator theory (de Figneiredo and Chen, 1993) and differential geometry (Isidori, 1995).

Describing function and operator theory are philosophically appealing, but are very difficult to understand and hence to implement. Volterra/Wienner approach uses some aspects of differential equation descriptions, transform representations, as well as some operator theories. The concepts involved are not difficult, but require a certain mathematical maturity in abstract realisation theory. The application of differential geometry, or Lie algebra, in nonlinear control system began in the 1970s.

The approach is considered as the equivalent mathematical tool to the matrix algebra in linear theory and it is able to provide *exact linearization*, valid over the entire operating region. The study in this thesis will focus entirely on this approach. The discussions to be presented will frequently use the term "nonlinear control theory" or "exact linearization" to refer to the class of "nonlinear control theory based on differential geometry".

1.3 MOTIVATIONS FOR THIS STUDY

The author aims to provide the practical aspects of the technique and demonstrate the superiority of the nonlinear control strategy to the linear based controller by applying it to a simulation of an evaporative stage of the liquor burning unit associated with Bayer process in the alumina refinery operation. Furthermore, the use of abstract mathematical language, as in most classic literatures, will be avoided or explained in great details. The thesis will also attempt to resolve the robustness problem for the multiple-input multiple-output (MIMO) nonlinear system associated with this rather "recent" control technique. The ultimate goal of this study is to bridge the gap between industry and academia by demonstrating the superiority and, most importantly, the simplicity of the differential geometry based nonlinear control theory. It is hoped that the insights provided during this study will encourage control engineers to, at least, consider nonlinear control strategies as an alternative to traditional linear techniques in their future projects.

The nonlinear systems to be considered here are autonomous in the standard input-linear form, the control affine system. Two static nonlinear control theories are studied in this context: Su-Hunt-Meyer (SHM) transformation (Hunt et al, 1983a) and global linearizing control (GLC) by Kravaris and Soroush (1990). In addition, a short investigation on GMC developed by Lee and Sullivan (1988) is performed. The three nonlinear control theories selected are based on their completeness as theories by themselves, and the amount of algebra and computational powers involved. Furthermore, the practical aspects and industrial values of the theories, except GMC (Lee, 1993), are rather unexplored in the chemical industry. These issues are

addressed using a fully developed simulation of a single effect evaporator and an extension to triple effects to demonstrate the intricacies of implementing these controllers in an industrial environment.

1.4 OBJECTIVES AND CONTRIBUTIONS

The objectives and contributions of the thesis are:

- To demonstrate the superiority and simplicity of the nonlinear control theory to control an industrial process (a single effect evaporator).
- To perform analysis and synthesis of a robust nonlinear controller for a MIMO system, based on the nonlinear control theories studied here, in the presence of parametric uncertainties in the modelling equations.
- To investigate the practical values and feasibility of implementing nonlinear control theory in an industrial setting.
- To further extend the application of the nonlinear control theory to a simulation of a more challenging and complicated industrial process (a triple effects evaporator).
- To bridge the gap between academia and industry in the understanding of process control knowledge and technology.
- To explore the usefulness of symbolic algebraic computations as an educational and research tool to illustrate the concept of control theory.
- To develop a software package so that systematic design methodologies for nonlinear control systems can be automated.

1.5 THESIS OVERVIEW

In Chapter 2, the introduction and literature survey of recently developed techniques of exact or feedback linearization based on differential geometry are presented. Particular attention is given to the input output linearization (I/O) and input state linearization (I/S). The mathematical preliminaries and theories of the three nonlinear control techniques, including GLC by Kravaris and Soroush (1990), SHM (Hunt et al, 1983a) and GMC by Lee and Sullivan (1988) are detailed. A simplified solution algorithm for SHM transformation is proposed in this chapter, and the symbolic

loop dynamics of the nonlinear system.

A more challenging control problem for the nonlinear control techniques is presented in Chapter 7. The target process is a triple effects evaporator which is designed on the basis of a five effects evaporation operation associated with a new liquor burning unit in Alcoa's Wagerup refinery. The process is mathematically modelled by performing mass and energy balances and is simulated as the single effect case in MAPLE. The closed loop responses are studied using the linear and nonlinear control strategies. The robust nonlinear control algorithm synthesis proposed in Chapter 6 is also tested.

Since the entire nonlinear control algorithm relies on full-state feedback, in Chapter 8 we briefly investigate situations where states are not available. In this case, either state estimation or output feedback is necessary. The example of the single effect evaporator of the liquor burning unit is used to illustrate the theories of state observation by Extended Kalman Filter (EKF) and Geometric Nonlinear Filter (GNF).

To bridge the gap between the academia and industry, the plant implementation of the I/O on the single effect evaporator of the liquor burning process in Kwinana refinery is necessary and is also significant for demonstrating the practicability of the nonlinear control strategy. The discussions and results are presented in Chapter 9. Finally, conclusions and recommendations are given in Chapter 10.

CHAPTER TWO

FEEDBACK LINEARIZATION AND SYMBOLIC COMPUTATIONS

2.1 INTRODUCTION TO FEEDBACK LINEARIZATION THEORY

In the past decade, there has been significant progress in nonlinear system theory using differential geometry techniques. This was motivated by the geometric approach of linear theory introduced by Woham (1979). Using differential geometry as the mathematical tool, one can apply "word by word" conversion of most of the result in linear systems to nonlinear systems. The major success is the development of the *state feedback linearization theory* in the design of nonlinear control system. The central idea of state feedback linearization is to transform a nonlinear system into either a fully, or a partially linear system via state feedback. One can then use the matured linear controller design technique to complete the control circuit. Being different from the traditional linearization using the first order Taylor's series, the linearization of a nonlinear system using this method is exact and valid throughout the entire operating region of interest.

2.1.1 Full State Feedback Linearization

Full state feedback linearization means that the linearization is performed in the input-state sense. Therefore, it is also called the input state linearization (I/S). The operation involves three fundamental steps: nonlinear change of coordinate in the state space, nonlinear change of coordinate in the input space and nonlinear state feedback.

Figure 2.1 shows the relationship between the nonlinear and transformed linear system. The control of the original nonlinear system is conducted through the transformed linear system because of the existence of exact linear relationship

between the transformed linear state and the transformed linear input. The transformations are algebraic functions mapping the nonlinear to linear space.

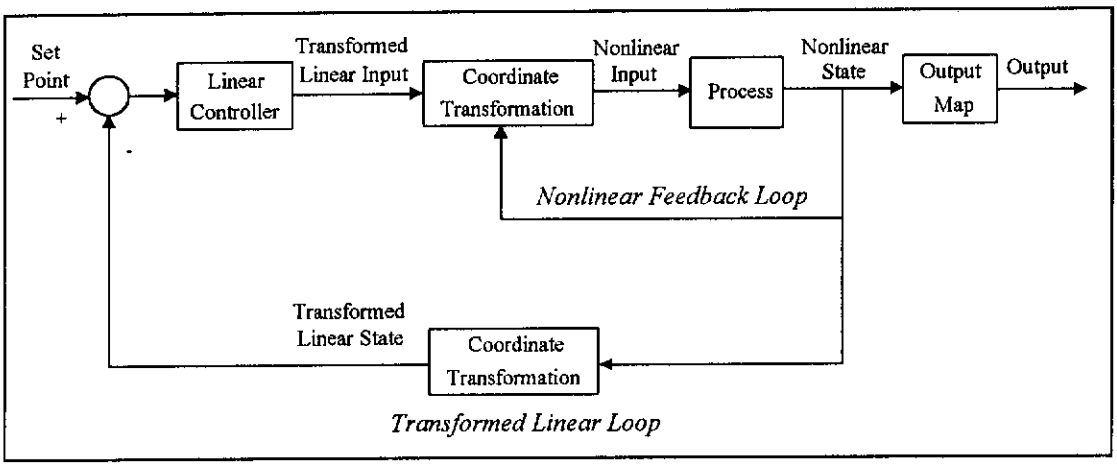


Figure 2.1: The Fundamental Structure of Full State Linearization

2.1.2 Partial State Feedback Linearization

Partial state feedback linearization linearizes the nonlinear system in the input-output sense. Exact linear relationship exists between the transformed linear inputs and the original outputs of the system. The steps involved in this operation are nonlinear changes of coordinates in the input space and nonlinear state feedback. *Figure 2.2* shows that the control of the nonlinear system is performed through the transformed linear input-output loop.

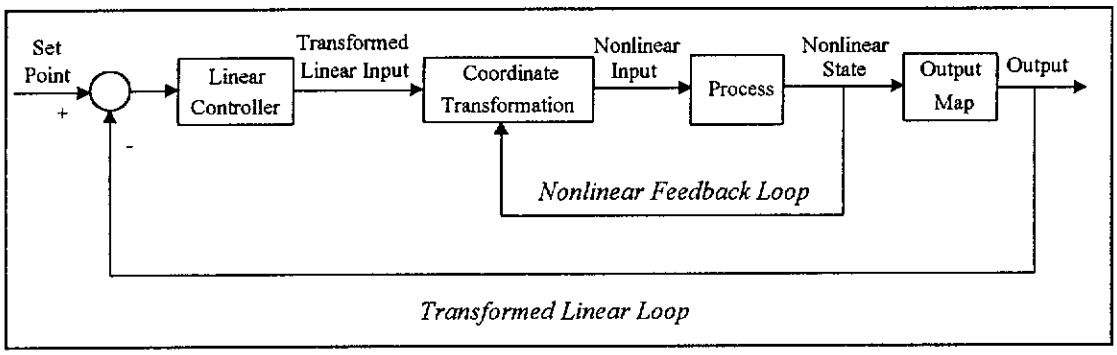


Figure 2.2: The Fundamental Structure of Partial State Feedback Linearization

2.2 LITERATURE RESEARCH

Two books are available on the state feedback linearization by differential geometry, including Isidori (1995) and Slotine and Li (1991). The book by Slotine and Li (1991) presents the fundamental results of nonlinear control theory while keeping the mathematical complexity to a minimum degree. It also demonstrates their uses and implications in the design of practical nonlinear systems. The other book by Isidori (1995) provides a detailed and recent perspective of the field. However, the book employs modern algebraic language.

Apart from these books, three review papers must also be mentioned. Kravaris and Kantor (1990) provides an excellent review on geometric methods for nonlinear process control. The discussion is limited to single-input single-output (SISO) systems. However, the references are current and extensive. The other two review papers are by Bequette (1991) and Kantor (1987). Bequette (1991) surveyed nonlinear control system techniques and its applications in industries from ad-hoc or process-specific strategies to predictive control approaches based on nonlinear programming. Kantor (1987), on the other hand, focuses on the theory and the applications of differential geometry in the nonlinear control system.

In this dissertation, static state feedback linearization is to be investigated. For dynamic state feedback linearization and systems which are not feedback linearizable, discussions and theories can be found in Kang (1991). The investigations in this content are divided into two main areas: static input-state linearization and static input-output linearization.

2.2.1 Input State Linearization (I/S)

I/S was firstly introduced by Brockett (1978) on the SISO nonlinear system. Su (1982) extended the theory to a wider class of linear equivalent nonlinear systems and gave a comparison of the two theories. Hunt et al (1983a) extended the transformation proposed by Su (1982) to MIMO systems and called it the Su-Hunt-

Meyer (SHM) transformation which becomes one of the representatives of I/S. A similar but more general theory of I/S was developed by Isidori (1995). This theory further advanced the idea of SHM transformation to cover a broader class of nonlinear systems. A discussion of the relationship between the two theories can be found in a later section. Another technique called pseudolinearization, developed by Reboulet and Champetier (1984), also extended the SHM transformation with less restrictive conditions of linearizability. Once again a broader class of nonlinear systems can then be linearized. However, the technique presented is only valid for SISO cases. An overview of SHM transformation can be found in Alsop (1987) as well. Alsop extended the singular value decomposition in the linear control theory to decompose a MIMO system into various SISO subsystems so that nonlinear control theories developed for the SISO system can be applied to MIMO systems. His investigations included SHM transformation, Krener's approximate transformation technique (Krener, 1984), and direct linearization approach (Hermann, 1984). Henson (1992) and Slotine and Li (1991) also provided brief discussions on I/S.

The actual application in chemical process industry of SHM transformation or other I/S is quite limited. Therefore, this motivates the current work to investigate the practical applicability of this theory in this area.

2.2.2 Input Output Linearization (I/O)

The literature on static I/O is more extensive than those on I/S. The first results appeared in 1980s and early works can be found in Isidori and Krener (1982), Nijmeiger (1982), Isidori and Ruberti (1984), Boothby (1984), Rugh (1984) and Gilbert and Ha (1984). The theories developed during that period assumed very restrictive assumptions on process dynamics and were very mathematically oriented. Therefore, their applications were very limited. Kravaris and Chung (1987) advanced the work of Gilbert and Ha (1984) and successfully developed a relatively simple, less restrictive theory on I/O for SISO nonlinear systems. The control structure is called global linearizing control (GLC). Daoutidis and Kravaris (1989) synthesised a feedforward/state feedback controllers for a broader class of SISO systems with

measurable disturbances based on the I/O presented in Kravaris and Chung (1987). In 1990, Kravaris and Soroush followed up on the ideas and extended the GLC with state feedback and feedforward/state feedback to MIMO systems.

The synthesis of the discrete-time nonlinear feedforward/state feedback and state feedback controllers were addressed in Soroush and Kravaris (1994) and (1992) respectively. Henson and Seborg (1990) extended the work of Daoutidis and Kravaris (1989) to a more general class of process models, in which the manipulated input and disturbances may appear nonlinearly, instead of the typical input-linear form. The aforementioned literature is concerned with minimum phase processes having stable zero dynamics. For non-minimum phase processes, the nonlinear controller synthesis based on I/O can be found in Kravaris et al (1994). In this article, Kravaris replaced the state feedback requirement in the GLC with output feedback, the idea of which was first introduced in Wright and Kravaris (1993). Dynamic output feedback controller for minimum phase processes was successfully synthesised in Daoutidis and Kravaris (1994). The notion and details of nonlinear zero dynamics of a multivariable process can be found in Calvet (1989) and Isidori (1995). The experimental studies of GLC in chemical processes include, for example, Nakamoto and Watanabe (1991), Soroush and Kravaris (1993, 1994), and Palanki et al (1994). Finally, a comprehensive overview of the status and development of GLC is provided in Kravaris and Arkun (1989) which listed several chemical processes using this technique.

Another famous control structure, the generic model control (GMC), was developed by Lee and Sullivan (1988). It is based on the theory of state feedback linearization in the input-output map. The technique is very simple and received a lot of attention in the control industry. A complete study of GMC can be found in Zhou (1990). It contains materials on robustness and extensions to systems with higher relative orders and with constraints. Lee (1993) presented some industrial applications of GMC such as distillation and evaporation processes.

The focus of this thesis in the area of I/O is on the practical aspects of this control structure. The I/O studied here is based on Kravaris and Soroush (1990), that is, the nonlinear multivariable process is minimum phase and the state feedback linearization is static which requires full state feedback. Investigations on GMC will also be carried out and the relationship between the two structures will be discussed in section 2.4.2.2.

2.3 MATHEMATICAL PRELIMINARIES FOR DIFFERENTIAL GEOMETRY

The objective of this section is to provide the necessary background for differential geometry which will be applied in subsequent chapters. Further details can be found in Kravaris and Kantor (1990), Isidori (1995) or Vidyasagar (1993).

2.3.1 Vector Fields and Scalar Fields

Let $\mathbf{R}^n$ denote the n-dimensional Euclidian space with elements being 'points' or 'vectors' depending on the context.

A *scalar field* on $\mathbf{R}^n$ is a function defined on an open subset $U \subset \mathbf{R}^n$ with values in $\mathbf{R}$. In other words, a scalar field $h(\mathbf{x})$ assigns to a given element $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ of U a real number $h(x_1, x_2, \cdots, x_n)$. A C^1 scalar field is a continuous function of $(x_1, x_2, \cdots, x_n)$ for which all partial derivatives exist while a C^∞ scalar field is a continuous function for which all partial derivatives of arbitrary order exist.

A *vector field* on $\mathbf{R}^n$ is a vector function defined on an open subset $U \subset \mathbf{R}^n$ with values in $\mathbf{R}^n$. Hence, a vector field $\mathbf{g}(\mathbf{x})$ assigns to a given element $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ of U the vector

$$\mathbf{g}(x_1, x_2, \ldots, x_n) = \begin{bmatrix} g_1(x_1, x_2, \ldots, x_n) \\ g_2(x_1, x_2, \ldots, x_n) \\ \vdots \\ g_n(x_1, x_2, \ldots, x_n) \end{bmatrix} \text{ in } \mathbf{R}^n.$$

A C^1 vector field on $\mathbf{R}^n$ is a continuous function of $(x_1, x_2, \cdots, x_n)$ for which all partial derivatives exist, while a C^∞ vector field on $\mathbf{R}^n$ is a continuous function for which all partial derivatives of arbitrary order exist.

2.3.2 Lie Derivative

Given a C^∞ vector field $\mathbf{f}$ on $\mathbf{R}^n$, and a C^∞ scalar field h on $\mathbf{R}^n$, the Lie derivative of h with respect to $\mathbf{f}$ is defined as

$$L_{\mathbf{f}} h = \langle dh, \mathbf{f} \rangle = \frac{\partial h}{\partial x_1} f_1 + \frac{\partial h}{\partial x_2} f_2 + \cdots + \frac{\partial h}{\partial x_n} f_n \quad \text{..... (2.1)}$$

The Lie derivative is a C^∞ scalar field on $\mathbf{R}^n$. Thus, higher order Lie derivatives can be defined inductively as follows:

$$L_{\mathbf{f}}^k h = L_{\mathbf{f}}\left(L_{\mathbf{f}}^{k-1} h\right) = \left\langle dL_{\mathbf{f}}^{k-1} h, \mathbf{f} \right\rangle \qquad k = 2, 3, \ldots \quad \text{..... (2.2)}$$

Algebraic Properties of the Lie derivative

1. $L_{\mathbf{f}}\left(c_1 h_1 + c_2 h_2\right) = c_1 L_{\mathbf{f}} h_1 + c_2 L_{\mathbf{f}} h_2$ ..... (2.3)

   where c_1 *and* c_2 are scalar constants.

2. $L_{a_1(\mathbf{x})\mathbf{f}_1 + a_2(\mathbf{x})\mathbf{f}_2} h = a_1(\mathbf{x}) L_{\mathbf{f}_1} h + a_2(\mathbf{x}) L_{\mathbf{f}_2} h$ ..... (2.4)

   where $a_1(\mathbf{x})$ and $a_2(\mathbf{x})$ are scalar functions of $\mathbf{x}$.

2.3.3 Lie Bracket

Given C^∞ vector fields $\mathbf{f}$, $\mathbf{g}$ on $\mathbf{R}^n$, the Lie bracket [f, g] is a C^∞ vector field on $\mathbf{R}^n$ defined by

$$[\mathbf{f}, \mathbf{g}] = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \mathbf{f} - \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \mathbf{g} \quad \text{..... (2.5)}$$

where $\partial \mathbf{f} / \partial \mathbf{x}$ and $\partial \mathbf{g} / \partial \mathbf{x}$ are the Jacobians.

Successive iterated Lie brackets can be defined with the standard notation as follows:

$$\begin{aligned} ad^0\mathbf{f},\mathbf{g} &= \mathbf{g} \\ ad^1\mathbf{f},\mathbf{g} &= [\mathbf{f},\mathbf{g}] \\ &\vdots \\ ad^k\mathbf{f},\mathbf{g} &= \left[\mathbf{f},\left(ad^{k-1}\mathbf{f},\mathbf{g}\right)\right] \end{aligned} \quad \text{..... (2.6)}$$

Algebraic Properties of the Lie bracket

1. Antisymmetry

$$[\mathbf{f},\mathbf{g}] = -[\mathbf{g},\mathbf{f}] \quad \text{..... (2.7)}$$

2. For the scalar constants c_1 *and* c_2,

$$[c_1\mathbf{f}_1 + c_2\mathbf{f}_2,\mathbf{g}] = c_1[\mathbf{f}_1,\mathbf{g}] + c_2[\mathbf{f}_2,\mathbf{g}] \quad \text{..... (2.8)}$$

3. Jacobi's identity

$$[[\mathbf{f}_1,\mathbf{f}_2],\mathbf{f}_3] + [[\mathbf{f}_2,\mathbf{f}_3],\mathbf{f}_1] + [[\mathbf{f}_3,\mathbf{f}_1],\mathbf{f}_2] = 0 \quad \text{..... (2.9)}$$

4. Leibnitz's formula

$$\begin{aligned} L_{\mathbf{g}} L_{\mathbf{f}} h &= L_{\mathbf{f}} L_{\mathbf{g}} h - L_{[\mathbf{f},\mathbf{g}]} h \\ L_{\mathbf{g}} L_{\mathbf{f}}^2 h &= L_{\mathbf{f}}^2 L_{\mathbf{g}} h - 2L_{\mathbf{f}} L_{[\mathbf{f},\mathbf{g}]} h + L_{[\mathbf{f},[\mathbf{f},\mathbf{g}]]} h \end{aligned}$$

In general,

$$\begin{aligned} L_{\mathbf{g}} L_{\mathbf{f}}^k h = {} & L_{\mathbf{f}}^k L_{\mathbf{g}} h - \binom{k}{1} L_{\mathbf{f}}^{k-1} L_{ad_{\mathbf{f}}^1 \mathbf{g}} h + \binom{k}{2} L_{\mathbf{f}}^{k-2} L_{ad_{\mathbf{f}}^2 \mathbf{g}} h - \\ & \binom{k}{3} L_{\mathbf{f}}^{k-3} L_{ad_{\mathbf{f}}^3 \mathbf{g}} h + \cdots + (-1)^k L_{ad_{\mathbf{f}}^k \mathbf{g}} h \end{aligned} \quad \text{..... (2.10)}$$

2.4 DEFINING THE NONLINEAR CONTROL SYSTEM

Throughout this thesis, we consider the following multivariable autonomous (time-invariant) input-linear system. It is given in state space form, with m inputs and w

outputs and without direct feed-through from input to output.

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t)) + \sum_{i=1}^{m} \mathbf{u}_i(t)\mathbf{g}_i(\mathbf{x}(t)) \\ y_i(t) &= h_i(\mathbf{x}(t)) \qquad i = 1,\ldots,w \end{aligned} \quad \ldots\ldots (2.11)$$

where $\mathbf{f}$, $\mathbf{g}_1$, $\mathbf{g}_2$, ..., $\mathbf{g}_m$ are C^∞ vector fields on $\mathbf{R}^n$, and $h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_w(\mathbf{x})$ are scalar fields on $\mathbf{R}^n$. In other words, the mappings $\mathbf{f}$, $\mathbf{g}_1$, $\mathbf{g}_2$, ..., $\mathbf{g}_m$, and the functions $h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_w(\mathbf{x})$ are smooth in their arguments, that is, all entries are real-valued functions of $\mathbf{x}$ with continuous partial derivatives of any order. In addition, $\mathbf{y}$, $\mathbf{x}$ and $\mathbf{u}$ represent the deviation vectors with $\mathbf{y} \in \mathbf{R}^w$, $\mathbf{x} \in \mathbf{R}^n$, and $\mathbf{u} \in \mathbf{R}^m$, respectively.

As a state space for a model, a subspace U of $\mathbf{R}^n$ is sometimes considered rather than $\mathbf{R}^n$ itself. This may be due to physical constraints in the process, a mathematical constraint established by the equations themselves when some solutions or trajectories are not allowed, or constraints imposed by the input or initial state in order to avoid singularities at some points in the state space. However, in many cases, it is allowed to simply set $U = \mathbf{R}^n$ (Vidyasagar, 1993).

2.4.1 Su-Hunt-Meyer (SHM) Transformation

The SHM transformation (Hunt et al, 1983a) maps the nonlinear system described in (2.11) to the linear system in Brunovsky's canonical form (Brunovsky, 1970; and Luenberger, 1967) given in (2.12) below. Kravaris and Kantor (1990) showed that the exact feedback linearization of Isidori (1995) is equivalent to the SHM approach when $n = r$, where r is the relative order for the SISO system. They indicated that this equivalence is preserved for MIMO systems. However, the dimension n of the state space must exactly equal the sum of the relative orders $r_1, r_2, \ldots, r_m$ (Isidori, 1995). The relative order r_i is defined in (2.30) in the next section. The Brunovsky's canonical form is:

$$\dot{\xi}(t) = \mathbf{A}\,\xi(t) + \mathbf{B}\,\mu(t) \quad \text{..... (2.12)}$$

where $\xi \in \mathbf{R}^n$, $\mu \in \mathbf{R}^m$ and

$$\mathbf{A}_{n \times n} = \begin{bmatrix} \begin{matrix}0100\ldots0\\0010\ldots0\\ \vdots\ \ddots\ \vdots\\0000\ldots1\\0000\ldots0\end{matrix} & \begin{matrix}0\\\\\\\\\end{matrix} & \cdots\cdots & \begin{matrix}0\\\\\\\\\end{matrix} \\ \begin{matrix}0\\\\\\\\\end{matrix} & \begin{matrix}0100\ldots0\\0010\ldots0\\ \vdots\ \ddots\ \vdots\\0000\ldots1\\0000\ldots0\end{matrix} & \cdots\cdots & \begin{matrix}0\\\\\\\\\end{matrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{matrix}0\\\\\\\\\end{matrix} & \begin{matrix}0\\\\\\\\\end{matrix} & \cdots\cdots & \begin{matrix}0100\ldots0\\0010\ldots0\\ \vdots\ \ddots\ \vdots\\0000\ldots1\\0000\ldots0\end{matrix} \end{bmatrix} \begin{matrix}\}K_1\\\}K_2\\\\\}K_m\end{matrix}$$

$$\mathbf{B}_{n \times m} = \begin{bmatrix} \begin{matrix}0000\ldots0\\\vdots\\1000\ldots0\end{matrix} \\ \begin{matrix}0000\ldots0\\\vdots\\0100\ldots0\end{matrix} \\ \vdots \\ \begin{matrix}0000\ldots0\\\vdots\\0000\ldots1\end{matrix} \end{bmatrix} \begin{matrix}\}K_1\\\}K_2\\\\\}K_m\end{matrix}$$

The *Kronecker indices* of the system are K_1, K_2, ..., K_m indicating the orders of the various subsystems of integrators in series which make up the Brunovsky's canonical form. The indices are invariant to feedback transformation and have the following characteristics:

1. $\sum_{i=1}^{m} K_i = n$
2. $K_1 \geq K_2 \geq \ldots \geq K_m$

The variable transformation generates new states in terms of the original nonlinear states and new inputs as a function of the original inputs and the original nonlinear states. The following sections give details on how to obtain the transformation.

2.4.1.1 Algorithm for Computing Kronecker Indices

The criterion to yield the correct Kronecker indices for the nonlinear system (2.11) is that the system is linearizable. The algorithm for determining the Kronecker indices is presented below with discussions in Alsop (1987), Calvet (1989) and Kalman (1972).

1. Generate the following matrix with $(n - m) + 1$ rows

$$\left.\begin{bmatrix} \mathbf{g}_1 & \mathbf{g}_2 & \cdots & \mathbf{g}_m \\ [\mathbf{f},\mathbf{g}_1] & [\mathbf{f},\mathbf{g}_2] & \cdots & [\mathbf{f},\mathbf{g}_m] \\ \vdots & \vdots & \vdots & \vdots \\ (ad^{n-m}\mathbf{f},\mathbf{g}_1) & (ad^{n-m}\mathbf{f},\mathbf{g}_2) & \cdots & (ad^{n-m}\mathbf{f},\mathbf{g}_m) \end{bmatrix}\right\} (n-m+1) \text{ rows} \quad \ldots\ldots\ldots (2.13)$$

2. Set

α_0 = number of independent vector fields in the first row

α_1 = number of independent vector fields in the first two rows

$\vdots$

α_{n-m-1} = number of independent vector fields in the first $(n-m)$ rows

α_{n-m} = number of independent vector fields in the entire $(n-m+1)$ rows

3. Define

r_0 = α_0 (always equal to the number of inputs, m)

r_1 = $\alpha_1 - \alpha_0$

$\vdots$

r_{n-m} = $\alpha_{n-m} - \alpha_{n-m-1}$

4. A value for K_i is determined by the number of r_j that satisfy $r_j \geq i$ ($i = 1, \ldots, m$ and $j = 0, \ldots, n-m$).

5. To minimise the generation of unnecessary rows in (2.13), it was recommended by Alsop (1987) to start with the first $n / (m+1)$ rows of the matrix in (2.13) and determine the value of K_i ($i = 1, \ldots, m$) by steps 2 to 4. If $\sum_{i=1}^{m} K_i < n$, then add one row in (2.13) and repeat steps 2 to 4. The addition of rows is repeated until $\sum_{i=1}^{m} K_i = n$.

2.4.1.2 Criteria for Transformation

After the values of K_i have been determined, three criteria (Hunt et al, 1983a) need to be satisfied before proceeding to the variable transformation.

1. The rank of the following matrix **C** must be n in some neighbourhood of the origin for the nonlinear process to be controllable and linearizable.

$$\mathbf{C} = \left\{\mathbf{g}_1, [\mathbf{f},\mathbf{g}_1], \cdots, ad^{K_1-1}\mathbf{f},\mathbf{g}_1, \mathbf{g}_2, [\mathbf{f},\mathbf{g}_2], \cdots, ad^{K_2-1}\mathbf{f},\mathbf{g}_2, \ldots, \mathbf{g}_m, [\mathbf{f},\mathbf{g}_m], \cdots, ad^{K_m-1}\mathbf{f},\mathbf{g}_m\right\} \quad \text{..... (2.14)}$$

2. The set of vector fields,

$$\mathbf{C}_j = \left\{\mathbf{g}_1, [\mathbf{f},\mathbf{g}_1], \cdots, ad^{K_j-2}\mathbf{f},\mathbf{g}_1, \mathbf{g}_2, [\mathbf{f},\mathbf{g}_2], \cdots, ad^{K_j-2}\mathbf{f},\mathbf{g}_2, \ldots, \mathbf{g}_m, [\mathbf{f},\mathbf{g}_m], \cdots, ad^{K_j-2}\mathbf{f},\mathbf{g}_m\right\} \quad \text{..... (2.15)}$$

must be involutive for $j = 1, 2, \ldots, m$.

For a set of vector fields $\mathbf{Z} = \left\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_p\right\}$, *involutivity* means that

$$\left[\mathbf{z}_i(\mathbf{x}), \mathbf{z}_j(\mathbf{x})\right] = \sum_{k=1}^{p} \alpha_k^{ij}(\mathbf{x})\mathbf{z}_k(\mathbf{x}),$$

where $\alpha_k^{ij}(\mathbf{x})$ is a function of **x** and $1 \le i, j \le p$, $i \ne j$. To illustrate the basic notion of involutivity, we consider the following pair of three dimensional first-order homogeneous linear partial differential equations with $h(x)$ as the solution.

$$L_p h = p_1(x)\frac{\partial h}{\partial x_1} + p_2(x)\frac{\partial h}{\partial x_2} + p_3(x)\frac{\partial h}{\partial x_3} = 0$$

$$L_q h = q_1(x)\frac{\partial h}{\partial x_1} + q_2(x)\frac{\partial h}{\partial x_2} + q_3(x)\frac{\partial h}{\partial x_3} = 0$$

where **p** and **q** are vector functions. If a nontrivial solution exists, then

$$L_{[\mathbf{p},\mathbf{q}]}\,h = L_{\mathbf{p}}\left(L_{\mathbf{q}}\,h\right) - L_{\mathbf{q}}\left(L_{\mathbf{p}}\,h\right) = L_{\mathbf{p}}(0) - L_{\mathbf{q}}(0) = 0$$

In particular, if $[p\ ,\ q]$ can be written as a weighted sum

$$[\mathbf{p},\mathbf{q}] \;=\; a(x)\mathbf{p} + b(x)\mathbf{q}$$

then $L_{[\mathbf{p},\mathbf{q}]}\,h = 0$ from (2.4). The constraints turn out to be necessary and sufficient conditions for the existence of solutions to the system of partial differential equations and is called involutivity.

3. $\operatorname{span}\left(\mathbf{C} \cap \mathbf{C}_j\right) = \operatorname{span}\left(\mathbf{C}_j\right) \qquad j = 1,\ldots,m$ ..... (2.16)

2.4.1.3 Transformation of Variables

We define the following transformation variables,

$$\begin{aligned}(x_1, x_2, \cdots, x_n) &= \left(T_1, T_2, \cdots, T_n\right)\\ (u_1, u_2, \cdots, u_m) &= \left(T_{n+1}, T_{n+2}, \cdots, T_{n+m}\right)\end{aligned} \quad \text{..... (2.17)}$$

Transformations are obtained as solutions of the following systems of partial differential equations.

1. Calculate $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$ with

$$\begin{aligned}\left\langle d\,T_1, \left(ad^j\mathbf{f}, \mathbf{g}_i\right)\right\rangle &= 0 \qquad j = 0,1,\ldots,\mathrm{K}_1 - 2\\ \left\langle d\,T_{\sigma_1+1}, \left(ad^j\mathbf{f}, \mathbf{g}_i\right)\right\rangle &= 0 \qquad j = 0,1,\ldots,\mathrm{K}_2 - 2\\ &\vdots\\ \left\langle d\,T_{\sigma_{m-1}+1}, \left(ad^j\mathbf{f}, \mathbf{g}_i\right)\right\rangle &= 0 \qquad j = 0,1,\ldots,\mathrm{K}_m - 2\end{aligned} \quad \text{..... (2.18)}$$

where $\sigma_j = \sum_{i=1}^{j} \mathrm{K}_i$ and $i = 1, 2,\ldots, m$.

2. With $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$ from (2.18), T_j's ($j = 2, \ldots, n, j \neq \sigma_1 + 1 \neq \sigma_2 + 1 \neq \ldots \neq \sigma_{m-1} + 1$) are obtained by solving the following set of partial differential equations,

$$T_j = \langle d T_{j-1}, \mathbf{f} \rangle \quad \ldots\ldots (2.19)$$

Hence (2.18) and (2.19) provide $T_1, \ldots, T_n$.

3. $T_{n+1}, \ldots, T_{n+m}$ can be determined by

$$\begin{aligned} \left\langle d T_{\sigma_1}, \mathbf{f} + \sum_{i=1}^{m} \mathbf{g}_i u_i \right\rangle &= T_{n+1} \\ \left\langle d T_{\sigma_2}, \mathbf{f} + \sum_{i=1}^{m} \mathbf{g}_i u_i \right\rangle &= T_{n+2} \\ &\vdots \\ \left\langle d T_{n}, \mathbf{f} + \sum_{i=1}^{m} \mathbf{g}_i u_i \right\rangle &= T_{n+m} \end{aligned} \quad \ldots\ldots (2.20)$$

The above equations show that only the solutions $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$ of (2.18) need to be found. Since the first-order linear partial differential equations can be solved by reducing to systems of ordinary differential equations, the required ordinary differential equations are given below and then the solutions are chosen. It is clear that these m functions will not be unique due to the arbitrary selection of the associated functions in (2.18).

With $K_1 \geq K_2 \geq \ldots \geq K_m$, we define

$$S_\ell = \operatorname{rank}\left\{ \left(ad^{K_1 - \ell} \mathbf{f}, \mathbf{g}_i \right) \mid \left(ad^{K_1 - \ell} \mathbf{f}, \mathbf{g}_i \right) \in \mathbf{C} \right\}_{i=1,\ldots,m} \qquad \ell = 1,2,\ldots \ldots (2.21)$$

Real parameters $t_1, t_2, \ldots, t_n$ are introduced and with S_1,

$$\begin{aligned}
\frac{\partial \mathbf{x}}{\partial t_1} &= \left(ad^{K_1-1}\mathbf{f},\mathbf{g}_1\right) \quad \text{s.t.} \quad \mathbf{x}(0)=0 \\
\frac{\partial \mathbf{x}}{\partial t_2} &= \left(ad^{K_1-1}\mathbf{f},\mathbf{g}_2\right) \quad \text{s.t.} \quad \mathbf{x}(t_1,0)=\mathbf{x}(t_1) \\
&\vdots \\
\frac{\partial \mathbf{x}}{\partial t_{S_1}} &= \left(ad^{K_1-1}\mathbf{f},\mathbf{g}_{S_1}\right) \quad \text{s.t.} \quad \mathbf{x}(t_1,t_2,\ldots,t_{S_1-1},0)=\mathbf{x}(t_1,t_2,\ldots,t_{S_1-1})
\end{aligned} \qquad \ldots\ldots (2.22)$$

then with S_2, the second set of equations is

$$\begin{aligned}
\frac{\partial \mathbf{x}}{\partial t_{S_1+1}} &= \left(ad^{K_1-2}\mathbf{f},\mathbf{g}_1\right) \quad \text{s.t. } \mathbf{x}(t_1,\ldots,t_{S_1},0)=\mathbf{x}(t_1,\ldots,t_{S_1}) \\
\frac{\partial \mathbf{x}}{\partial t_{S_1+2}} &= \left(ad^{K_1-2}\mathbf{f},\mathbf{g}_2\right) \quad \text{s.t. } \mathbf{x}(t_1,\ldots,t_{S_1},t_{S_1+1},0)=\mathbf{x}(t_1,\ldots,t_{S_1},t_{S_1+1}) \\
&\vdots \\
\frac{\partial \mathbf{x}}{\partial t_{S_1+S_2}} &= \left(ad^{K_1-2}\mathbf{f},\mathbf{g}_{S_2}\right) \quad \text{s.t. } \mathbf{x}(t_1,\ldots,t_{S_2},0)=\mathbf{x}(t_1,\ldots,t_{S_2})
\end{aligned} \qquad \ldots\ldots (2.23)$$

The above process is continued until the last set of equations is generated with $S_r = m$ and all the parameters $t_1, t_2, \ldots, t_n$ are introduced. The process will end with

$$\frac{\partial \mathbf{x}}{\partial t_n} = \mathbf{g}_m \qquad \text{s.t.} \quad \mathbf{x}(t_1,t_2,\ldots,t_{n-1},0)=\mathbf{x}(t_1,t_2,\ldots,t_{n-1}) \qquad \ldots\ldots (2.24)$$

Hence, from the sequential solutions of the above equations, the $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$ transforms are set to be equal to the solution for the t_i variable determined from the set of differential equations of the form:

$$\frac{\partial \mathbf{x}}{\partial t_i} = \left(ad^{K_\ell-1}\mathbf{f},\mathbf{g}_\ell\right) \qquad \text{for } \ell = 1,\ldots,m \qquad \ldots\ldots (2.25)$$

From (2.18), $T_j = t_i(x)$ for $j = 1, \sigma_1 + 1, \ldots, \sigma_{m-1} + 1$.

2.4.1.4 Proposed Special Case of the SHM Transformation

The equations in SHM transformation can become very involved and impossible to handle in some situations. Therefore, in this thesis, a simplified solution algorithm is developed and formulated. It is observed that a special case of the SHM transformation arises when the number of state variables is equal to that of manipulated inputs (i.e. $n = m$). With $n = m$ and the proposed solution algorithm presented below, the calculations involved are simplified and can easily be handled by existing computational software.

1. Kronecker indices

   $K_1 = K_2 = \ldots = K_m = 1$

2. Criteria for Transformation

   With $K_1 = K_2 = \ldots = K_m = 1$ and if $\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_m$ are linearly independent, criteria 1 and 3 described above are trivially satisfied. The criterion on involutivity still poses a problem but is now easier to handle than before.

3. Transformation of Variables

   As mentioned earlier, only the solutions $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$ need to be found. By setting the initial conditions of the ordinary differential equations (2.22) to (2.25) to zero, these solutions are *arbitrarily* selected to be

$$\begin{aligned} T_1 &= \int_0^{x_1} \frac{1}{\mathbf{g}_1(1)} dx_1 \qquad \text{s.t. } \mathbf{x}(0) = 0 \\ T_2 &= \int_0^{x_2} \frac{1}{\mathbf{g}_2(2)} dx_2 \qquad \text{s.t. } \mathbf{x}(0) = 0 \\ &\vdots \\ T_n &= \int_0^{x_n} \frac{1}{\mathbf{g}_m(n)} dx_n \qquad \text{s.t. } \mathbf{x}(0) = 0 \end{aligned} \qquad \ldots\ldots (2.26)$$

where $\mathbf{g}_m(n)$ refers to the n^{th} component of the vector $\mathbf{g}_m$. To ensure the existence of the integrals in (2.26), $\mathbf{g}_m(n)$ is not equal to zero. The initial conditions are selected using the fact that all solution trajectories pass through the origin.

The solutions $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$ determined by the above initial condition $\mathbf{x}(0) = 0$ will not be as general as those provided by Hunt et al (1983). Nevertheless, these solutions are solutions which can be easily obtained analytically and are simpler and easier to handle in most circumstances. The effectiveness of the modified algorithm depends on the actual process and its degree of nonlinearity. Since the special case is a simplified version, or a subset, of the original SHM approach, the degree of nonlinearity that can be handled will be reduced. It should be noted that (2.26) can also be applied to situations where $n \geq m$.

2.4.1.5 Summary of the Algorithm for the SHM Transformation

The steps of the SHM transformation are listed below.

1. The Kronecker indices are computed using algorithm described in section 2.4.1.1.
2. Through (2.22) to (2.25) (or using the simplified algorithm in (2.26)), determine $T_1, T_{\sigma_1+1}, \ldots, T_{\sigma_{m-1}+1}$.
3. The rest of the **T** transforms are calculated using (2.19) and (2.20).

2.4.1.6 Feedback Control by the SHM Transformation

Using the **T** transformation computed above, the feedback control scheme can be designed for the nonlinear process. The following block diagram which is a simplified version of the figure provided in Hunt et al (1983a) explains how the transformation can be implemented and how a linear controller is incorporated in the control scheme. The linear control strategy used is usually the proportional and integral linear controller because it is most commonly used in the plant.

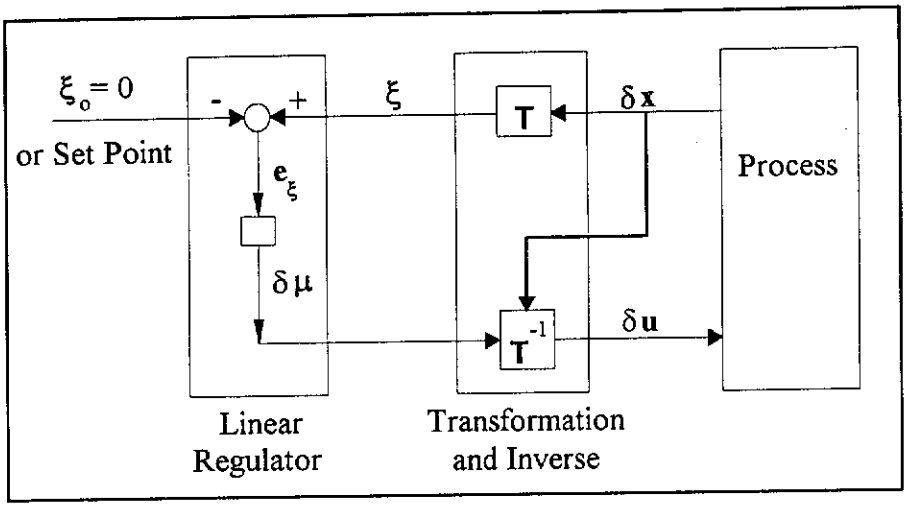


Figure 2.3: Feedback Control by SHM Transformation

2.4.2 Input Output Linearization (I/O)

Consider a minimum phase nonlinear system as shown in (2.11) with equal number of inputs and outputs ($m = w$) , input output linearization (Kravaris and Soroush, 1990) determines a static state feedback law of the form:

$$\mathbf{u}(\mathbf{x}) = \mathbf{P}(\mathbf{x}) + \mathbf{Q}(\mathbf{x})\,\nu \qquad (2.27)$$

where ν is the external input vector of the transformed linear closed loop system so that the closed loop behaviour between the external input and the output is linear. The required $\mathbf{P}(\mathbf{x})$ and $\mathbf{Q}(\mathbf{x})$ for a decoupled closed loop input output behaviour in (2.27) are defined as

$$\begin{aligned} \mathbf{P}(\mathbf{x}) &= -\mathbf{A}(\mathbf{x})^{-1}\mathbf{B}(\mathbf{x}) \\ \mathbf{Q}(\mathbf{x}) &= \mathbf{A}(\mathbf{x})^{-1} \end{aligned} \qquad (2.28)$$

where

$$\mathbf{A}(\mathbf{x}) = \begin{bmatrix} \hat{\beta}_{1r_1} L_{g_1} L_f^{r_1-1} h_1(x) & \cdots & \hat{\beta}_{1r_1} L_{g_m} L_f^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ \hat{\beta}_{mr_m} L_{g_1} L_f^{r_m-1} h_m(x) & \cdots & \hat{\beta}_{mr_m} L_{g_m} L_f^{r_m-1} h_m(x) \end{bmatrix}$$

$$\mathbf{B}(\mathbf{x}) = \begin{bmatrix} \sum_{k=0}^{r_1} \hat{\beta}_{1k} L_{\mathbf{f}}^{k} h_1(x) \\ \vdots \\ \sum_{k=0}^{r_m} \hat{\beta}_{mk} L_{\mathbf{f}}^{k} h_m(x) \end{bmatrix}$$

$\hat{\beta}_{ik}$ are scalar design parameters, $\mathbf{A}(\mathbf{x})$ is the characteristic matrix, and r_i is the relative order of the i^{th} output y_i.

The *relative order* r_i (Isidori, 1995 and Karvaris and Soroush, 1990) associated with the i^{th} output y_i of the multivariable system in (2.11) is defined as the smallest non-negative integer such that the following row matrix

$$\left[L_{\mathbf{g}_1} L_{\mathbf{f}}^{\eta-1} h_i(\mathbf{x}) \quad L_{\mathbf{g}_2} L_{\mathbf{f}}^{\eta-1} h_i(\mathbf{x}) \quad \cdots \quad L_{\mathbf{g}_m} L_{\mathbf{f}}^{\eta-1} h_i(\mathbf{x}) \right] \quad \text{..... (2.29)}$$

has at least one nonzero element at the steady state $\mathbf{x}_o$. The total relative order of the system is defined as $\sum_{i=1}^{w} r_i$.

The relative order r_i can be interpreted as the smallest order of the derivative of y_i at any time t, which explicitly depends on at least one of the components of the input vector u. In analogy with the linear system, the relative order of an output y_i is equal to the difference between the degree of the denominator polynomial and the degree of the numerator polynomial of the corresponding transfer function.

Using the Leibnitz's formula in (2.10), the row matrix in (2.29) can also be written as

$$\left[L_{ad_{\mathbf{f}}^{r_i-1}\mathbf{g}_1} h_i(\mathbf{x}) \quad L_{ad_{\mathbf{f}}^{r_i-1}\mathbf{g}_2} h_i(\mathbf{x}) \quad \cdots \quad L_{ad_{\mathbf{f}}^{r_i-1}\mathbf{g}_m} h_i(\mathbf{x}) \right] \quad \text{..... (2.30)}$$

In cases where no relative order can be defined, the system would have an infinite relative order. This means the output does not depend on the input **u**, but only on the steady state $\mathbf{x}_o$.

Further discussions on the property of relative order can be found in Isidori (1995). Lu and Bell (1994) also provided an insight into the intrinsic property and the application of the relative order for SISO affine systems.

2.4.2.1 MIMO Globally Linearizing Control (GLC) Structure

In this thesis, the decoupled input output behaviour, i.e. the i^{th} output depends on the i^{th} input only, is considered and given by the equation below.

$$\begin{aligned} \sum_{k=0}^{r_1} \hat{\beta}_{1k} \frac{d^k y_1}{dt^k} &= \nu_1 \\ &\vdots \\ \sum_{k=0}^{r_m} \hat{\beta}_{mk} \frac{d^k y_m}{dt^k} &= \nu_m \end{aligned} \quad \text{..... (2.31)}$$

Using the **u-ν** relationship in (2.27) and the state variables measured for the decoupled system, a simple proportional and integral controller for each $(\nu_i - y_i)$ pair is used to control the corresponding linear output to its set point. The control law is given by:

$$\nu_i = \hat{\beta}_{i0}\, y_i^{sp}(t) + K_{c_i}\left(y_i^{sp}(t) - y_i(t)\right) + \frac{K_{I_i}}{\tau_{I_i}} \int_0^t \left(y_i^{sp}(t) - y_i(t)\right) dt \quad \text{..... (2.32)}$$
$$(i = 1, \ldots, m)$$

The entire control structure with the external linear and internal state feedback controller is called *Multi-Input Multi-Output Globally Linearizing Control* (MIMOGLC) structure (Kravaris and Soroush, 1990) shown in *Figure 2.4* with the following closed loop transfer function:

$$\frac{y_i(s)}{y_i^{sp}(s)} = \frac{K_{C_i}\, s + K_{I_i}/\tau_{I_i}}{\hat{\beta}_{ir_i}\, s^{r_i+1} + \hat{\beta}_{i(r_i-1)}\, s^{r_i} + \cdots + \left(\hat{\beta}_{i0} + K_{C_i}\right) s + K_{I_i}/\tau_{I_i}} \quad \text{..... (2.33)}$$

From (2.33), it follows that the poles or the stability of the system depends on the design parameters. For a perfect model, the perfect control is achieved by assigning the poles to be negative infinity. However, the presence of modelling errors may limit the achievable closed loop performance, and hence may pose limitations on the

choice of the design parameters. In other words, the design parameters can be determined using the concept of pole placements in linear control theory.

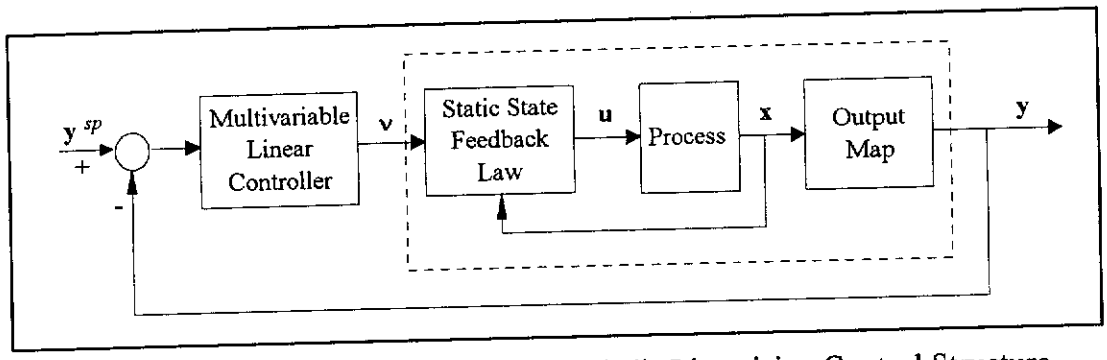


Figure 2.4: Multi-Input Multi-Output Globally Linearizing Control Structure

2.4.2.2 A Special Case of the Input Output Linearization: Generic Model Control

A special case of the input output linearization arises when the relative order of the system is equal to one. With $r_i = 1 \quad (i = 1,\ldots,m)$, the decoupled input output behaviour is described by:

$$\begin{aligned} v_1 &= \hat{\beta}_{10}\, y_1 + \hat{\beta}_{11} \frac{d\, y_1}{d\, t} \\ &\vdots \\ v_m &= \hat{\beta}_{m0}\, y_m + \hat{\beta}_{m1} \frac{d\, y_m}{d\, t} \end{aligned} \qquad \text{..... (2.34)}$$

and by setting

$$\begin{aligned} \hat{\beta}_{10} &= \hat{\beta}_{20} = \cdots = \hat{\beta}_{m0} = 0 \\ \hat{\beta}_{11} &= \hat{\beta}_{21} = \cdots = \hat{\beta}_{m1} = 1 \end{aligned} \qquad \text{..... (2.35)}$$

The input output behaviour in (2.32) becomes

$$v_i = \frac{d\, y_i}{d\, t} \qquad (i = 1,\ldots,m) \qquad \text{..... (2.36)}$$

and the control law with the PI controller is

$$v_i = K_{c_i}\left(y_i^{sp}(t) - y_i(t)\right) + \frac{K_{c_i}}{\tau_{I_i}}\int_0^t \left(y_i^{sp}(t) - y_i(t)\right) dt \quad (i = 1,\ldots,m) \quad \text{..... (2.37)}$$

Now compare (2.36) and (2.37) with the control law of GMC (Lee and Sullivan, 1988) for a decoupled system, that is,

$$\frac{d\,y_i}{d\,t} = k_{1i}\left(y_i^{sp} - y_i\right) + k_{2i}\int_0^t \left(y_i^{sp} - y_i\right) dt \quad \text{..... (2.38)}$$

where k_{1i} and k_{2i} are the performance specification parameters for y_i. It can be proved that the GMC control law in (2.38) is the same as that by the I/O shown in (2.36) and (2.37) and the closed loop transfer function yielded by using GMC is

$$\frac{y_i(s)}{y_i^{sp}(s)} = \frac{k_{1i}\,s + k_{2i}}{s^2 + k_{1i}\,s + k_{2i}} \quad \text{..... (2.39)}$$

This is equivalent to (2.33) which is obtained by using I/O. This equivalence has been established independently by To et al (1995a), and briefly described in Barolo (1994).

Zhou (1990) has developed an algorithm for appling GMC in a higher relative order system. It should be noted that there is no clear relationship between GMC and I/O when the relative order is greater than one.

2.4.2.3 Design Procedures of MIMOGLC Structure

In summary, the following steps are involved in the design procedure of a MIMOGLC system.

1. Determine the relative order r_i for each y_i
2. Compute the static state feedback law
3. Select $\hat{\beta}_{ik}$ so that the performance of the resulting linear $(v_i - y_i)$ system is satisfactory
4. Design an external linear multivariable controller for the $(v_i - y_i)$ system

More elaborate discussion on I/O for a general process with no decoupled input output behaviour is provided in Kravaris and Soroush (1990).

2.5 SYMBOLIC COMPUTATIONS

2.5.1 What are Symbolic Computations?

Computer algebra systems or symbolic algebraic computation on computers replace the traditional pencil and paper in doing mathematical computations and manipulations with keyboard and display. Being different from the basic programming language such as FORTRAN, the interactive systems allow the users to compute and program not only with numbers, but also with symbols, formulae, equations and so on. That is, exact analytical solutions can be obtained for calculations such as differentiation and matrix algebra with symbolic entries without much tedious human effort. This makes the computer algebra systems a very powerful and valuable tool for not only mathematicians, but also scientists including engineers.

The idea of computer algebra was pioneered by Lady Ada Lovelace (Haper et al, 1991) about a century and a half ago. The first computer algebra systems were written to solve specific problems in the field of applied mathematics. Since those systems were non-interactive and were available for a limited number of operating systems, the applications were restricted. The first generation of general purpose computer algebra systems were developed in early sixties. These programs, for example, REDUCE (Rayna, 1987) and MACSYMA (Pavelle and Wang, 1985), could systematically deal with unassigned variables in general purpose mathematical problems such as solving equations and differentiation. The drawbacks of these packages were prohibitive cost, computational demand and limited problem types and applications. With new developments in mathematics, computer hardware and software, a new generation of symbolic computation system has arisen. At the moment, two leading packages, namely MAPLE (Char et al, 1991) and MATHEMATICA (Wolfram, 1988), are widely used. According to Van Essen (1992), the two

programs offer roughly the same possibilities and constraints. The differences lie only in details like syntaxes and internal structure. Overall MAPLE is significantly faster for symbolic computations, having slightly more possibilities and support of mathematical procedures as well as having neater semantics and structure. On the other hand, MATHEMATICA provides a wide range of graphics functions and is famous for its documentation. Both packages are still being enhanced and expanding. Each new release is more powerful than the previous version and less different from the other. Other systems such as muMATH (Wooff and Hodgkinson, 1987), Derive (Glynn, 1989), and SCRATCHPAD (Haper et al, 1991) are also available. Comprehensive descriptions and guides for the above packages can be found in Harper et al (1991). The book also provides short reviews of other books and articles on symbolic computational systems. Heck (1993) has also provided a good introduction on several major computer algebra systems, especially MAPLE.

2.5.2 MAPLE: The Introduction

In this study, the computer algebra system, MAPLE Version V.3, is chosen to be used as the computing medium because

- the exact analytical solutions are preferred to numerical ones because of the complex nonlinear control theories involved;
- it saves time and effort in performing the algebra and mathematics involved in implementing the nonlinear control algorithm; and,
- it allows the user to study the implications and properties of solutions and different scenarios efficiently and quickly.

MAPLE V.3 was developed at the University of Waterloo in 1980 and is a powerful mathematical tool that has already found significant use in investigating and analysing problems in science and engineering. The symbolic, numeric, and graphical power that MAPLE offers can greatly assist in the mathematical aspects of any study. The internal structure of MAPLE is composed of small kernels which contain all standard mathematical operations. It also has a library of other standard mathematical

functions which are only loaded when they are explicitly needed. Apart from the kernels and the library, MAPLE contains several powerful packages of procedures written in MAPLE programming language. Among all the packages, the Linear Algebra package, *linalg*, has about 100 procedures concerning linear algebra that allow matrix manipulations, computations of rank, eigenvalues, determinant, inverse matrices and so on. It is understood that linear algebra is the essential tool in both linear and nonlinear control theory. Therefore, this package is used frequently in the implementations developed for this thesis.

Another aspect of MAPLE is its internal programming language which allows all conventional programming structures, like repetitions, conditional executions, data structures and visualisation. Hence, combining with MAPLE commands, the user is able to write procedures for specific applications. The two and three dimensional graphic capability allow MAPLE to plot functions of up to two independent variables. All the above explain why MAPLE is a suitable computing environment for the research here as well as for general chemical engineering study. As a matter of fact, the popular simulation package, SIMULINK (SIMULINK, 1992), has incorporated MAPLE as one of its many features in its latest version. In the past, mathematics was usually the major hindrance for implementing new technology in industry. However, with the availability of mega computing powers and the advanced development in symbolic computational theories, this hurdle is being gradually removed. Therefore, the computer algebra system will not only have an important role in the future of the industrial process engineering control practice, but will also provide a new aspect of mathematics to both academic and industrial practitioner.

2.5.3 Applications of Symbolic Computations in Process Control

Recently, substantial interests have been shown by scientists and engineers in the use of the symbolic computation programs for both research and industrial applications. The application of symbolic computation in control engineering has been investigated by several researchers. For example, Ogunye (1994) evaluated the application of MAPLE V as a tool for advanced control studies and several linear control theories;

Rothfuβ et al (1993) analysed observability and reachability and designed observers and controllers for nonlinear systems using the program MACNON which is based on MACSYMA; and Akhrif and Blankenship (1988) used MACSYMA to solve some control problems. The use of MAPLE in analysis and design of nonlinear control system has been illustrated by Essen (1992), Essen and Jager (1993), and Jager (1992, 1994, and 1995). In Jager (1995), a package called NONLINCON was developed for mechanical engineering applications. The package, which was based on the nonlinear control theories developed by Isidori (1995), is able to compute the zero dynamics, normal form, and the feedback law of the input output and state space linearization. Taylor and Atherley (1995) have demonstrated some of the ways MAPLE can be used in the chemical engineering curriculum and commented that programming code for MAPLE is more natural for mathematical work and produces much shorter, easier to understand, programs than FORTRAN.

An integrated MAPLE package for designing and simulating the closed loop dynamics of systems using nonlinear control theories is developed and detailed in *Appendix A*. The *appendix* addresses the usefulness of the symbolic algebraic computation for both symbolic and numerical analysis in nonlinear control using differential geometry. The procedures are developed in MAPLE V.3 environment and are based on two feedback linearizations, namely, input output linearization by Kravaris and Soroush (1990) and input state linearization by Hunt et al (1983a). Apart from computing the associated transformation for each nonlinear control theories, the procedures presented in the *appendix*, which take one step further than NONLINCON, are able to simulate the closed loop responses of chemical systems graphically by making use of the graphic capability of MAPLE V.3. As a result, different chemical systems and scenarios can be investigated with these procedures. Jager (1995) has already shown that NONLINCON can solve textbook problems successfully. Therefore, a more complicated industrial problem will be employed here to demonstrate the application of the nonlinear control package developed here and to explore the capabilities of MAPLE. To the best of the author's knowledge, no chemical engineering application has been reported using MAPLE as an integrated computing substratum in the nonlinear control area.

2.6 CONCLUSIONS

Two nonlinear control theories, input output linearization and Su-Hunt-Meyer transformation are presented in this chapter. The algorithms for determining the state feedback law and the transformation for each strategy are detailed. The MIMO system studied in this thesis are autonomous and input-linear. It was shown that the generic model control is a subset of the input output linearization for relative order one systems. For systems where the number of inputs is equal to the number of states, a simplified algorithm was proposed for the Su-Hunt-Meyer transformation to reduce the computational complexity. Finally, a brief literature review was provided on symbolic computational softwares, and MAPLE V.3 was introduced as the computing environment for implementation studies necessary for the thesis.

CHAPTER THREE

NONLINEAR CONTROL OF A SIMULATED SINGLE EFFECT EVAPORATOR PROCESS

3.1 LIQUOR BURNING PROCESS (LBP)

Alcoa of Australia at Kwinana has been operating a liquor burning facility since 1988. The objective of the operation is to remove the organic impurities of the spent caustic liquor so that the liquor can be re-used in the mainstream Bayer process. The current control strategies are purely feedback in nature and according to the technical personnel poor control performance causes significant ramifications for the downstream operation. Therefore, request was made to investigate the options available to improve the process control of the operation. Many evaporation units operate in the Kwinana refinery and Alcoa is currently designing a liquor burning plant for their Wagerup refinery. Therefore, any knowledge and insights gained from this investigation will be essential for improving the control performance of existing units and for the control strategies to be installed for the new liquor burning plant at Wagerup. This has been one of the main motivations of this thesis.

3.1.1 Process Description

The first stage of the liquor burning process is a single-effect evaporator, in which a side stream of the main caustic liquor circuit is evaporated at high recycle rates, to achieve product of specified density. Evaporator flash tank inventory is also regulated, but is of lesser importance. The process is open loop unstable but controllable. A simplified process flow diagram is shown in *Figure 3.1*. Product density and flash tank inventory are each controlled by SISO feedback loops, that use proportional and

integral (PI) action. The flowrate of the cooling water (Q_{CW}) is manipulated for density (ρ_P) regulation, while the feed liquor flow (Q_F) is adjusted for inventory control (h). The two loops tend to interact, and do not cope well with disturbances. Density deviations which impact on downstream operations are undesirable. Two main disturbances can be identified which are the variation of the product flowrate (Q_P) caused by downstream process and the reduction of recycle flow (Q_R) or heater feed flow (Q_{HF}) during a heater wash cycle. In this study, the effect of the first disturbance will be investigated because this disturbance occurs on a day-to-day basis while the second one only happens once a year during a heat wash.

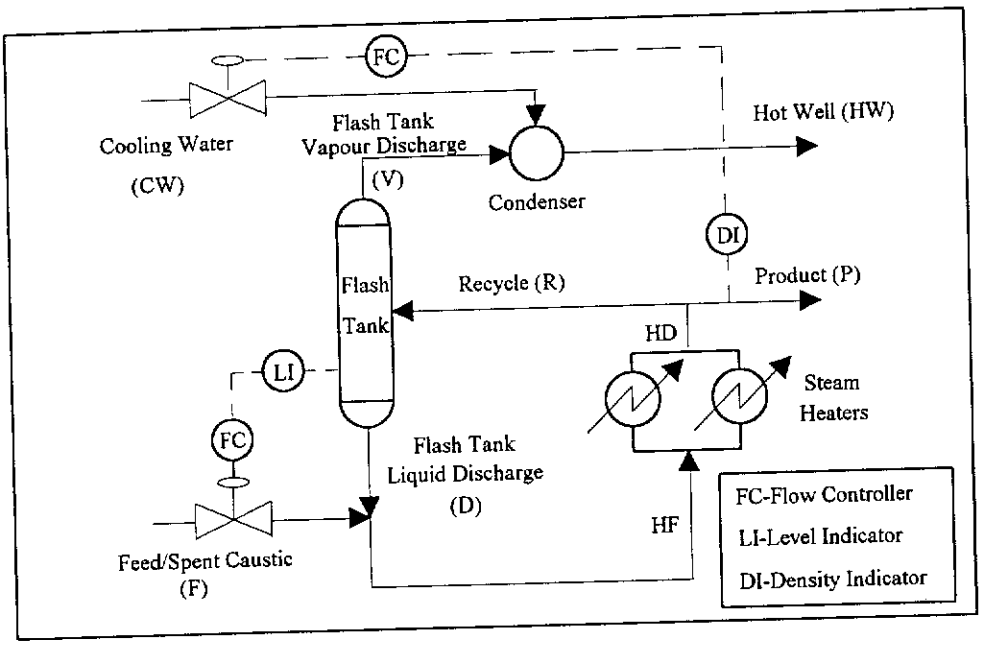


Figure 3.1: Flowsheet of Liquor Burning Process

3.1.2 Dynamic Modelling

The equations used to model the liquor burning process evaporator were derived from mass and energy balances around the flowsheet in *Figure 3.1*. The model is for a two-input two-output system (2×2) representing an idealised description of a highly nonlinear, complex unit operation. The strict validity of the equations was not established. However, the equations were considered to provide adequate descriptions of the process for the purpose of the evaluation of control strategies. Moreover, the qualitative trends and relationships revealed by the simulation of the model are considered to be realistic by plant personnel.

3.1.2.1 Assumptions

The assumptions involved in the derivation of the dynamic equations are as follows:

1. Total liquor volume in heat exchangers and piping is equivalent to a tank of area of 23.12 m^2, and height of 2.21 m. This capacity is added to the flash tank inventory.
2. The temperature of the discharge of the steam heaters is assumed to be constant at 95 °C.
3. The liquor boiling point elevation is assumed to be constant at 16 °C.
4. The hot well approach temperature is assumed to be constant at 5 °C.
5. The cooling water temperature is assumed to be constant at 40 °C.
6. The specific heat capacity of liquor is assumed to be constant at 3142 J/kg/°C.
7. Flow controller dynamics are not simulated.
8. All temperature equations are solved consistently in °C unit.
9. The product density is assumed to be equal to the flash tank liquid discharge density.
10. The process is assumed to be adiabatic.
11. The volume of the liquor remains constant in all ancillary pipe works since the pipes run full.
12. The evaporation rate is calculated on liquid water basis.
13. The setpoint is assumed to be constant at the corresponding steady state.
14. Flow ranges:

    Feed Flowrate = 0 to 50 m^3/hr

    Cooling Water Flowrate = 0 to 700 m^3/hr
15. Initial steady state conditions:

    Feed Flowrate = 25.0 m^3/hr

    Cooling Water Flowrate = 313 m^3/hr

    Flash Tank Level = 2.71 m

    Flash Tank Liquid Discharge Density = 1.762 tonne/m^3 or 1762 kg/m^3

    Flash Tank Liquid Discharge Temperature = 88.4 °C

The above assumptions are considered to be valid according to the plant personnel and reasonable in this case study. The major concern here is the temperature of the heater discharge assumed to be constant at 95 °C which, in fact, varies slightly in the actual situation. The implication of this assumption is that no heat transfer efficiency is taken into account. Hence, the mathematics of the model is simplified and the assumption is actually justified in the later chapter on the plant implementation.

3.1.2.2 Dynamic Equations

The dynamic model including three state equations with two inputs and two outputs is summarised below. Detailed derivations can be found in To et al (1995c).

1. Flash Tank Level

$$\frac{dh}{dt} = \frac{1}{A}\left(Q_F - Q_P - \dot{m}_V\right) \quad \text{..... (3.1)}$$

2. Flash Tank Liquid Discharge Density, which is assumed to be equal to the product density (ρ_P)

$$\frac{d\rho_D}{dt} = \frac{1}{Ah}\left(\frac{(Q_{HF} - Q_P)}{Q_{HF}} Q_F \rho_F - \dot{m}_V - \left(\left(Q_{HF} - Q_P - \dot{m}_V\right) - \frac{(Q_{HF} - Q_P)}{Q_{HF}}\left(Q_{HF} - Q_F\right) \right) \rho_D \right) \quad \text{..... (3.2)}$$

3. Flash Tank Liquid Discharge Temperature

$$\frac{dT_D}{dt} = \frac{1}{C_{P,D}\, Ah\, \rho_D}\left(\frac{\left(Q_{HF} - Q_P\right)\left(Q_F \rho_F + \left(Q_{HF} - Q_F\right)\rho_D\right)}{Q_{HF}} C_{P,D}\, T_{HD} - \dot{m}_V H_V - C_{P,D}\, T_D \left(\frac{\left(Q_{HF} - Q_P\right)\left(Q_F \rho_F + \left(Q_{HF} - Q_F\right)\rho_D\right)}{Q_{HF}} - \dot{m}_V \right) \right) \quad \text{... (3.3)}$$

4. Evaporation Rate Calculation

$$\dot{m}_V = \frac{C_{P,CW}\left(T_D - T_{CW} - \left(T_e + T_a\right)\right)}{H_V - C_{P,HW}\left(T_D - \left(T_e + T_a\right)\right)} Q_{CW} \quad \text{..... (3.4)}$$

Equations (3.1) to (3.4) show that the dynamic model of the liquor burning process is highly nonlinear, especially (3.3). The nonlinearity of the model will become more evident in the open loop response presented in section 3.2.1.

3.1.2.3 Selection of Input and Output Variables

The state, input and output variables are listed in *Table 3.1*. All these variables were selected based on the existing situation on site. Investigations have been carried out by Le Page (1993) to evaluate different linear control strategies using the Matlab-Simulink package (SIMULINK, 1992). *Table 3.2* lists the four configurations evaluated. A summary of this report is included in *Appendix G*.

Table 3.1: List of State, Input, and Output Variables for 2×2 LBP

State Variables		Manipulated Inputs		Outputs	
x_1	h	u_1	Q_F	y_1	h
x_2	ρ_D	u_2	Q_{CW}	y_2	$\rho_p (\approx \rho_D)$
x_3	T_D				

Table 3.2: Potential Linear Control Strategies for 2×2 LBP

Configuration	Output Variables	Input Variables
A (Existing)	Flash Tank Level	Feed Liquor Flowrate
	Product Density	Cooling Water Flowrate
B	Flash Tank Level	Feed Liquor Flowrate
	Flash Tank Temperature	Cooling Water Flowrate
	Product Density	Flash Tank Temperature
C (Feedforward with Feedback Trim by Shinsky, 1988)	Flash Tank Level	Feed Liquor Flowrate
	Cooling Water Flowrate	Ratioed to Feed Liquor Flowrate
	Product Density	Ratio of Cooling Water Flowrate to Feed Liquor Flowrate
D (similar to C)	As per C, but with Feedforward adjustment for Recycle Flowrate	Ratio factor scaled by Recycle Flowrate Factor

Two separate disturbance scenarios were investigated. They are 2 m^3/hr increase in product flowrate at time = 1000 seconds and 500 kl/hr decrease in heater feed flowrate at t = 1000 seconds. The results showed that configuration B provided the worst performance. Configurations C and D gave excellent performance for both scenarios, followed by the existing strategy A. The trial of configuration D on site was completed in the middle of 1995 (App G) and improved control performance was observed.

The reasons for using configuration A in this particular study are as follow:

- Configuration A is simple, yet it is able to provide reasonable control performance so that it can be implemented and compared easily with the nonlinear control strategies investigated in the later section.

- Configurations C and D are slightly more complicated than configuration A to be implemented in the nonlinear control framework as a result of the ratio and feedforward relationship even though they might provide better control performance.

- One of the objectives of this thesis is to compare different nonlinear control strategies with the linear theory. Therefore, it is not necessary to use the best input-output configuration, for the purpose of pure comparison here, as long as the same configuration is used throughout the entire study.

Even though configuration A is used in both linear and nonlinear control scheme in this study, the results obtained will also be compared to the control performance achieved using configuration D so as to illustrate the superiority of the nonlinear controllers.

3.2 SIMULATION RESULTS

Based on the disturbance of an increase of the product flowrate from 10 to 12 m^3/hr, the simulation results are first presented for the open loop system obtained using the local linearization and the SHM transformation. The closed loop results are then

provided in order to compare the I/O and the SHM transformation, using the local linearization as the bench mark. Finally, the GMC results are compared with those obtained using I/O.

For this simulation, the dynamic equations in section 3.1.2.2 were re-arranged in the standard form of (2.11) as shown below.

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -\frac{1}{A} \\ \frac{\bar{u}_1 (x_2 - \rho_F)}{A\, x_1\, \bar{Q}_{HF}} \\ \left(\frac{T_{HD} - x_3}{A\, x_1\, x_2}\right)\left(\frac{\bar{u}_1}{\bar{Q}_{HF}}(x_2 - \rho_F) - x_2\right) \end{bmatrix} \bar{Q}_P + \begin{bmatrix} 0 \\ -\frac{\bar{u}_1\, \bar{Q}_P (x_2 - \rho_F)}{A\, x_1\, \bar{Q}_{HF}^2} \\ \left(\frac{T_{HD} - x_3}{A\, x_1\, x_2}\right)\left(x_2 - (x_2 - \rho_F)\left(\frac{\bar{u}_1\, \bar{Q}_P}{\bar{Q}_{HF}^2}\right)\right) \end{bmatrix} \bar{Q}_{HF} \quad \text{..... (3.5)}$$

$$\mathbf{g}_1(\mathbf{x}) = \begin{bmatrix} \frac{1}{A} \\ \left(\frac{\rho_F - x_2}{A\, x_1}\right)\left(1 - \frac{\bar{Q}_P}{\bar{Q}_{HF}}\right) \\ (\rho_F - x_2)\left(\frac{T_{HD} - x_3}{A\, x_1\, x_2}\right)\left(1 - \frac{\bar{Q}_P}{\bar{Q}_{HF}}\right) \end{bmatrix} \quad \text{..... (3.6)}$$

$$\mathbf{g}_2(\mathbf{x}) = \begin{bmatrix} -\frac{C_{P,CW}\left(x_3 - T_{CW} - (T_e + T_a)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - (T_e + T_a)\right)\right)} \\ \frac{C_{P,CW}\left(x_3 - T_{CW} - (T_e + T_a)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - (T_e + T_a)\right)\right)}\left(\frac{x_2 - 1}{x_1}\right) \\ \frac{C_{P,CW}\left(x_3 - T_{CW} - (T_e + T_a)\right)}{A\, x_1\, x_2\left(H_V - C_{P,CW}\left(x_3 - (T_e + T_a)\right)\right)}\left(x_3 - \frac{H_V}{C_{P,D}}\right) \end{bmatrix} \quad \text{..... (3.7)}$$

All the responses were calculated in terms of deviation variables. The simulations were implemented using MAPLE V.3 (Char et al ,1988). *Appendix A* provides the manual for using the procedures developed here for the I/O and the SHM transformation. The code for the procedures are too long to be included in this thesis and can be found in To et al (1996b). The calling sequences and sample outputs including the transformation relationships for the SHM transformation and state feedback law for the I/O are presented in *Appendices B* and *C*. The procedures for the GMC can be easily obtained from that developed for the I/O (because the GMC is a subset of I/O).

3.2.1 Open Loop Simulation Results

Figure 3.2 shows the open loop dynamics for the height of the liquid in the flash tank, product density and liquid discharge temperature of the flash tank obtained by the local linearization and the SHM transformation. It is observed that local linearization over-estimated the values when the deviation from steady state values, that is, zero, is comparatively large. Overall, the two results are comparable. It is important to mention that the linear system obtained using first order Taylor series (or the transformed linear system in the case of the SHM transformation) were solved analytically, as described by Ray (1981). More details can be found in *Appendix A*. Moreover, the eigenvalues of the characteristic matrix of the linearized model obtained using Taylor series are 0.309×10^{-6}, -0.135 and -19.5. This implies that the system is open loop stable, except for the control of inventory of the flash tank h which is open loop unstable due to the 'zero' eigenvalue. The controllability and observability matrices show that the system is controllable and observable.

3.2.2 Closed Loop Simulation Results

3.2.2.1 Input Output Linearization versus Su-Hunt-Meyer Transformation

The closed loop responses of the liquor burning facility obtained by local linearization (Linear), I/O and SHM transformation are shown in *Figure 3.3*. The sampling time used was 0.1 hr. The PI controller tuning parameters and design parameters, $\hat{\beta}_{ik}$, in

I/O are listed in *Table 3.3*. The tuning parameters are close to the practical values which give the best performance in the simulation and in the plant and $\hat{\beta}_{ik}$ are also selected to achieve the best responses. The integral of the time-weighted absolute error (ITAE) performance indices for each control strategy is computed on the basis of 20 hours time span and are shown in *Table 3.4*. These results are discussed in section 3.3 below.

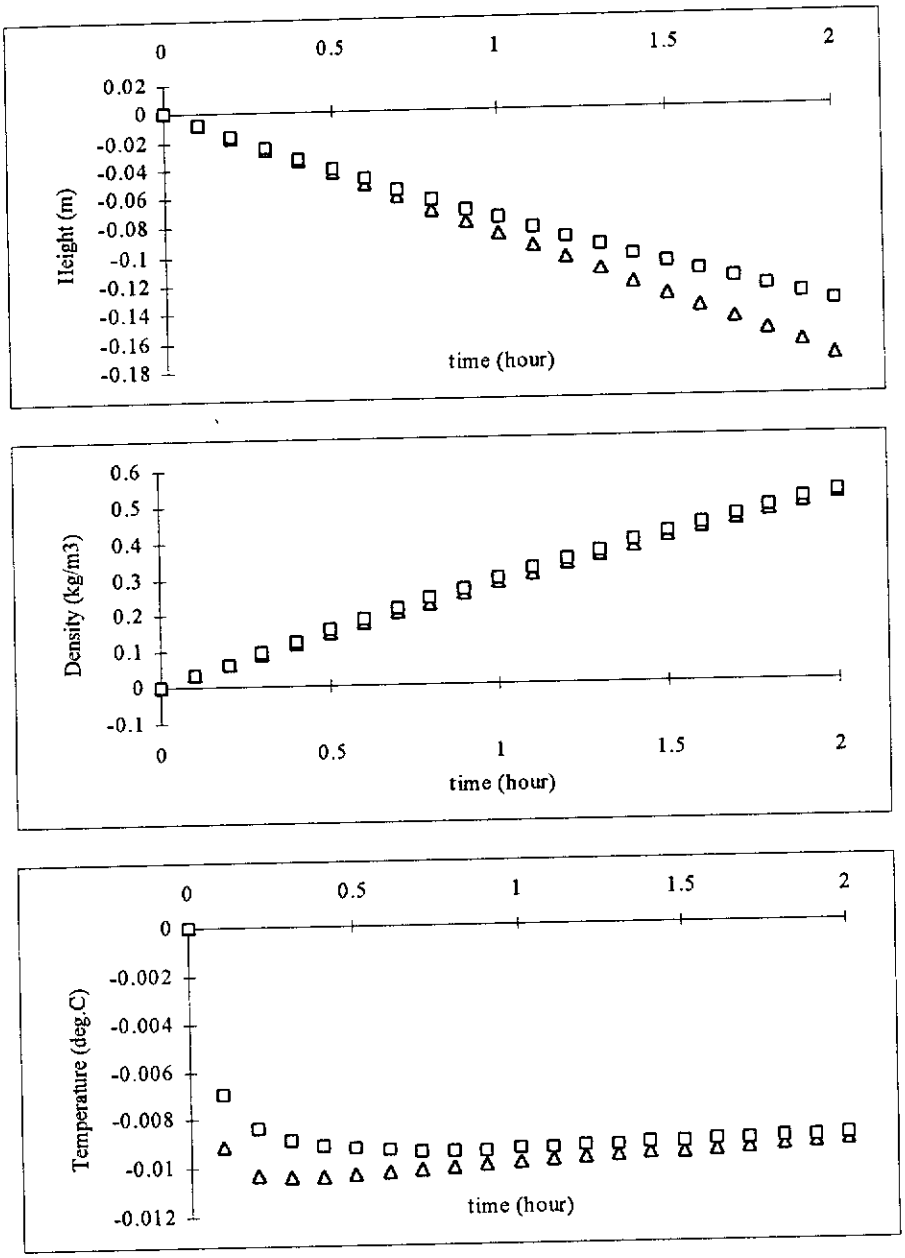


Figure 3.2: Open Loop Dynamics of 2×2 LBP by Local Linearization and SHM
(△ Linear □ SHM)

Table 3.3: PI Tuning Parameters and Design Parameters $\hat{\beta}_{ik}$ for 2×2 LBP

Design Parameters, $\hat{\beta}_{ik}$				PI Controller Parameters	
$\hat{\beta}_{10}$	1	$\hat{\beta}_{20}$	1	$K_{C1} = K_{C2}$	-10
$\hat{\beta}_{11}$	25	$\hat{\beta}_{21}$	900	$K_{I1} = K_{I2}$	-10
				τ_{I1}	20
				τ_{I2}	30

Table 3.4: ITAE of 2×2 LBP using Local Linearization, I/O and SHM

Control Strategy	Height	Density
Local Linearization	0.2082	187.2
I/O Linearization	0.0077	1.158
SHM Transformation	0.0104	35.88

Table 3.5: Design Parameters for GMC and I/O in 2×2 LBP

Generic Model Control				Input Output Linearization			
y_1	h	y_2	ρ_p	y_1	h	y_2	ρ_p
k_{11}	-15	k_{12}	-15	$\hat{\beta}_{10}$	1	$\hat{\beta}_{20}$	1
k_{21}	-40	k_{22}	-40	$\hat{\beta}_{11}$	0.5	$\hat{\beta}_{21}$	0.5
				K_{C1}	-0.5	K_{C2}	-0.03
				K_{I1}	-3.25	K_{I2}	-0.02
				τ_{I1}	1	τ_{I2}	1

Table 3.6: ITAE of 2×2 LBP using GMC and I/O

Control Strategy	Height	Density
I/O Linearization	0.0004147	0.02402
GMC	0.0008927	0.02951

3.2.2.2 Input Output Linearization versus Generic Model Control

As mentioned in Chapter 2, the GMC is a special case of the I/O. In this section, the closed loop dynamics of the liquid burning process obtained using I/O is compared with GMC results. The simulation was implemented with the sampling time of 0.01

hr. The responses of the three states and the two input variables are shown in *Figure 3.4*. The design parameters used and the ITAE indices computed based on 2.5 hours time span for each control strategy are shown in *Tables 3.5* and *3.6*, respectively. A different set of design parameters was used for I/O to investigate the effect of the parameters on the performances and the poles of the system to ensure stability.

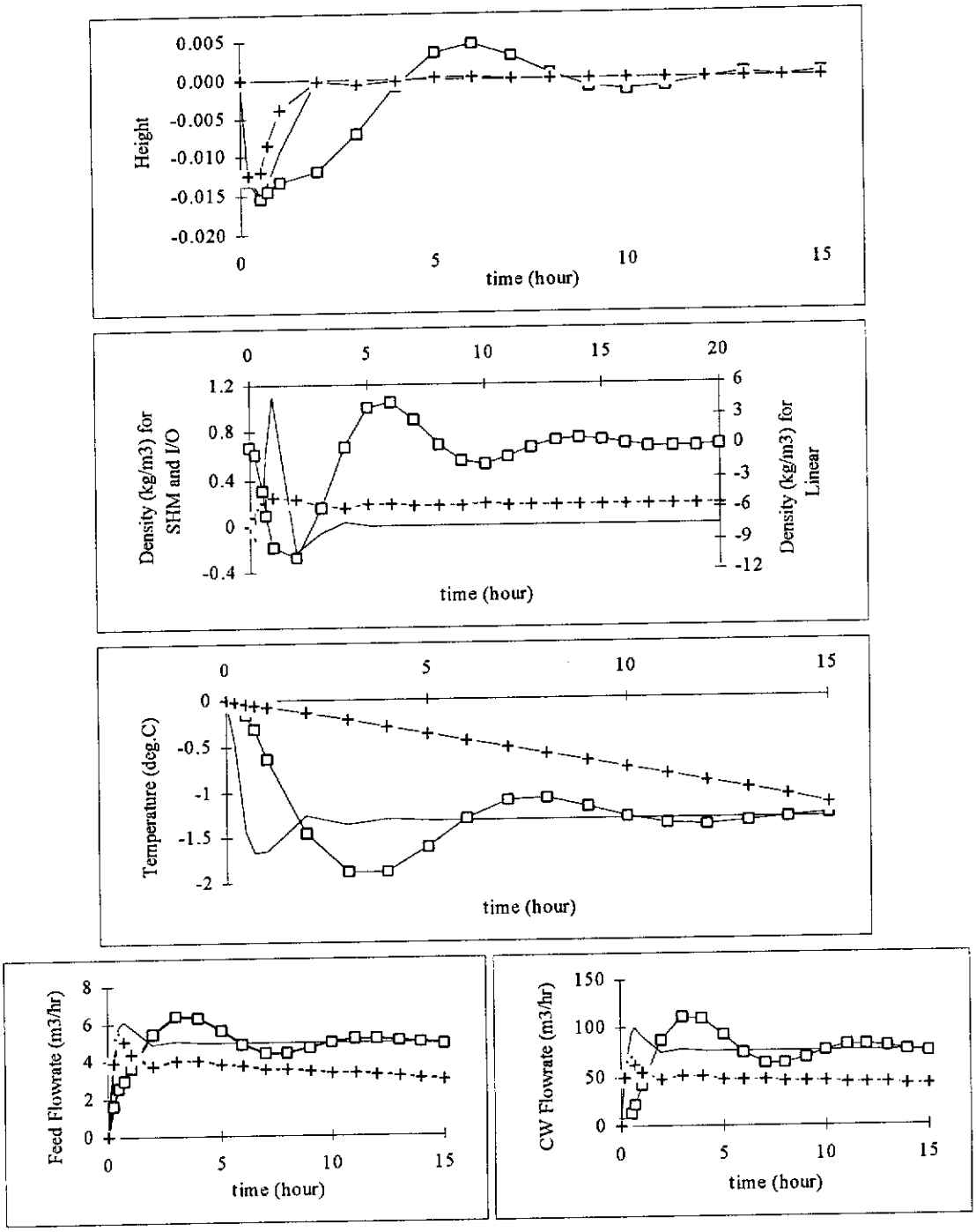


Figure 3.3: Closed Loop Dynamics of 2×2 LBP
(—□—Linear ——I/O —+—SHM)

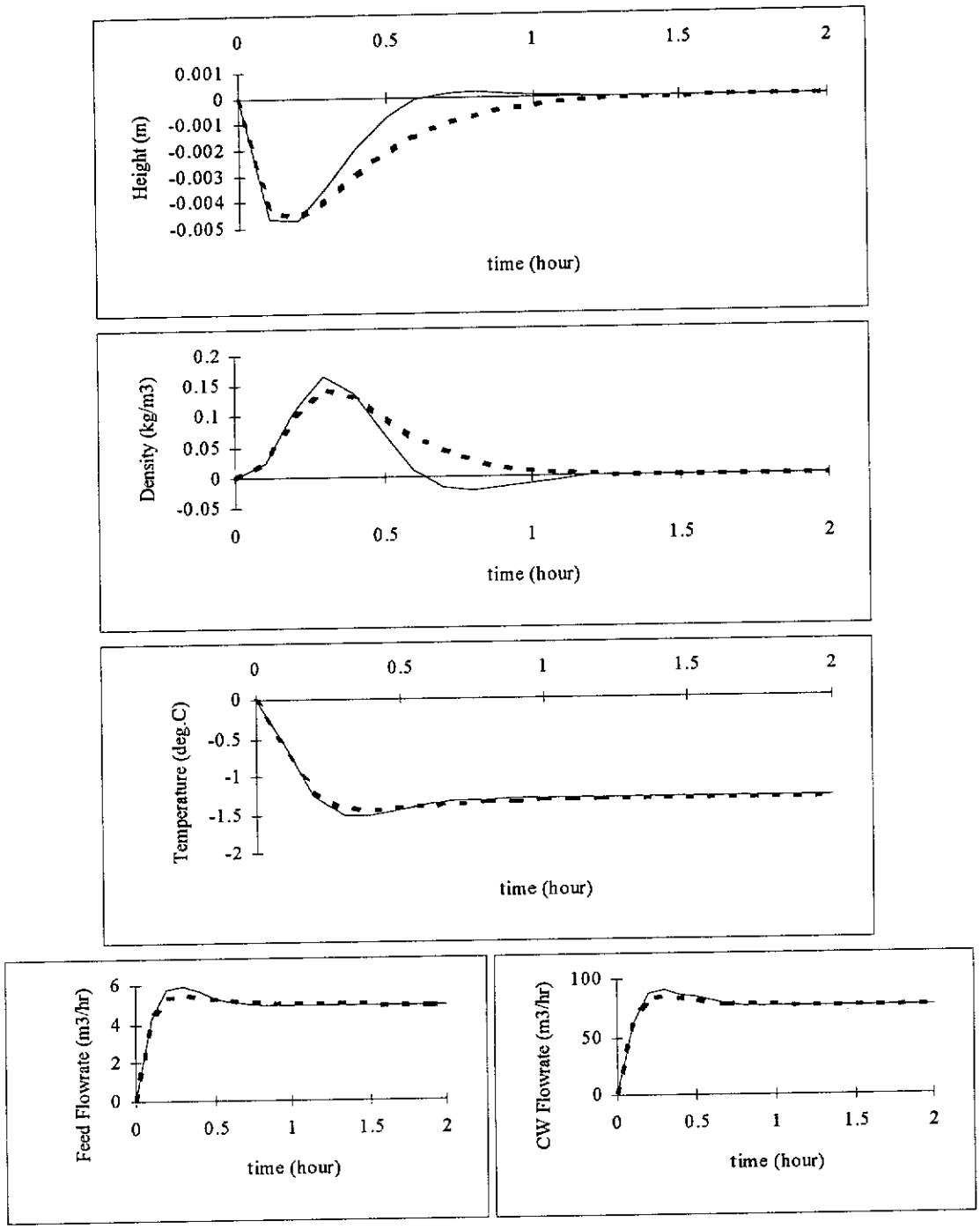


Figure 3.4: Closed Loop Dynamics of 2×2 LBP (—— I/O — — GMC)

3.3 DISCUSSIONS

3.3.1 Local Linearization

Local linearization provided a reasonable controller performance as expected from the results of Le Page (1993). It was observed that the performance of the height control can be improved by increasing the gain to a value of 40%/% while the 10%/% gain used for density control was close to the optimum value. However, the input dynamics must be investigated in using such a high gain.

3.3.2 Input Output Linearization and Generic Model Control

The control performance obtained using I/O was superior to the other two techniques based on the ITAE performance index. The calculation involved is not as complicated as for SHM transformation and can be easily handled by MAPLE. The four design parameters $\hat{\beta}_{ik}$ gave great flexibility in improving control performance. The improvements in the closed loop responses obtained by simply decreasing the values of $\hat{\beta}_{11}$ and $\hat{\beta}_{21}$ from 25 to 0.5 were obvious when comparing the simulation results in sections 3.2.2.1 and 3.2.2.2. As mentioned in Chapter 2, there is a definite relationship between the design parameters of the control law and the poles of the input output dynamics. Therefore, the corresponding poles of the system were computed and they are listed in *Tables 3.7* and *3.8*. The values of the negative poles become more negative as the design parameters were decreased, so that the system became more stable. In addition, since the system was more stable and 'linear' as a result of the transformation, the tuning of the PI controller was much easier than the traditional linear PI control strategy. The control performance provided by GMC was very similar to that for I/O since both methods are equivalent as indicated in the previous chapter. The poles computed for the height and density control loops were both -2.31 which is in between -3.10 (pole of height control loop in I/O Case 2) and -1.96 (pole of density control loop in I/O Case 2). This explains why the ITAE of the height control obtained using I/O is only half of that using GMC while similar ITAE values were obtained for the density control in both scenarios.

Table 3.7: Parameters of Height Control Loop by I/O

	$\hat{\beta}_{10}$	$\hat{\beta}_{11}$	K_{C1}	K_{I1}	τ_{I1}	Poles
I/O case 1	1	25	-10	-10	20	-2.01
I/O case 2	1	0.5	-0.5	-3.25	1	-3.10

Table 3.8: Parameters of Density Control Loop by I/O

	$\hat{\beta}_{20}$	$\hat{\beta}_{21}$	K_{C2}	K_{I2}	τ_{I2}	Poles
I/O case 1	1	900	-10	-10	30	-1.27
I/O case 2	1	0.5	-0.03	-0.02	1	-1.96

Table 3.9: Parameters of Both Height and Density Control Loop by GMC

	$\hat{\beta}_{10}$	$\hat{\beta}_{11}$	k_{11}	k_{21}	Poles
GMC	0	1	-15	-40	-2.31

It should be noted that singular points which were identified for the characteristic matrix **A(x)** in (2.29) (i.e. points at which this matrix is singular) for the I/O were well beyond the desirable operating region. Therefore, we do not need to be overly concerned about them.

Overall, it is the flexibility of the design parameters that makes the I/O control strategy better than the GMC and SHM transformation. However, extra parameters were introduced into the control system which may require tuning from time to time. Furthermore, comparing the response curves, the control performance obtained using both GMC and I/O are about 10 times better than that using configuration D in Le Page (1993).

3.3.3 Su-Hunt-Meyer Transformation

The control strategy by SHM transformation requires very involved and complicated calculations which could not be handled by MAPLE. Therefore, in this investigation, the simplified solution algorithm proposed in Chapter 2 was employed. The algorithm

requires the number of states to be equal to that of inputs; hence, the $\mathbf{f}(\mathbf{x})$ in (3.5) was simplified by expressing the disturbance explicitly as a constant input thereby converting the system to a three states, three inputs and two outputs system. This system can be easily solved by MAPLE. However, the constant disturbance input was treated as a manipulated input in the feedback law of SHM transformation. This caused an undesirable drop in the liquid discharge temperature shown in *Figure 3.2* which lead to the decrease in the values of the two manipulated inputs. In addition, it was noticed that there is a slight offset of 0.17 kg/m^3 in the density control. The obvious solution to these problems was to remove the disturbance input by appropriate control action such as a simple PI controller. In other words, a feedforward action could be introduced into the system. An improved control performance resulted when the feedforward action was implemented. Unfortunately, the product flowrate, that is, the disturbance used in this study, is determined by downstream process. Therefore, varying the disturbance is not a practical choice here.

Another solution was to replace the constant disturbance input with a real manipulated input. After consulting Alcoa, it was found that the heater discharge temperature (T_{HD}), which was assumed to be constant, can be used as the third input so that the liquid discharge temperature of the flash tank becomes the third output. The heater discharge temperature can then in turn be adjusted by varying the steam flowrate (Q_S) through the steam heaters. As a result, assumption 2 in section 3.1.2.1 can be removed. The details of this investigation will be provided in the next chapter.

3.4 CONCLUSIONS

The SHM transformation, I/O and GMC have been applied to the liquor burning process. The control strategies were compared with the classical local linearization. The open loop results using SHM transformation revealed that the performance by local linearization was compatible when the deviation is small. The simulation showed that similar control performances were obtained in the case of GMC and I/O which substantiate the theoretical finding of the fact that the GMC is a subset of the I/O as in the previous chapter. The mathematics involved in the original SHM

transformation were too complicated to be solved by MAPLE. Therefore, equations were simplified and solved using the special case proposed in Chapter 2. The preliminary investigation showed that nonlinear control theories are superior to local linearization since the later required 20 hours to return the system to its steady state while the nonlinear strategies took only 2 hours. In all cases, the responses of the two manipulated inputs were observed to be very smooth. Finally, the results showed that I/O gave the most flexible and satisfactory control performance against the disturbance specified. Further investigation is necessary for the SHM transformation with the real three inputs, three states and three outputs system proposed in section 3.3.3. The results of this investigation are shown in the next chapter.

CHAPTER FOUR

ROBUSTNESS STUDY OF NONLINEAR CONTROL THEORIES

4.1 INTRODUCTION

A robust control system is insensitive to changes in process conditions and to errors in the process model. Robustness is a very important property for a model based control system such as the nonlinear control theories studied in this thesis, because the mathematical model used to derive the control strategy or transformation relationships will never be able to provide an exact picture of the actual process. Therefore, the transformation relationships or the control strategy employed must at least be able to remain stable and converge under the adverse effect of a certain degree of uncertainty in the model, or mismatch between the model and the plant. The issue of robustness of nonlinear control theory has been addressed by many researchers in recent years. Details of literature research in this area will be given in the two chapters that follow. The robustness associated with each nonlinear control strategy will be studied by introducing an unknown parametric modelling error into the mathematical model. It is acknowledged that this type of robustness (i.e., a change in only one or more process parameters) is limited in practice. However, it is a good starting point to demonstrate the capability of the algorithms in handling the general robustness problems associated with modelling errors in the process description. The objective of this chapter is to address the robustness issue of nonlinear control theories through the simulation of closed loop responses of the evaporative stage of the liquor burning process under a model uncertainty. Moreover, the investigation in this chapter will act as a guideline for selecting the most suitable nonlinear control strategy to be tested on site. The results for this trial will be reported in Chapter 9. Further robustness study and analysis are presented in the next two chapters.

4.2 SIMULATION RESULTS ON ROBUSTNESS PROPERTY

Robustness was studied by changing the temperature factor T_f (hot well approach temperature + boiling point elevation) in the process model described in Chapter 3 from 21 to 30°C. With the same disturbance of 2 m³/hr increase in product flowrate as in the previous chapter, the closed loop responses obtained using each control strategy before and after the change of the temperature factor T_f were observed and compared according to the corresponding ITAE performance index. For the case of SHM transformation, the proposed three-input three-output system was studied. More information will be provided in the next section. The robustness issue of GMC and I/O were investigated with the two-input two-output model as described in Chapter 3. The linear control strategy was also included as a bench mark.

4.2.1 Input Output Linearization and Generic Model Control

The closed loop dynamics using local linearization was simulated with a sampling time of 0.1 hour. GMC and I/O were studied for the 2×2 system with a sampling time of 0.01 hour. The reason for using 0.1 hour for the linear controller is that the system took 20 hours to converge. Using 0.1 hour, instead of 0.01 hour, as the sampling time significantly shortened the simulation time. It was also found that there is no significant changes in performance of the linear controller when 0.01 hour was used. Moreover, the sampling time of 0.01 hour is chosen because the actual industrial setting is usually at approxmiately 30 seconds (0.083 hour). The corresponding ITAE performance indices based on 20 hours time span for local linearization and 2.5 hours for GMC and I/O are tabulated in *Tables 4.1*, *4.2* and *4.3*, respectively. The 20 hours period was used for local linearization since this was how long it took the response to reach steady state. The response curves for each case are presented in *Figures 4.1*, *4.2* and *4.3*. All design and tuning parameters are as described in *Table 3.3* for local linearization and *Table 3.5* for GMC and I/O in Chapter 3.

Table 4.1: ITAE for Local Linearization with $T_f = 21$ and $T_f = 30$

	$T_f = 21$	$T_f = 30$
Height	0.2082	0.2025
Density	187.2	281.2

Table 4.2: ITAE for GMC with $T_f = 21$ and $T_f = 30$

	$T_f = 21$	$T_f = 30$
Height	0.0008927	0.000224
Density	0.02951	0.1932

Table 4.3: ITAE for I/O with $T_f = 21$ and $T_f = 30$

	$T_f = 21$	$T_f = 30$
Height	0.0004147	0.0003096
Density	0.02402	0.1496

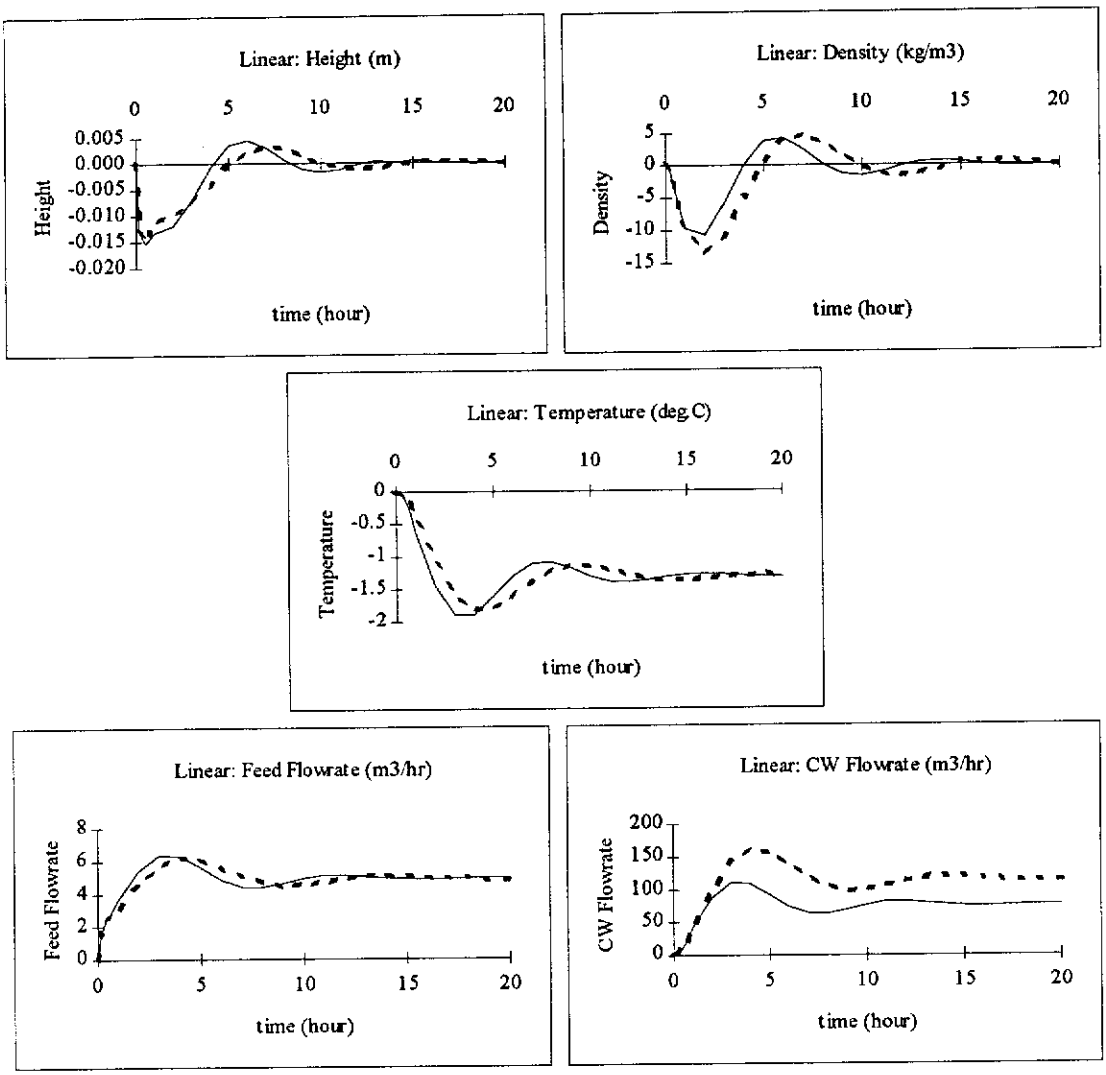


Figure 4.1: Closed Loop Dynamics of 2×2 LBP using Local Linearization
(—— $T_f = 21$ — — $T_f = 30$)

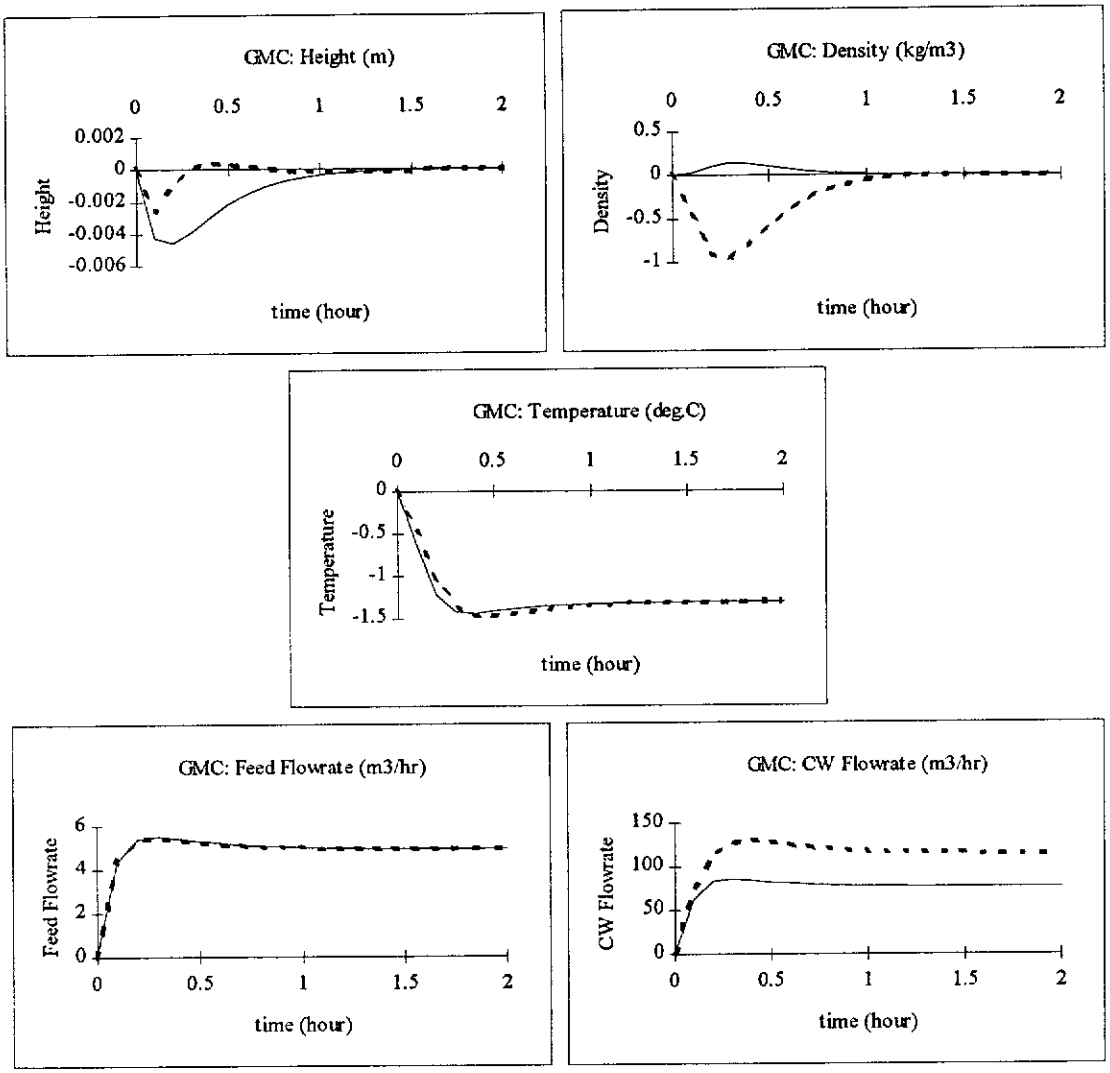


Figure 4.2: Closed Loop Dynamics of 2×2 LBP using GMC
(—— $T_f = 21$ — — $T_f = 30$)

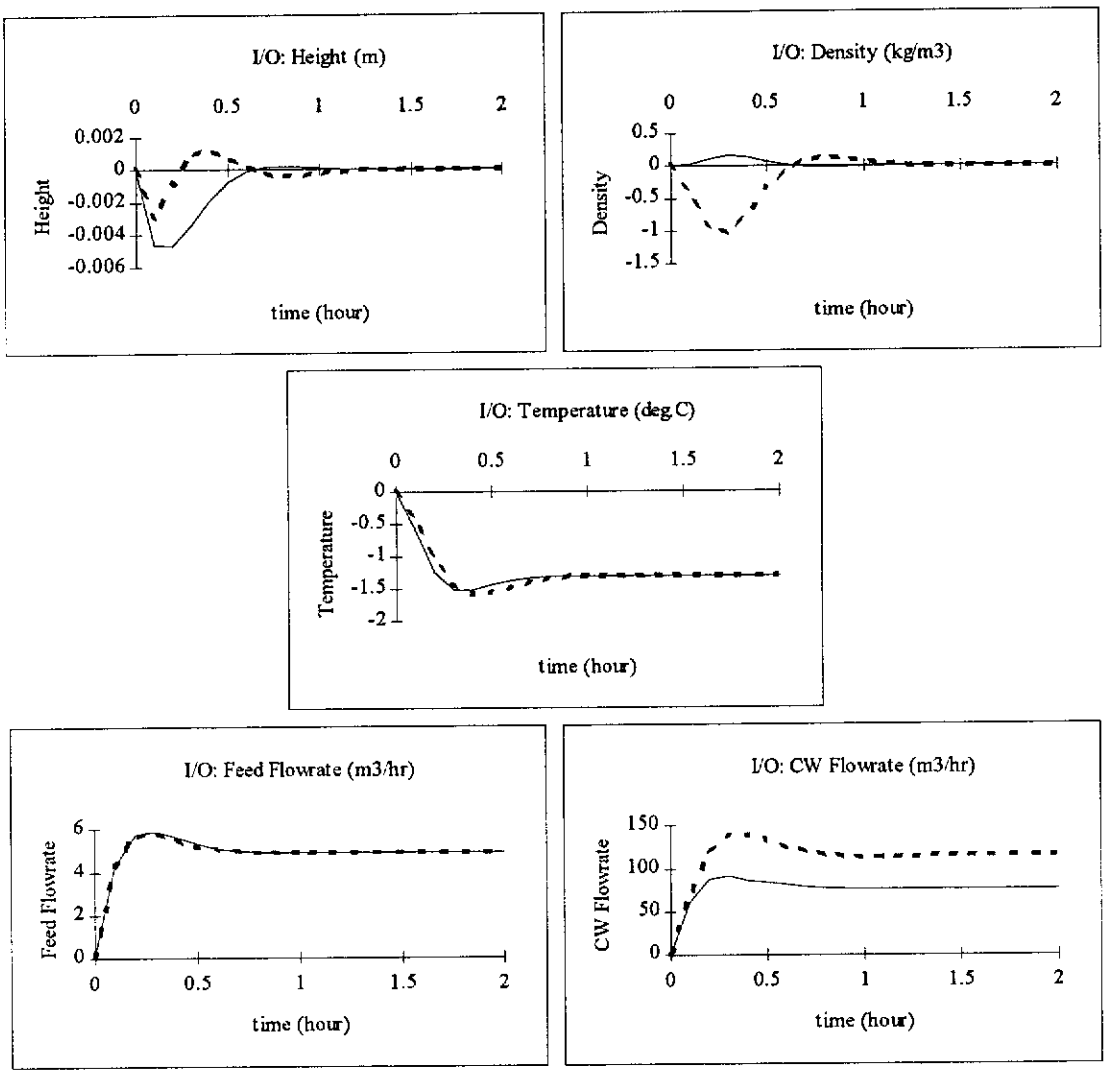


Figure 4.3: Closed Loop Dynamics of 2×2 LBP using I/O
(—— $T_f = 21$ — — $T_f = 30$)

4.2.2 Su-Hunt-Meyer Transformation

As a result of the undesirable behaviour observed in the SHM transformation for the virtual three states, three inputs and two outputs system, a real three inputs and three outputs (3×3) system was proposed. In the following, the additional mass and energy balance equations required and the simulation results are given.

The state, input and output variables involved in the proposed model are shown in *Table 4.4* below.

Table 4.4: List of State, Input, and Output Variables for 3×3 LPB

State Variables		Manipulated Inputs		Outputs	
x_1	h	u_1	Q_F	y_1	h
x_2	ρ_D	u_2	Q_{CW}	y_2	$\rho_p (\approx \rho_D)$
x_3	T_D	u_3	T_{HD}	y_3	T_D

The third manipulated input is the heater discharge temperature which is controlled by manipulating the steam flowrate through the steam heaters. The related assumptions and dynamic relationships are restated from Chapter 3 and presented as follows with additional details:

1. The volume and the mass remain unchanged as the pipes are full, that is,

$$Q_{HF} = Q_{HD}$$
$$\rho_{HF} = \rho_{HD}$$

2. The process is adiabatic and all heat is transferred via condensing steam. Hence, no sensible heat is involved.

3. The specific heat capacity of the heater feed and the heater discharge are both equal to that of the flash tank liquid discharge.

$$C_{P,HF} = C_{P,HD} = C_{P,D}$$

With the above assumptions, the equation relating the steam flowrate and the heater discharge temperature is

$$T_{HD} = T_{HF} + \frac{\dot{m}_S H_V}{Q_{HF} \rho_{HF} C_{P,D}} \quad \text{.....(4.1)}$$

The temperature and density of the heater feed are described below.

$$T_{HF} = \frac{Q_F \rho_F T_F + Q_D \rho_D T_D}{Q_F \rho_F + Q_D \rho_D} \quad \text{.....(4.2)}$$

$$\rho_{HF} = \frac{Q_F \rho_F + Q_D \rho_D}{Q_F + Q_D} \quad \text{.....(4.3)}$$

The state equations presented in section 3.1.2.2 are re-arranged in the standard form of (2.11) in Chapter 2. These equations are given below. The first order Taylor series was applied to linearize the nonlinear term $(u_1 \times u_3)$ around their steady state values before the SHM transformation was employed. The calling sequences and sample outputs can be found in *Appendices B and C*. The sampling time used was 0.01 hour which is approximately equal to the value used in the industry (Chapter 9).

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -\dfrac{\overline{Q}_P}{A} \\ 0 \\ \dfrac{1}{A x_1}\left(x_3\left(\overline{Q}_P - \overline{Q}_{HF}\right) + \overline{u}_1 \overline{u}_3 \left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right)\left(\dfrac{x_2 - \rho_F}{x_2}\right)\right) \end{bmatrix}$$

$$\mathbf{g}_1(\mathbf{x}) = \begin{bmatrix} \dfrac{1}{A} \\ \left(\dfrac{\rho_F - x_2}{A x_1}\right)\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right) \\ \left(\rho_F - x_2\right)\left(\dfrac{\overline{u}_3 - x_3}{A x_1 x_2}\right)\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right) \end{bmatrix}$$

$$\mathbf{g}_2(\mathbf{x}) = \begin{bmatrix} -\dfrac{C_{P,CW}\left(x_3 - T_{CW} - (T_e + T_a)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - (T_e + T_a)\right)\right)} \\ \dfrac{C_{P,CW}\left(x_3 - T_{CW} - (T_e + T_a)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - (T_e + T_a)\right)\right)}\left(\dfrac{x_2 - 1}{x_1}\right) \\ \dfrac{C_{P,CW}\left(x_3 - T_{CW} - (T_e + T_a)\right)}{A\,x_1\,x_2\left(H_V - C_{P,CW}\left(x_3 - (T_e + T_a)\right)\right)}\left(x_3 - \dfrac{H_V}{C_{P,D}}\right) \end{bmatrix}$$

$$\mathbf{g}_3(\mathbf{x}) = \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{A\,x_1\,x_2}\left(x_2\left(\overline{Q}_{HF} - \overline{Q}_P\right) + \overline{u}_1\left(\rho_F - x_2\right)\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right)\right) \end{bmatrix}$$

The dotted lines in *Figure 4.4* represent the responses when the temperature factor T_f equals 30 while the continuous curves represent the nominal situation with no modelling error. The tuning parameters used are shown in *Table 4.5*. The tuning parameters for the height and density control were different from those used in the two input-two output case presented in Chapter 3 because the gains used were too high and hence caused instability. The ITAE performance indices computed on the basis of 2.5 hours time span are tabulated in *Table 4.6*.

Table 4.5: PI Controller Tuning Parameters for SHM transformation

	$K_C = K_I$	τ_I
Height	-1	20
Density	-0.05	30
Temperature	-20	No integral action

Table 4.6: ITAE for SHM Transformation with $T_f = 21$ and $T_f = 30$

	$T_f = 21$	$T_f = 30$
Height	0.003594	0.0009056
Density	0.009492	0.5160

4.3 DISCUSSIONS ON SIMULATED ROBUSTNESS PROPERTY

Comparing the nonlinear control strategies with the linear control technique, the nonlinear strategies are more sensitive to modelling errors or less robust than the traditional linear technique. However, the control performance, obtained using the nonlinear control strategies, is still far better than the linear case. A detailed discussion of the simulation results is given below.

4.3.1 Input Output Linearization, Generic Model Control and Linear Control

The height control in each case was improved when the temperature factor was increased from 21 to 30°C. However, the density control, which is the primary aspect of the entire investigation, deteriorated in each case. It was observed that I/O still provided the best control performance, followed by GMC. Since both GMC and I/O are equivalent, their performances are expected to be similar. Even though the control performance obtained using the nonlinear control theories was not as good as before, it remains superior to that using the classical linear control based on the ITAE indices computed.

4.3.2 Su-Hunt-Meyer Transformation

With the three input-three output model, the control performance obtained using SHM transformation is considerably better than before and is now very comparable with the I/O and GMC. In the robustness study, the ITAE value for density control is 100 times the value before the error was introduced. For the flash tank inventory control, the performance is much better than before which corresponds to the results obtained from other control strategies. This implies that this strategy is very sensitive to modelling errors. Consequently, the SHM transformation may be unsuitable for industrial implementation. More investigation is required to explore this transformation further.

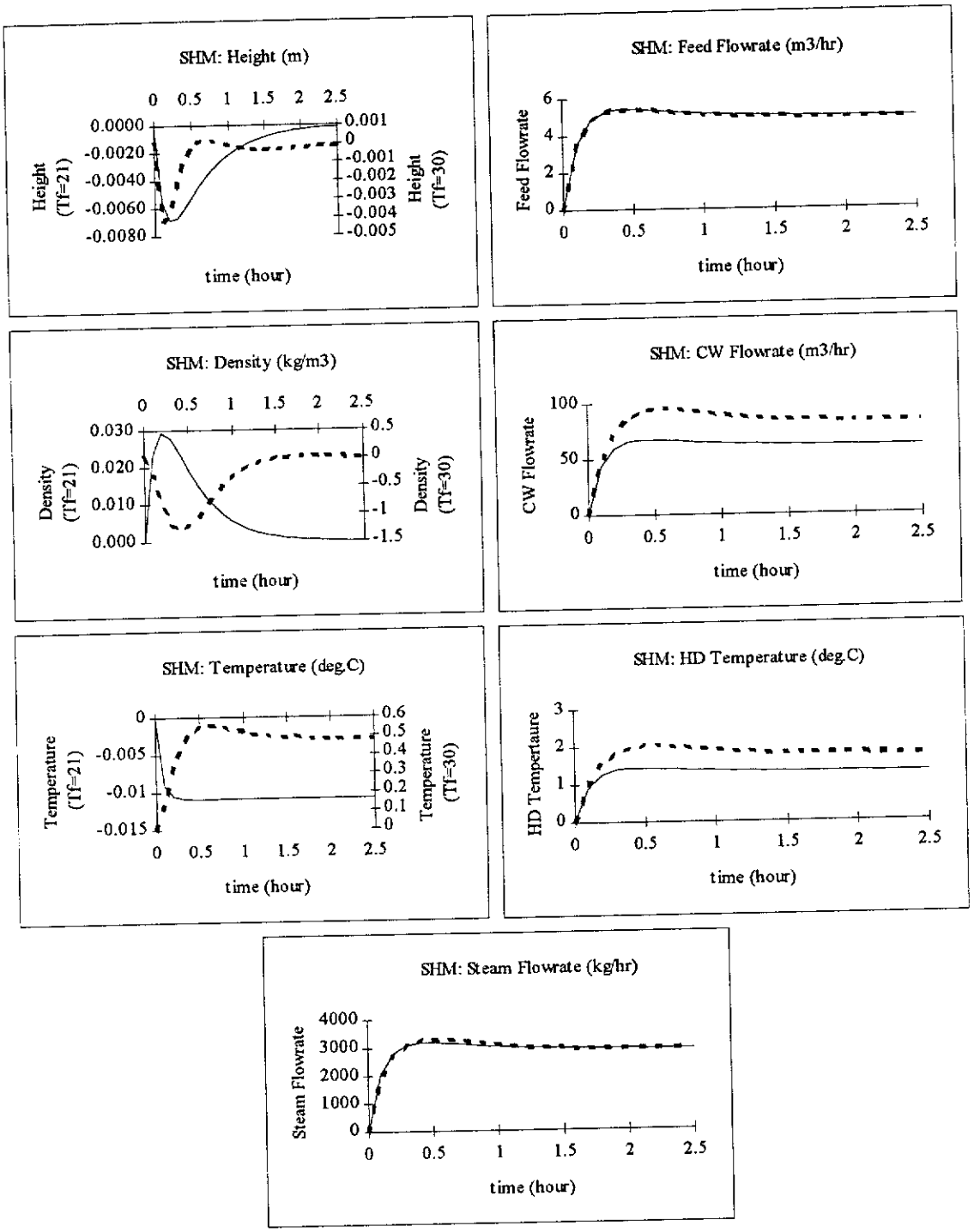


Figure 4.4: Closed Loop Dynamics of 3×3 LBP using SHM Transformation
(—— $T_f = 21$ — — $T_f = 30$)

4.4 CONCLUSIONS ON SIMULATED ROBUSTNESS PROPERTY

This chapter showed that the control performances obtained using the nonlinear control strategies deteriorated with the introduction of the modelling error. Hence, a serious robustness problem was identified with the use of each nonlinear control strategy. Since in the actual industrial situation, more modelling errors are involved in implementing the control strategies, the control performance is expected to deteriorate even more which may result in instability. It can be inferred from the single parametric error simulation here that the linear control may not be able to handle the actual system because of the longer settling time and higher ITAE indices. On the other hand, the nonlinear control may be able to provide reasonable and satisfactory control with probably a slight adjustment to the design and tuning parameters. Among the three nonlinear control strategies, it was concluded that the I/O is the most suitable scheme to be implemented on site, because of its flexibility and the excellent control performance observed in this and the previous chapter. Therefore, further investigation on the robustness of I/O is necessary and proposed adjustment procedures and analysis are given in the following chapters.

CHAPTER FIVE

ROBUST CONTROLLER SYNTHESIS USING UNCERTAINTY VECTOR ADJUSTMENT

5.1 INTRODUCTION

The robustness issue of three nonlinear control theories: I/O (Kravaris and Soroush, 1990), GMC (Lee and Sullivan, 1988) and SHM transformation by Hunt et al (1983a) were addressed in Chapter 4. The simulation results obtained using the evaporative stage of the liquor burning process showed that the control performances deteriorated when a modelling error was introduced. The ITAE performance indices were approximately 6 times the nominal ITAE for both I/O and GMC. In the case of SHM transformation, the ITAE index was approximately 54 times higher. The objective of this chapter is to propose a new compensation method for I/O and SHM transformation to improve their closed loop control performance under the effects of modelling errors, and hence, improve their robustness. Since it was shown in Chapter 2 that the GMC is a subset of the I/O, discussions on the I/O given here can equally apply to GMC.

5.2 BRIEF LITERATURE REVIEW ON MODELLING ERRORS

Modelling errors, either parametric or structural, associated with the dynamic model play an important role in determining the effectiveness of a control scheme. The most obvious effects of modelling errors on the control performance are overshoots, and longer rise and settling times. In some cases, it may lead to instability. Therefore, some adjustments must be made to the static transformation relationships in order to compensate for the adverse effects of modelling errors. However, it should be noted that, in some cases, modelling errors can have a positive effect on the closed loop dynamics. Such instance was observed in the inventory control of the liquor burning

process shown in Chapter 4.

In the past, different approaches were presented to deal with the impact of modelling errors on closed loop responses. One simple solution to modelling errors is to change the controller design or the controller tuning parameters. However, this method may not be very convenient and practical when the nature and magnitude of error is mostly unknown. A robust controller design based on the I/O was synthesised for a SISO system by Kravaris and Palanki (1988). The approach was mathematically very involved and required prior knowledge of the error structure. For the SHM transformation, Su et al (1983) gave a robustness analysis of the stability of the transformation. However, this is not practical and may be difficult to check. Zhou (1990) presented a process/model mismatch compensation for the GMC. The approach was simple and effective. However, it involved a derivative estimator which may cause instability in a noisy process unless a filter was used (Seborg et al, 1989). Other attempts have been made to design a stable robust controller for uncertain feedback linearizable systems (for example, Arkun and Calvet (1992), Chen and Leitmann (1987), Khorasani (1989), Liao et al (1991) and Ha (1989)). These approaches require restrictive conditions on the structure of the model/plant mismatch and are difficult and complex to implement in many situations. Other techniques include adaptive control and gain scheduling that update the model parameters (Seborg et al (1986), Pollard and Brosilow (1985) and Clarke et al (1987 a, b)) to minimise the effect of modelling errors. While these techniques are practical and easily implementable, they only work for a certain class of linear systems and cannot be used for the control affine system under study in this thesis.

In this work, a new approach, called uncertainty vector adjustment (UVA), is proposed for both I/O and SHM transformation in order to improve the control performance degraded by parametric modelling errors. The main advantages of the new approach are that it requires no prior knowledge of the modelling errors, and no restrictive conditions need to be satisfied. Using adjustments provided by the new approach, the control performance is improved in terms of the ITAE index,

overshoots and settling time criteria. Even though the focus of this study is on parametric modelling errors, the proposed procedure is theoretically valid for compensating structural modelling uncertainties, based on the development and characteristics of UVA.

5.3 THEORETICAL DEVELOPMENT OF UNCERTAINTY VECTOR ADJUSTMENT

As mentioned in the previous section, an adjustment must be made to the state feedback law or the transformation in order to compensate for the adverse effects of modelling errors. The required adjustment must be estimated according to the nature and magnitude of modelling errors which, unfortunately, are unknown in most circumstances. Therefore, the first task is to estimate the effects of modelling errors with the state model equations.

Given the state model equations in the standard input linear form obtained from mass and energy balances around the process,

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \sum_{i=1}^{m} \mathbf{u}_i(t)\,\mathbf{g}_i(\mathbf{x}(t)) \quad \text{..... (5.1)}$$

Suppose that there are modelling errors in the mathematical model (5.1), the values of $\mathbf{x}$ at any time t, obtained from the model will be different from the values of $\mathbf{x}_p$ measured in the plant. Consequently, at time t, $\dot{\mathbf{x}}$ will be different from $\dot{\mathbf{x}}_p$ by an uncertainty vector δ. The true model for the plant can therefore be represented by the following state equation,

$$\dot{\mathbf{x}}_p = \dot{\mathbf{x}} + \delta \quad \text{..... (5.2)}$$

where $\delta = [\delta_1 \quad \delta_2 \quad \cdots \quad \delta_n]^T$ and T denotes transpose of the vector .

Substituting (5.1) into (5.2), we obtain,

$$\dot{\mathbf{x}}_p = \mathbf{F}(\mathbf{x}) + \sum_{i=1}^{m} \mathbf{u}_i\,\mathbf{g}_i(\mathbf{x}) \quad \text{..... (5.3)}$$

where $\mathbf{F}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \delta$. At a specific time t, $\delta(t)$ can be determined using the following equation.

$$\delta(t) = \dot{\mathbf{x}}_p(t) - \dot{\mathbf{x}}(t) \quad \text{..... (5.4)}$$

In order to compute $\delta(t)$ by (5.4), $\dot{\mathbf{x}}(t)$ and $\dot{\mathbf{x}}_p(t)$ have to be calculated first. $\dot{\mathbf{x}}(t)$ can be obtained from the state model equation (5.1), but $\dot{\mathbf{x}}_p(t)$ has to be estimated from state variables $\mathbf{x}_p(t)$ measured from the plant by a simple difference equation,

$$\dot{\mathbf{x}}_p(t) = \frac{\mathbf{x}_p(t) - \mathbf{x}_p(t - t_s)}{t_s} \quad \text{..... (5.5)}$$

where t_s is the sampling time of the system.

Since (5.3) is in the standard input linear form, I/O and SHM transformation can be directly applied to it. The approach appears to be simple and practical, but it has some associated problems. A theoretical analysis is presented below to reduce the severity of these problems.

5.4 THEORETICAL ANALYSIS OF UNCERTAINTY VECTOR ADJUSTMENT

Two potential problems associated with the UVA are:

- the instability caused by the estimation of $\dot{\mathbf{x}}_p(t)$ using the difference equation in (5.5)
- the computation of $\dot{\mathbf{x}}(t)$ requires a complete simulation of the nominal process which leads to complication in programming and in actual implementation.

5.4.1 Instability caused by the Derivative Estimation

The first problem involving the estimation of the derivative $\dot{\mathbf{x}}_p$ at time t can be solved by using the property of an asymptotically stable process (Slotine and Li, 1991).

Definition 5.1:

The equilibrium state $\mathbf{x_p}(t) = 0$ is said to be asymptotically stable if, for any $R > 0$, there exists $0 < r < R$, such that if $\|\mathbf{x_p}(0)\| < r$, then $\|\mathbf{x_p}(t)\| < R$ for all $t \geq 0$ and $\mathbf{x_p}(t) \to 0$ as $t \to \infty$, where $\|\,.\,\|$ denotes the Euclidean norm on $\mathbf{R}^n$.

Essentially, asymptotic stability means that the system trajectory can be kept arbitrarily close (and eventually converges) to the origin by starting sufficiently close to it. Hence, if the system is asymptotically stable at the origin, for any $t > 0$, the mean of the states $\bar{\mathbf{x}}_\mathbf{p}$ is bounded and remains sufficiently close to the origin.

Assuming that $\mathbf{x_p}$ is asymptotically stable as defined in *Definition 5.1*, for every $\chi > 0$, there exists a small φ such that $0 < \varphi < \chi$, and $T_o = T_o(\varphi)$ where

$$\begin{aligned} &\|\bar{\mathbf{x}}_\mathbf{p}(t)\| \leq \chi && \forall\, t > 0 \\ &\|\bar{\mathbf{x}}_\mathbf{p}(t)\| \leq \varphi \quad \text{and} \quad \bar{\mathbf{x}}_\mathbf{p}(t) \to 0 && \forall\, t > T_o \end{aligned} \qquad \text{.....(5.6)}$$

This implies that for every $\varepsilon > 0$, there exists a small γ such that $0 < \gamma < \varepsilon$, and $T_p = T_p(\gamma)$ for

$$\begin{aligned} &\|\dot{\bar{\mathbf{x}}}_\mathbf{p}(t)\| \leq \varepsilon && \forall\, t > 0 \\ &\|\dot{\bar{\mathbf{x}}}_\mathbf{p}(t)\| \leq \gamma \quad \text{and} \quad \dot{\bar{\mathbf{x}}}_\mathbf{p}(t) \to 0 && \forall\, t > T_p \end{aligned} \qquad \text{.....(5.7)}$$

where $\bar{}$ indicates the mean of a variable, and T_o and T_p are referred to as the transition periods.

The transition period is the time allowed for the mean values of either the states or the time derivatives of the states to return to an arbitrary bound which is close to the origin. For any time beyond this transition period, the mean values will remain in the bound and can be approximated by zero. The idea of transition period can be related to the time constant in linear control theory. The size of the period depends entirely

on the dynamics of the process and the arbitrary values of φ and γ selected. The smaller the values chosen for φ and γ, the longer the transition period.

(5.7) allows $\bar{\dot{\mathbf{x}}}_p$ to approximate $\dot{\mathbf{x}}_p$ at any time t. When the transition period is small, T_p can be assumed to be negligible. However, when T_p is large, (5.7) can still be used because it provides a more stable, but not necessarily more accurate approximation than the difference equation in (5.5) at all time t. Therefore, it is necessary that

$$\bar{\dot{\mathbf{x}}}_p(t) \to 0 \qquad \forall\, t > 0 \qquad \text{..... (5.8)}$$

to ensure a consistently stable adjustment to the transformation relationships. It is important to point out again that the main concern here is the stability of the uncertainty vector. Another advantage of using (5.8) is that the uncertainty vector will not depend on the size of the transition period. With the foregone, the computation of the uncertainty vector in (5.4) at any time t can be rewritten as

$$\delta(t) \cong -\dot{\mathbf{x}}(t) \qquad \text{..... (5.9)}$$

As a result of the approximation given in (5.8), (5.9) can only provide an estimate of the true uncertainty vector. Therefore, parameters are introduced into (5.9) to adjust the estimated value of the uncertainty vector to ensure a better controller performance. That is,

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_n \end{bmatrix} = \begin{bmatrix} \delta_{p1}(-\dot{x}_1) \\ \delta_{p2}(-\dot{x}_2) \\ \vdots \\ \delta_{pn}(-\dot{x}_n) \end{bmatrix} \qquad \text{..... (5.10)}$$

where $\delta_{p1}, \delta_{p2}, \ldots, \delta_{pn} \in \mathbf{R}$. Equation (5.10) also implies that the uncertainty vector can be either equal to or greater than the size of modelling errors, depending on the uncertainty vector parameters used. Discussion on selection of the uncertainty vector parameters will be provided in sections 5.5.3 and 5.7.

The instability caused by the derivative estimator is eliminated by allowing $\dot{\mathbf{x}}_p(t)$ to be approximated by its mean value which tends to the origin at all time. (5.10) implies that only the model equations are required to estimate the uncertainty vector. Moreover, unlike the difference equation, the uncertainty vector estimated using (5.10) does not depend on the sampling time of the system.

5.4.2 Simulation of the Nominal Process

The UVA in (5.10) requires the simulation of the nominal process in order to compute the model $\dot{\mathbf{x}}(t)$. This will complicate the programming and make the approach very hard to implement as described below. In order to simulate the nominal process, a knowledge of the characteristics of the disturbances is required. In addition, the simulations of the nominal process must start simultaneously with the actual process and follow through at each time step so that appropriate adjustments can be estimated and applied to the actual process. Therefore, the approach becomes inflexible and impractical.

Before proposing a solution to this problem, the original model equations for $\dot{\mathbf{x}}$ are rewritten in a simpler format as in (5.11). It is important to note that no uncertainty vector is applied here.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}(\mathbf{x})) \quad \text{..... (5.11)}$$

To obtain $\dot{\mathbf{x}}$ for the estimation of uncertainty vector requires the values of the states $\mathbf{x}$, which is the solution trajectory of the above nonlinear model, and the nominal manipulated inputs $\mathbf{u}$, which is computed from the nominal transformation relationships obtained using either the I/O or the SHM transformation.

Instead of using the states $\mathbf{x}$, the actual plant state variables $\mathbf{x}_p$, is substituted into (5.11) and the nominal transformation relationships to obtain $\mathbf{u}$, and then $\dot{\mathbf{x}}$. Therefore, an approximation of $\dot{\mathbf{x}}$ can be expressed as

$$\dot{\mathbf{x}} \cong \mathbf{f}\left(\mathbf{x}_p, \mathbf{u}\left(\mathbf{x}_p\right)\right) \quad (5.12)$$

The implication of (5.12) is that the uncertainty vector now depends on two factors, the nominal state-space model, and the actual measured plant states, which makes the approach more flexible and practical.

5.5 ALGORITHM FOR UNCERTAINTY VECTOR ADJUSTMENT

The UVA is modified with the solutions proposed in the above sections. The complete algorithm for the UVA is described below.

Step 0: Obtain the nominal transformation relationships with the nominal dynamic model

Step 1: Rewrite the state model equations in terms of the uncertainty vector as shown in (5.3)

Step 2: Obtain the *adjusted* transformation relationships by applying the nonlinear control strategy to *adjusted* state equations in Step 1 directly

Step 3: Set the uncertainty vector parameters in (5.10)

At each sampling time,

Step 4: Measure the plant states $\mathbf{x}_p$

Step 5: Obtain the transformed inputs by any linear control strategy as described in Chapter 2

Step 6: Compute $\mathbf{u}(\mathbf{x}_p)$ with the plant states in Step 4 and the transformed inputs in Step 5 by the nominal transformation relationships in Step 0

Step 7: Approximate $\dot{\mathbf{x}}$ with the plant states in Step 4 and $\mathbf{u}(\mathbf{x}_p)$ computed in Step 6 by the nominal state equations

Step 8: Estimate the uncertainty vector δ by (5.10)

Step 9: Calculate the *adjusted* manipulated inputs $\mathbf{u}_{adjusted}$ for the plant by substituting $\mathbf{x}_p$, δ and the transformed inputs into the *adjusted* transformation relationships obtained in Step 2

Repeat Steps 4 to 9 until the process has returned to steady state. Note that the word *adjusted* implies that UVA is used.

5.5.1 Uncertainty Vector Adjustment and Input Output Linearization

The implementation of the UVA with the I/O is shown in *Figure 5.1*. The adjusted static state feedback law obtained by I/O with the uncertainty vector is shown below.

$$\mathbf{u}_{\text{adjusted}}(\mathbf{x}) = \mathbf{P}(\mathbf{x}) + \mathbf{Q}(\mathbf{x})\,\nu + \mathbf{R}(\mathbf{x}) \quad \text{..... (5.13)}$$

P(x), Q(x) and **R(x)** are defined as follow.

$$\begin{aligned}\mathbf{P}(\mathbf{x}) &= -\mathbf{A}_{\text{adjusted}}(\mathbf{x})^{-1}\mathbf{B}(\mathbf{x})\\ \mathbf{Q}(\mathbf{x}) &= \mathbf{A}_{\text{adjusted}}(\mathbf{x})^{-1}\\ \mathbf{R}(\mathbf{x}) &= -\mathbf{A}_{\text{adjusted}}(\mathbf{x})^{-1}\mathbf{C}(\mathbf{x})\end{aligned}$$

where

$$\mathbf{A}_{\text{adjusted}}(\mathbf{x}) = \begin{bmatrix} \hat{\beta}_{1 r_1} L_{\mathbf{g}_1} L_{\mathbf{f}+\delta}^{r_1-1} h_1(x) & \cdots & \hat{\beta}_{1 r_1} L_{\mathbf{g}_m} L_{\mathbf{f}+\delta}^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ \hat{\beta}_{m r_m} L_{\mathbf{g}_1} L_{\mathbf{f}+\delta}^{r_m-1} h_m(x) & \cdots & \hat{\beta}_{m r_m} L_{\mathbf{g}_m} L_{\mathbf{f}+\delta}^{r_m-1} h_m(x) \end{bmatrix},$$

$$\mathbf{B}(\mathbf{x}) = \begin{bmatrix} \sum_{k=0}^{r_1} \hat{\beta}_{1k} L_{\mathbf{f}}^k h_1(x) \\ \vdots \\ \sum_{k=0}^{r_m} \hat{\beta}_{mk} L_{\mathbf{f}}^k h_m(x) \end{bmatrix} \text{ and } \mathbf{C}(\mathbf{x}) = \begin{bmatrix} \sum_{k=1}^{r_1} \hat{\beta}_{1k} L_{\delta}^k h_1(x) \\ \vdots \\ \sum_{k=1}^{r_m} \hat{\beta}_{mk} L_{\delta}^k h_m(x) \end{bmatrix}$$

By substituting (5.10) in (5.11), it can be shown that when $\delta_{p1} = \delta_{p2} = \ldots = \delta_{pn} = 0$,

$$\Rightarrow \quad \delta = 0$$
$$\Rightarrow \quad \mathbf{C}(\mathbf{x}) = 0 \quad \text{and} \quad \mathbf{A}_{\text{adjusted}}(\mathbf{x}) = \mathbf{A}(\mathbf{x})$$

Therefore,

$$\mathbf{u}_{\text{adjusted}}(\mathbf{x}) = \mathbf{u}(\mathbf{x}) \quad \text{..... (5.14)}$$

and hence the original feedback law can be obtained from the adjusted law by setting

the uncertainty vector parameters to zero. The following theorems can be applied in this situation.

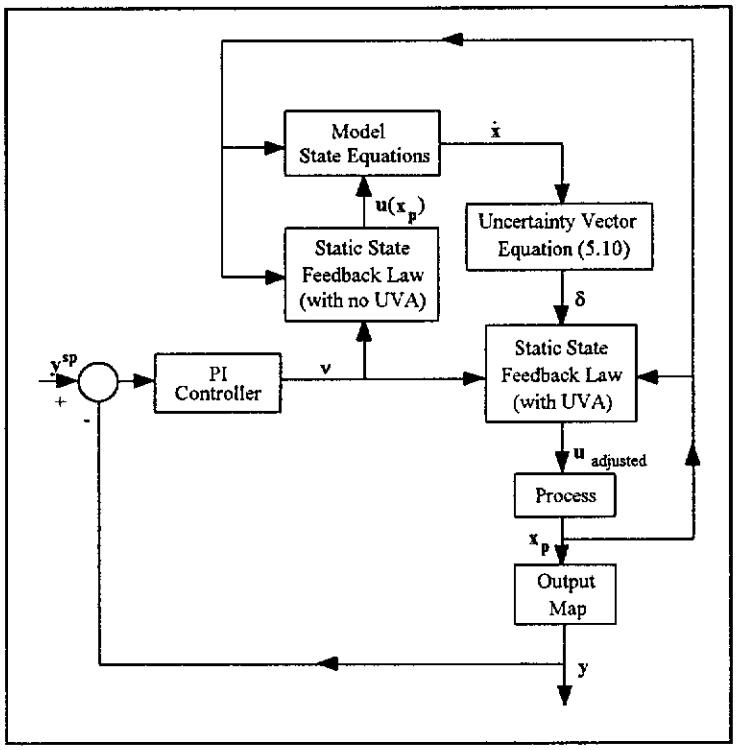


Figure 5.1: Application of UVA to I/O

Theorem 5.1:

The relative order of the adjusted system is equal to that of the nominal system.

Proof:

The relative order r_i of the i^{th} output y_i defined in Isidori (1995) is

$$\left[L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_i-1} h_i(\mathbf{x}) \quad L_{\mathbf{g}_2} L_{\mathbf{f}}^{r_i-1} h_i(\mathbf{x}) \quad \cdots \quad L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_i-1} h_i(\mathbf{x})\right]$$

which has at least a nonzero element at the steady state $\mathbf{x}_o$.

Now with the uncertainty vector,

$$\begin{aligned}&\left[L_{g_1} L_{f+\delta}^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad L_{g_2} L_{f+\delta}^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad \cdots \quad L_{g_m} L_{f+\delta}^{\hat{r}_i-1} h_i(\mathbf{x}_o)\right] \\ &= \left[L_{g_1} L_f^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad L_{g_2} L_f^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad \cdots \quad L_{g_m} L_f^{\hat{r}_i-1} h_i(\mathbf{x}_o)\right] \\ &\quad + \left[L_{g_1} L_\delta^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad L_{g_2} L_\delta^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad \cdots \quad L_{g_m} L_\delta^{\hat{r}_i-1} h_i(\mathbf{x}_o)\right]\end{aligned}$$

where $\hat{r}_i$ is the relative order of the i^{th} output in the adjusted system at $\mathbf{x}_o$. However,

$$\left[L_{g_1} L_\delta^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad L_{g_2} L_\delta^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad \cdots \quad L_{g_m} L_\delta^{\hat{r}_i-1} h_i(\mathbf{x}_o)\right] = \begin{bmatrix}0 & 0 & \cdots & 0\end{bmatrix}$$

because by definition the uncertainty vector has to be exactly zero at $\mathbf{x} = \mathbf{x}_o$. Therefore,

$$\begin{aligned}&\left[L_{g_1} L_{f+\delta}^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad L_{g_2} L_{f+\delta}^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad \cdots \quad L_{g_m} L_{f+\delta}^{\hat{r}_i-1} h_i(\mathbf{x}_o)\right] \\ &= \left[L_{g_1} L_f^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad L_{g_2} L_f^{\hat{r}_i-1} h_i(\mathbf{x}_o) \quad \cdots \quad L_{g_m} L_f^{\hat{r}_i-1} h_i(\mathbf{x}_o)\right]\end{aligned}$$

Hence, the relative order of the adjusted system has to be equal to that of the nominal system, that is, $\hat{r}_i = r_i$.

Theorem 5.2:

At the steady state ($\mathbf{x} = \mathbf{x}_o$), the adjusted characteristic matrix will be nonsingular if the nominal characteristic matrix is nonsingular.

Proof:

The characteristic matrix of the system with the UVA is defined as

$$\mathbf{A}_{\text{adjusted}}(\mathbf{x}) = \begin{bmatrix} \hat{\beta}_{1r_1} L_{g_1} L_{f+\delta}^{r_1-1} h_1(x) & \cdots & \hat{\beta}_{1r_1} L_{g_m} L_{f+\delta}^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ \hat{\beta}_{mr_m} L_{g_1} L_{f+\delta}^{r_m-1} h_m(x) & \cdots & \hat{\beta}_{mr_m} L_{g_m} L_{f+\delta}^{r_m-1} h_m(x) \end{bmatrix}$$

which is equivalent to

$$\mathbf{A}_{\text{adjusted}}(\mathbf{x}) = \begin{bmatrix} \hat{\beta}_{1r_1} L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_1-1} h_1(x) & \cdots & \hat{\beta}_{1r_1} L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ \hat{\beta}_{mr_m} L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_m-1} h_m(x) & \cdots & \hat{\beta}_{mr_m} L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_m-1} h_m(x) \end{bmatrix} + \begin{bmatrix} \hat{\beta}_{1r_1} L_{\mathbf{g}_1} L_{\delta}^{r_1-1} h_1(x) & \cdots & \hat{\beta}_{1r_1} L_{\mathbf{g}_m} L_{\delta}^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ \hat{\beta}_{mr_m} L_{\mathbf{g}_1} L_{\delta}^{r_m-1} h_m(x) & \cdots & \hat{\beta}_{mr_m} L_{\mathbf{g}_m} L_{\delta}^{r_m-1} h_m(x) \end{bmatrix}$$

That is,

$$\mathbf{A}_{\text{adjusted}}(\mathbf{x}) = \mathbf{A}(\mathbf{x}) + \begin{bmatrix} \hat{\beta}_{1r_1} L_{\mathbf{g}_1} L_{\delta}^{r_1-1} h_1(x) & \cdots & \hat{\beta}_{1r_1} L_{\mathbf{g}_m} L_{\delta}^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ \hat{\beta}_{mr_m} L_{\mathbf{g}_1} L_{\delta}^{r_m-1} h_m(x) & \cdots & \hat{\beta}_{mr_m} L_{\mathbf{g}_m} L_{\delta}^{r_m-1} h_m(x) \end{bmatrix}$$

Since the relative order remains unchanged by the introduction of the uncertainty vector and the second part of the above equation is zero when $\mathbf{x} = \mathbf{x}_0$, therefore, at the steady state, $\mathbf{A}_{\text{adjusted}}(\mathbf{x})$ is nonsingular if the nominal characteristic matrix $\mathbf{A}(\mathbf{x})$ is nonsingular, that is, the determinant of $\mathbf{A}(\mathbf{x})$ is not equal to zero.

5.5.2 Uncertainty Vector Adjustment and Su-Hunt-Meyer Transformation

The special case of SHM transformation ($n = m$) proposed in Chapter 2 is considered here. The transformation of the original states is obtained using

$$\begin{aligned} T_1 &= \int_0^{x_1} \frac{1}{g_1(1)} dx_1 \qquad \text{s.t. } x(0) = 0 \\ T_2 &= \int_0^{x_2} \frac{1}{g_2(2)} dx_2 \qquad \text{s.t. } x(0) = 0 \\ &\vdots \\ T_n &= \int_0^{x_n} \frac{1}{g_m(n)} dx_n \qquad \text{s.t. } x(0) = 0 \end{aligned}$$

Since the solutions T_i ($i = 1,\ldots,n$) depend on $[\mathbf{g}_1 \quad \mathbf{g}_2 \quad \cdots \quad \mathbf{g}_m]$ only, it is easy to see that

$$T_{i,\text{adjusted}} = T_i \qquad (i = 1,\ldots,n) \qquad (5.15)$$

However, the adjusted transformation of the manipulated inputs $T_{n+j,\text{adjusted}}\ (j=1,\ldots,m)$ is a function of the original states, the original inputs and the uncertainty vector, and can be expressed as

$$T_{n+1,\text{adjusted}} = \left\langle dT_1, (\mathbf{f}+\delta) + \sum_{j=1}^{m} \mathbf{g}_j u_j \right\rangle$$

$$T_{n+2,\text{adjusted}} = \left\langle dT_2, (\mathbf{f}+\delta) + \sum_{j=1}^{m} \mathbf{g}_j u_j \right\rangle$$

$$\vdots$$

$$T_{n+m,\text{adjusted}} = \left\langle dT_n, (\mathbf{f}+\delta) + \sum_{j=1}^{m} \mathbf{g}_j u_j \right\rangle$$

so that

$$\begin{aligned} T_{n+j,\text{adjusted}} &= \left\langle dT_i, \mathbf{f} + \sum_{j=1}^{m} \mathbf{g}_j u_j \right\rangle + \langle dT_i, \delta \rangle \quad (i=1,\ldots,n,\ j=1,\ldots,m) \\ &= T_{n+j} + \langle dT_i, \delta \rangle \end{aligned} \quad \ldots (5.16)$$

where

$$\langle dT_i, \delta \rangle = \frac{\partial T_i}{\partial x_1}\delta_1 + \frac{\partial T_i}{\partial x_2}\delta_2 + \cdots + \frac{\partial T_i}{\partial x_n}\delta_n$$

By substituting (5.10) into (5.16), it can be easily shown that

$$T_{n+j,\text{adjusted}} = T_{n+j} \qquad \text{when } \delta_{p1} = \delta_{p2} = \ldots = \delta_{pn} = 0 \quad \ldots (5.17)$$

It should be noted that, for systems with $n > m$, the adjusted transformed states are functions of the original states and the uncertainty vector, while the adjusted transformation of the inputs can be computed using (5.16). The block diagram in *Figure 5.2* shows how the SHM transformation can be implemented using the UVA approach.

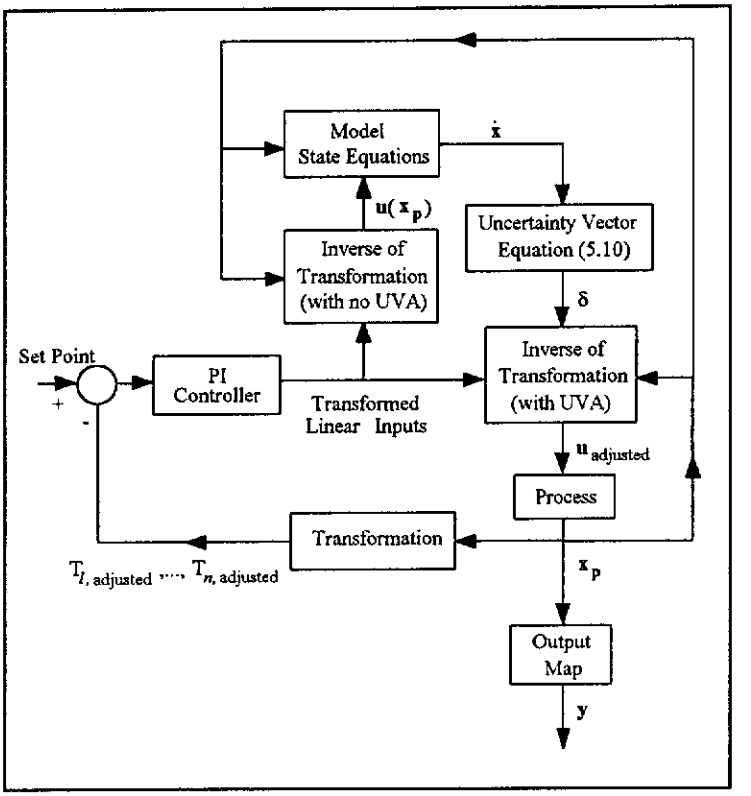


Figure 5.2: Application of UVA to SHM

5.5.3 Selection of the Uncertainty Vector Parameters

The uncertainty vector parameters are used to adjust the values of the uncertainty vector estimated so that more appropriate actions can be taken against the modelling errors. Since the parameters can be any real number, it is difficult to determine their approximate values. Suggestions are given below to assist in determining suitable values of these parameters.

Firstly, it is important to emphasis that the adjustment is removed by setting the parameters to zero. Secondly, depending on the overall effects of the modelling errors on the dynamic equations, the sign of the parameters will determine whether the adjustment is to compensate or to enhance the overall effect of the modelling errors. Therefore, it is important that the correct sign is used or the process will become unstable. Thirdly, in general, parameters with magnitudes greater than 1 will lead to better control performance. Finally, since the overall effect of modelling errors depends on the process as well as the control strategy, the choice of the

parameters will be different from one strategy to another. However, from our experience, the values of *positive one* are recommended as starting values for most cases. Further discussions on the choice of uncertainty vector parameters will be provided with illustrations in the next section.

5.6 SIMULATION RESULTS USING UNCERTAINTY VECTOR ADJUSTMENT

The target process was the evaporative stage of the liquor burning process with the product flowrate increased from 10 to 12 m^3/hr as the main disturbance, as described in Chapter 3. In this investigation, an additional modelling error was introduced into the model. Apart from the increase of temperature factor from 21 to 30°C in Chapter 4, the error of a decrease in feed density from 1300 to 1250 kg/m^3 was introduced. The nonlinear control strategies studied with the UVA were the I/O and SHM transformation. Two cases were investigated which were labelled as

- *Case 1*: the closed loop dynamics of the liquor burning process with the temperature factor as the only modelling error
- *Case 2*: the closed loop dynamics of the liquor burning process with both temperature factor and feed density as modelling errors

The effects of different uncertainty vector parameters were also investigated in both cases. The simulation results were compared with the dynamics obtained without the new approach. The performance was assessed using the ITAE index. All ITAE indices in simulations were computed on the basis of 2.5 hours time span. All responses were calculated in deviation variables. The sampling time used was 0.01 hour for all simulations.

5.6.1 Input Output Linearization

A two-input two output liquor burning process was studied as in chapter 4. The state model equations were firstly rewritten in the form of (5.3) and the vectors were,

$$\mathbf{F(x)} = \begin{bmatrix} -\dfrac{1}{A} \\ \dfrac{\bar{u}_1 \left(x_2 - \rho_F\right)}{A\, x_1\, \bar{Q}_{HF}} \\ \left(\dfrac{T_{HD} - x_3}{A\, x_1\, x_2}\right)\left(\dfrac{\bar{u}_1}{\bar{Q}_{HF}}\left(x_2 - \rho_f\right) - x_2\right) \end{bmatrix} \bar{Q}_P + \begin{bmatrix} 0 \\ -\dfrac{\bar{u}_1\, \bar{Q}_P \left(x_2 - \rho_F\right)}{A\, x_1\, \bar{Q}_{HF}^2} \\ \left(\dfrac{T_{HD} - x_3}{A\, x_1\, x_2}\right)\left(x_2 - \left(x_2 - \rho_F\right)\left(\dfrac{\bar{u}_1\, \bar{Q}_P}{\bar{Q}_{HF}^2}\right)\right) \end{bmatrix} \bar{Q}_{HF} + \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}$$

$$\mathbf{g}_1\mathbf{(x)} = \begin{bmatrix} \dfrac{1}{A} \\ \left(\dfrac{\rho_F - x_2}{A\, x_1}\right)\left(1 - \dfrac{\bar{Q}_P}{\bar{Q}_{HF}}\right) \\ \left(\rho_F - x_2\right)\left(\dfrac{T_{HD} - x_3}{A\, x_1\, x_2}\right)\left(1 - \dfrac{\bar{Q}_P}{\bar{Q}_{HF}}\right) \end{bmatrix}$$

$$\mathbf{g}_2\mathbf{(x)} = \begin{bmatrix} -\dfrac{C_{P,CW}\left(x_3 - T_{CW} - \left(T_e + T_a\right)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - \left(T_e + T_a\right)\right)\right)} \\ \dfrac{C_{P,CW}\left(x_3 - T_{CW} - \left(T_e + T_a\right)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - \left(T_e + T_a\right)\right)\right)}\left(\dfrac{x_2 - 1}{x_1}\right) \\ \dfrac{C_{P,CW}\left(x_3 - T_{CW} - \left(T_e + T_a\right)\right)}{A\, x_1\, x_2\left(H_V - C_{P,CW}\left(x_3 - \left(T_e + T_a\right)\right)\right)}\left(x_3 - \dfrac{H_V}{C_{P,D}}\right) \end{bmatrix}$$

The design and tuning parameters used were the same as those in Chapter 4 and are summarized in the following table. The effects of three different sets of uncertainty vector parameters which are listed in *Table 5.1* were studied. The nominal ITAE indices are shown in *Table 5.2* for comparison with the indices obtained in Cases 1 and 2 tabulated in *Tables 5.3* and *5.4*. The closed loop responses for Case 1 are presented in *Figures 5.3, 5.4* and *5.5*. *Figures 5.6* to *5.8* show the simulation results for Case 2.

Table 5.1: Design Parameters $\hat{\beta}_{ik}$, PI Controller Tuning Parameters and Uncertainty Vector Parameters for I/O

Design Parameters, $\hat{\beta}_{ik}$		PI Controller Parameters				Uncertainty Vector Parameters			
$\hat{\beta}_{10} = \hat{\beta}_{20}$	1	K_{C1}	0.5	K_{C2}	0.03	δ_{p1}	0	0	1
$\hat{\beta}_{11} = \hat{\beta}_{21}$	0.5	K_{I1}	3.25	K_{I2}	0.02	δ_{p2}	1	2	1
		τ_{I1}	60	τ_{I2}	60	δ_{p3}	0	0	0

Table 5.2: Nominal ITAE for Closed Loop Dynamics with No Modelling Error using I/O

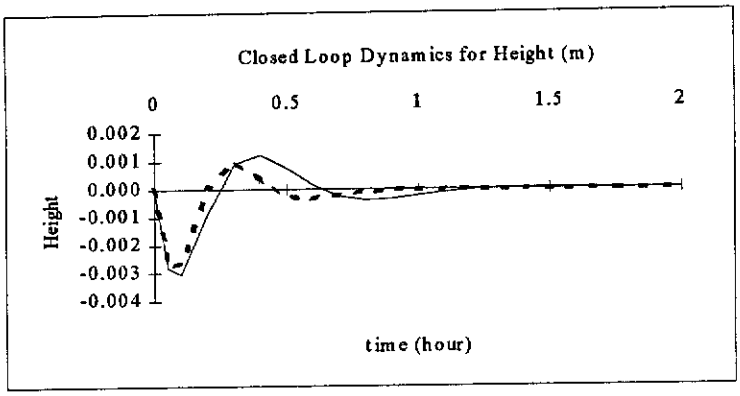
	Height	Density
No modelling Error	0.0004147	0.02402

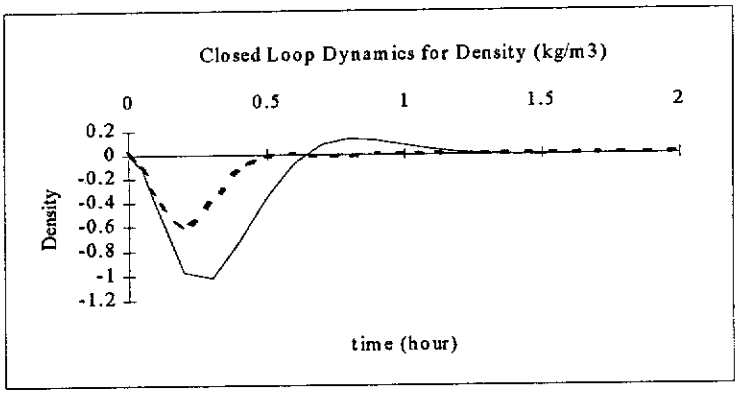
Table 5.3: ITAE for Closed Loop Dynamics with Temperature Factor as Modelling Error using I/O (Case 1)

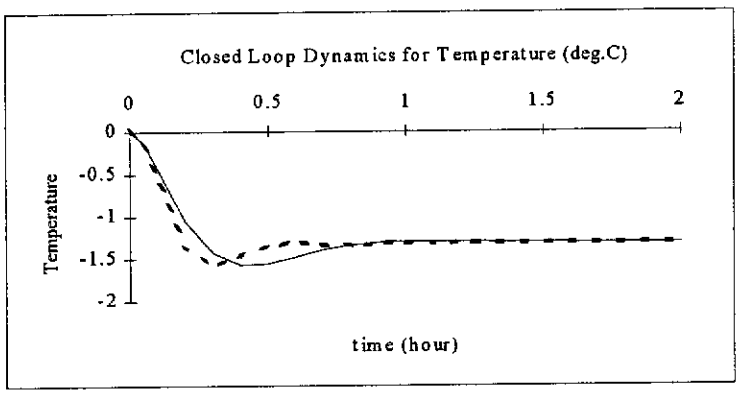
Case 1	Height	Density
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.0001405	0.03562
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 2$ and $\delta_{p3} = 0$)	0.00009470	0.02131
with uncertainty vector ($\delta_{p1} = 1$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.0001743	0.04399
without uncertainty vector	0.0003096	0.1496

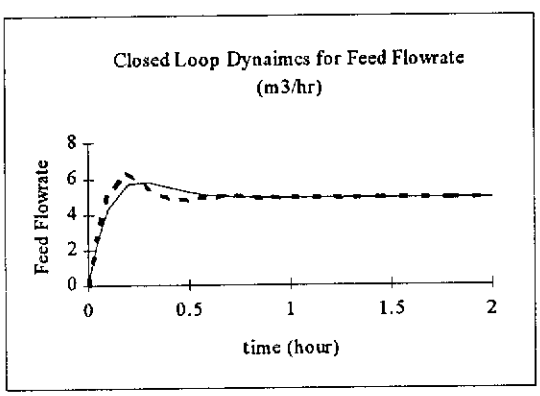
Table 5.4: ITAE for Closed Loop Dynamics with Temperature Factor and Feed Density as Modelling Errors using I/O (Case 2)

Case 2	Height	Density
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.0002116	0.05752
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 2$ and $\delta_{p3} = 0$)	0.0001496	0.03344
with uncertainty vector ($\delta_{p1} = 1$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.0001424	0.06649
without uncertainty vector	0.0004656	0.2307









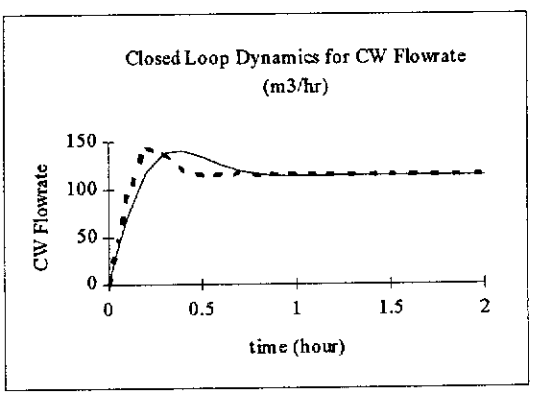
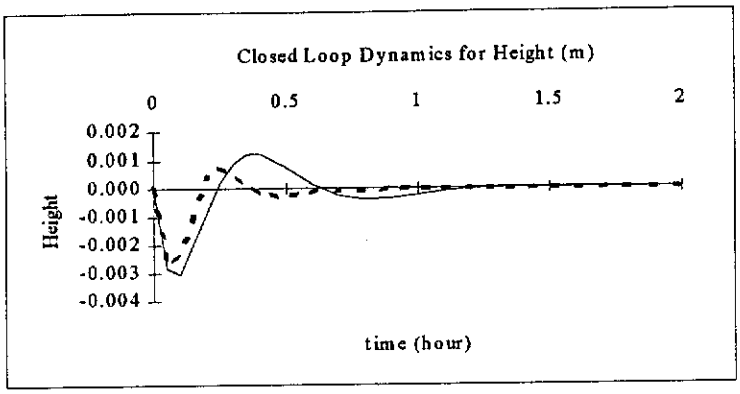
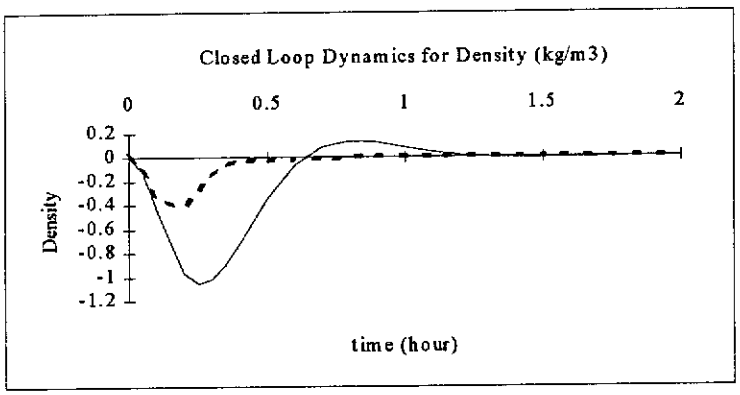
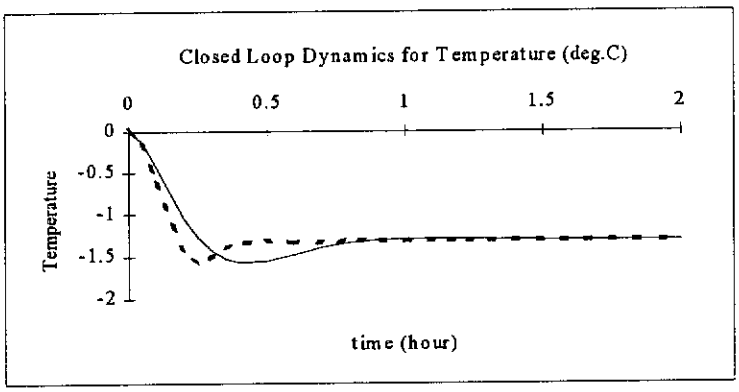


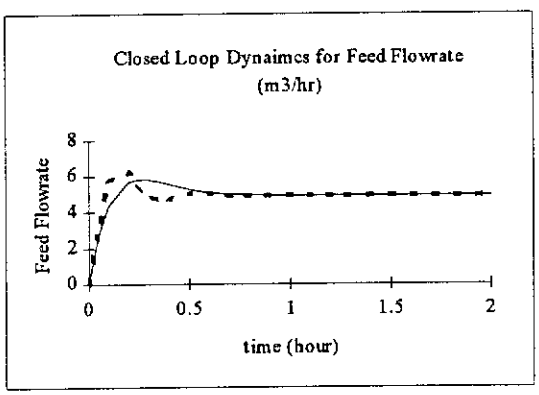
Figure 5.3: Closed Loop Dynamics for Case 1 using I/O with $\delta_p = (0, 1, 0)$

(— — with UVA ——— without UVA)









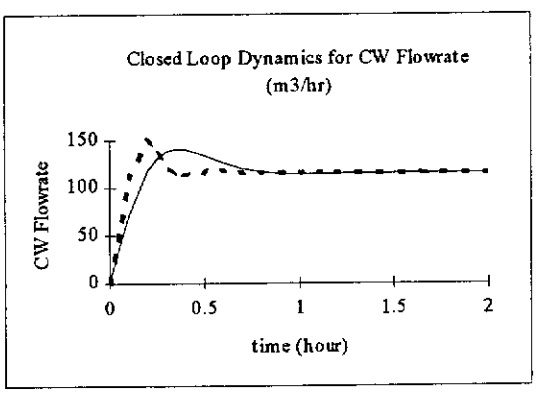


Figure 5.4: Closed Loop Dynamics for Case 1 using I/O with δ_p = (0, 2, 0)
(— —with UVA——without UVA)

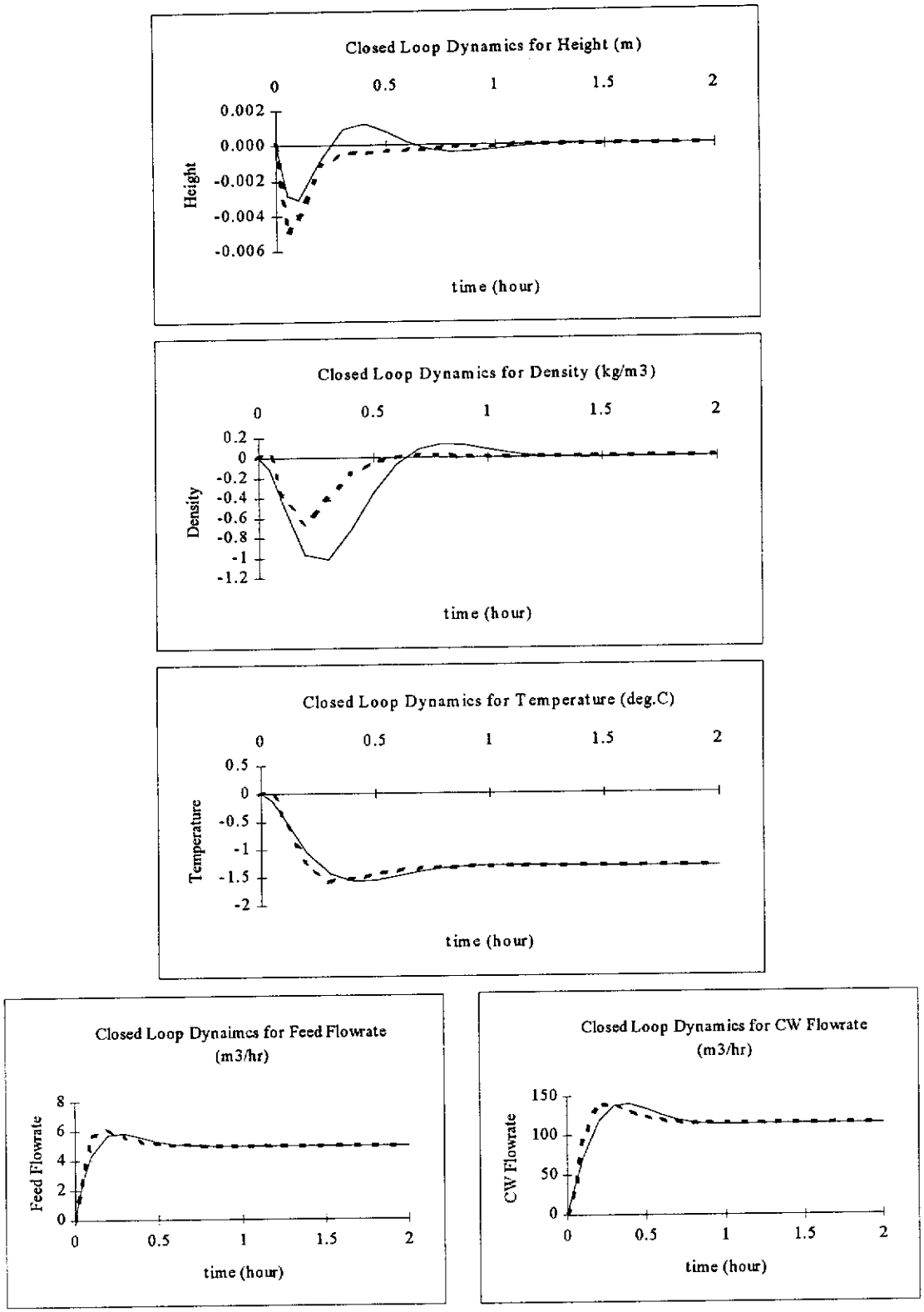


Figure 5.5: Closed Loop Dynamics for Case 1 using I/O with δ_p = (1, 1, 0)
(— —with UVA——without UVA)

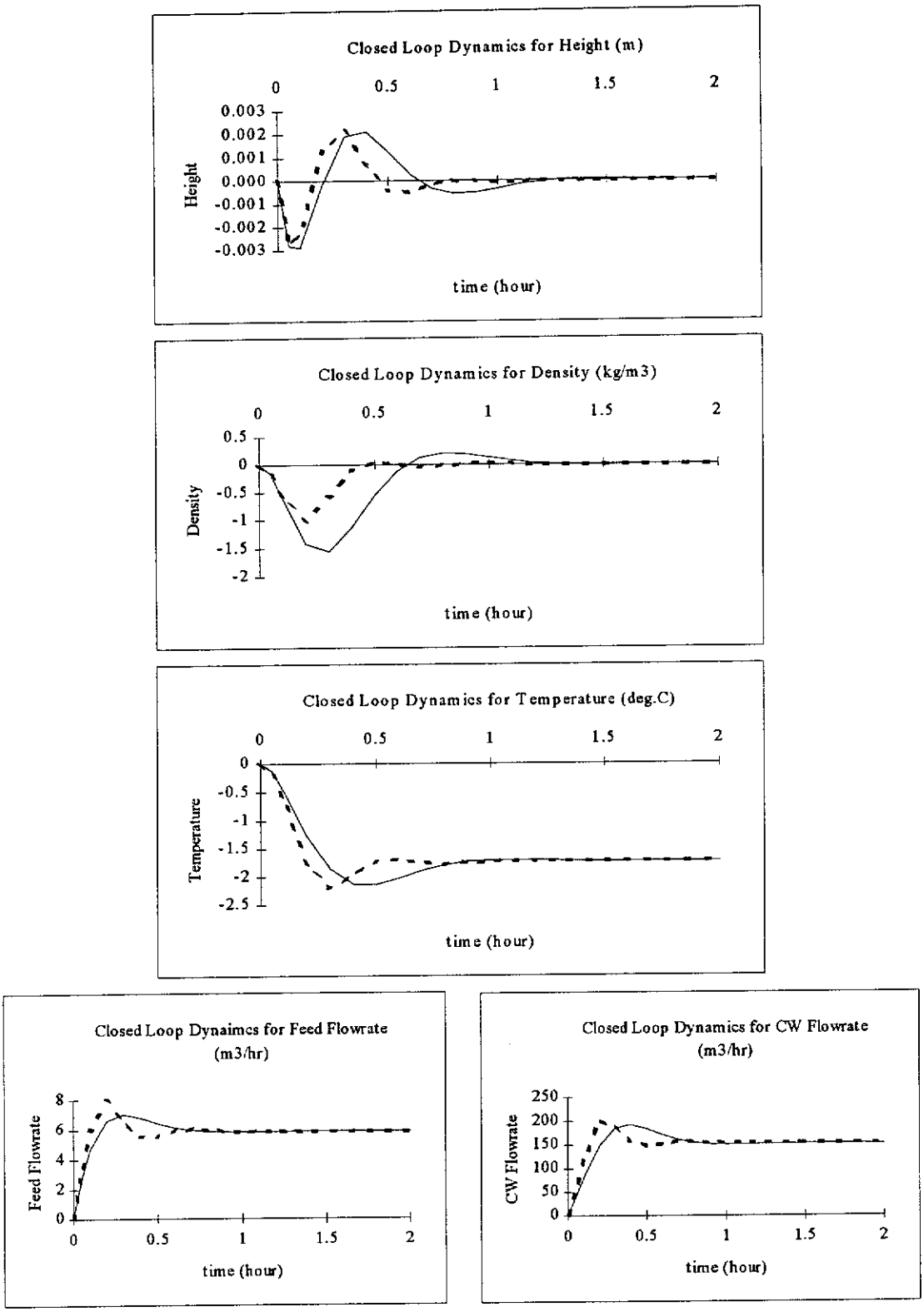


Figure 5.6: Closed Loop Dynamics for Case 2 using I/O with δ_p = (0, 1, 0)
(— —with UVA——without UVA)

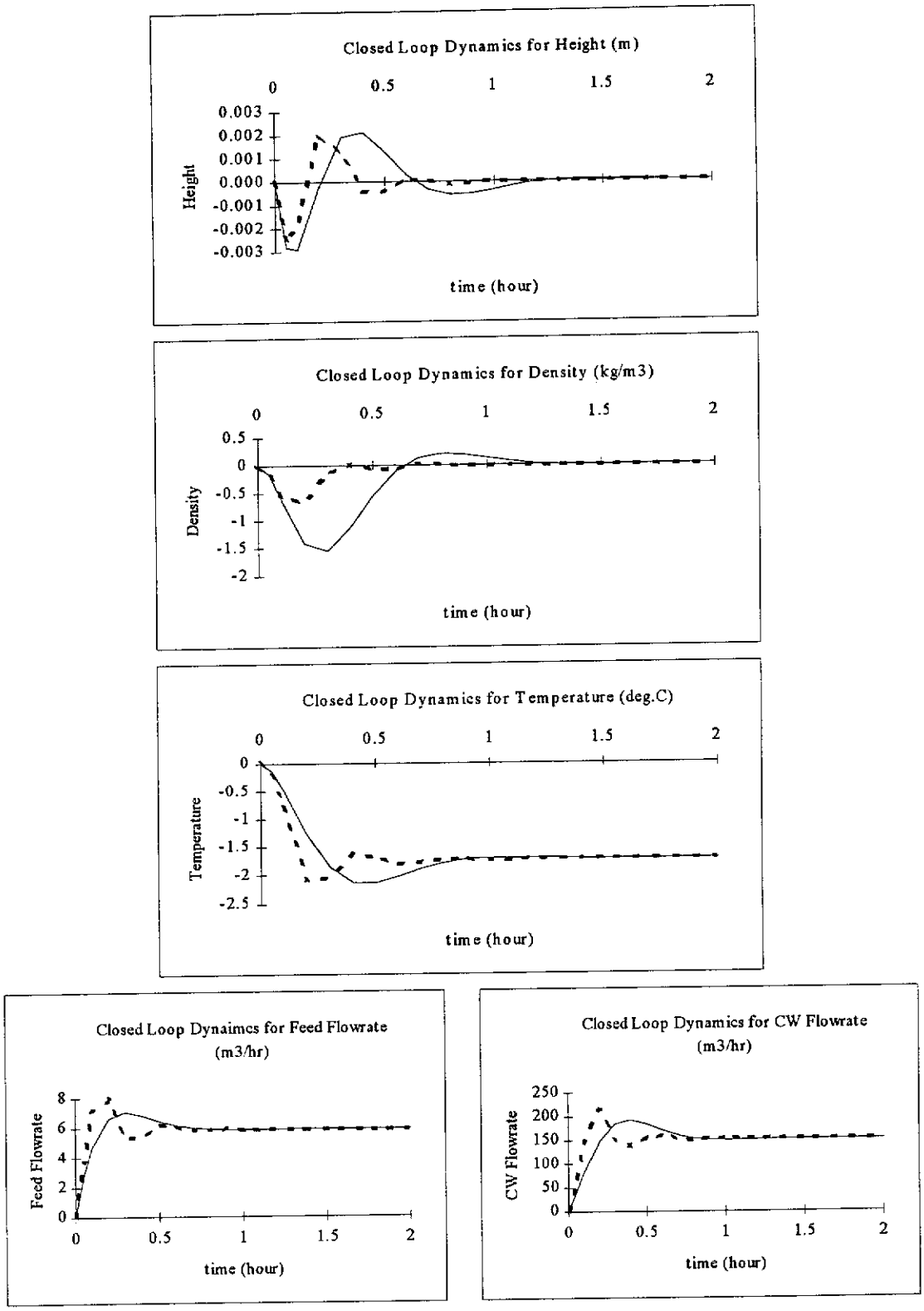


Figure 5.7: Closed Loop Dynamics for Case 2 using I/O with δ_p = (0, 2, 0)
(– – with UVA——without UVA)

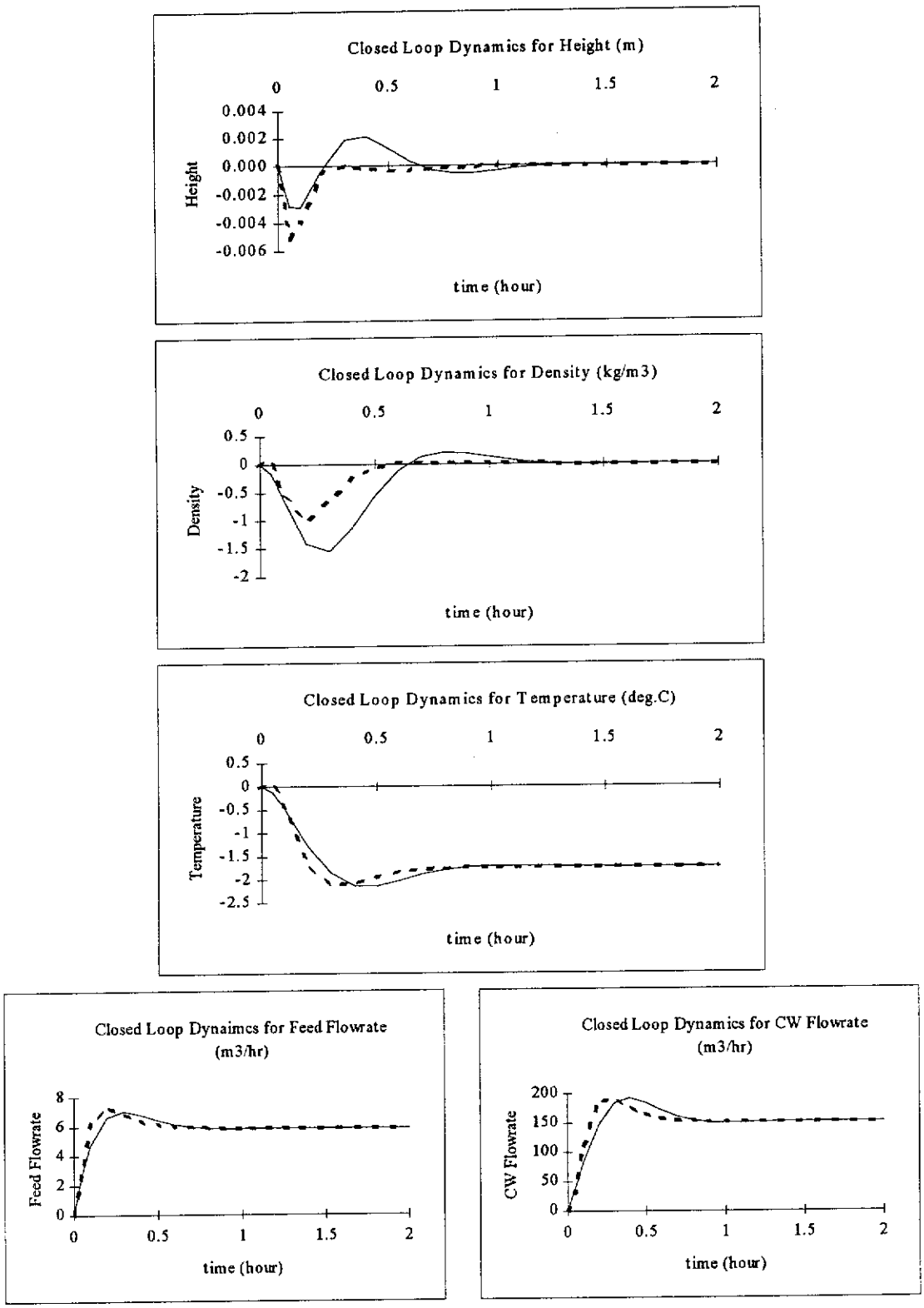


Figure 5.8: Closed Loop Dynamics for Case 2 using I/O with δ_p = (1, 1, 0)
(— — with UVA——— without UVA)

5.6.2 Su-Hunt-Meyer Transformation

A three-input three-output system was studied as described in Chapter 4. The special case of SHM transformation was investigated with and without the UVA. The state model equations were rewritten in the form of (5.3) as shown below.

$$\mathbf{F}(\mathbf{x}) = \begin{bmatrix} -\dfrac{\overline{Q}_P}{A} \\ 0 \\ \dfrac{1}{A x_1}\left(x_3\left(\overline{Q}_P - \overline{Q}_{HF}\right) + \overline{u}_1\,\overline{u}_3\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right)\left(\dfrac{x_2 - \rho_F}{x_2}\right)\right) \end{bmatrix} + \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}$$

$$\mathbf{g}_1(\mathbf{x}) = \begin{bmatrix} \dfrac{1}{A} \\ \left(\dfrac{\rho_F - x_2}{A x_1}\right)\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right) \\ \left(\rho_F - x_2\right)\left(\dfrac{\overline{u}_3 - x_3}{A x_1 x_2}\right)\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right) \end{bmatrix}$$

$$\mathbf{g}_2(\mathbf{x}) = \begin{bmatrix} -\dfrac{C_{P,CW}\left(x_3 - T_{CW} - \left(T_e + T_a\right)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - \left(T_e + T_a\right)\right)\right)} \\ \dfrac{C_{P,CW}\left(x_3 - T_{CW} - \left(T_e + T_a\right)\right)}{A\left(H_V - C_{P,CW}\left(x_3 - \left(T_e + T_a\right)\right)\right)}\left(\dfrac{x_2 - 1}{x_1}\right) \\ \dfrac{C_{P,CW}\left(x_3 - T_{CW} - \left(T_e + T_a\right)\right)}{A x_1 x_2\left(H_V - C_{P,CW}\left(x_3 - \left(T_e + T_a\right)\right)\right)}\left(x_3 - \dfrac{H_V}{C_{P,D}}\right) \end{bmatrix}$$

$$\mathbf{g}_3(\mathbf{x}) = \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{A x_1 x_2}\left(x_2\left(\overline{Q}_{HF} - \overline{Q}_P\right) + \overline{u}_1\left(\rho_F - x_2\right)\left(1 - \dfrac{\overline{Q}_P}{\overline{Q}_{HF}}\right)\right) \end{bmatrix}$$

The third manipulated input, steam flowrate, was derived from the temperature of the heater discharge. As observed in Chapter 4, the two response curves follow closely to each other. Therefore, the investigation on the closed loop dynamics of the steam flowrate was omitted. Only the response of the heater discharge temperature was studied in this research. As a result, the three manipulated inputs under investigation here are feed flowrate, cooling water flowrate and heater discharge temperature.

The same proportional and integral controller tuning parameters were used as in the previous chapter. *Table 5.5* shows the controller tuning and the uncertainty vector parameters.

Table 5.5: PI Controller Tuning Parameters and Uncertainty Vector Parameters for SHM

PI Controller Tuning Parameters				Uncertainty Vector Parameters			
$K_{C1} = K_{I1}$	1.00	τ_{I1}	20	uvp_1	0	0	1
$K_{C2} = K_{I2}$	0.06	τ_{I2}	30	uvp_2	1	2	1
$K_{C3} = K_{I3}$	20.0	τ_{I3}	0	uvp_3	0	0	0

Table 5.6: Nominal ITAE for Closed Loop Dynamics with no Modelling Error using SHM

	Height	Density
No modelling Error	0.003594	0.009492

Table 5.7: ITAE for Closed Loop Dynamics with Temperature Factor as Modelling Error using SHM (Case 1)

Case 1	Height	Density
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.0009050	0.2119
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 2$ and $\delta_{p3} = 0$)	0.0009302	0.1357
with uncertainty vector ($\delta_{p1} = 1$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.002199	0.2178
without uncertainty vector	0.0009055	0.5160

Table 5.8: ITAE for Closed Loop Dynamics with Temperature Factor and Feed Density as Modelling Errors using SHM (Case 2)

Case 2	Height	Density
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.0005319	0.3478
with uncertainty vector ($\delta_{p1} = 0$, $\delta_{p2} = 2$ and $\delta_{p3} = 0$)	0.0003559	0.2240
with uncertainty vector ($\delta_{p1} = 1$, $\delta_{p2} = 1$ and $\delta_{p3} = 0$)	0.001730	0.3587
without uncertainty vector	0.0013934	0.8712

The above tables show the ITAE indices for the nominal case, cases 1 and 2 obtained by SHM transformation. The closed loop responses for these two cases are shown in *Figures 5.9* to *5.14*.

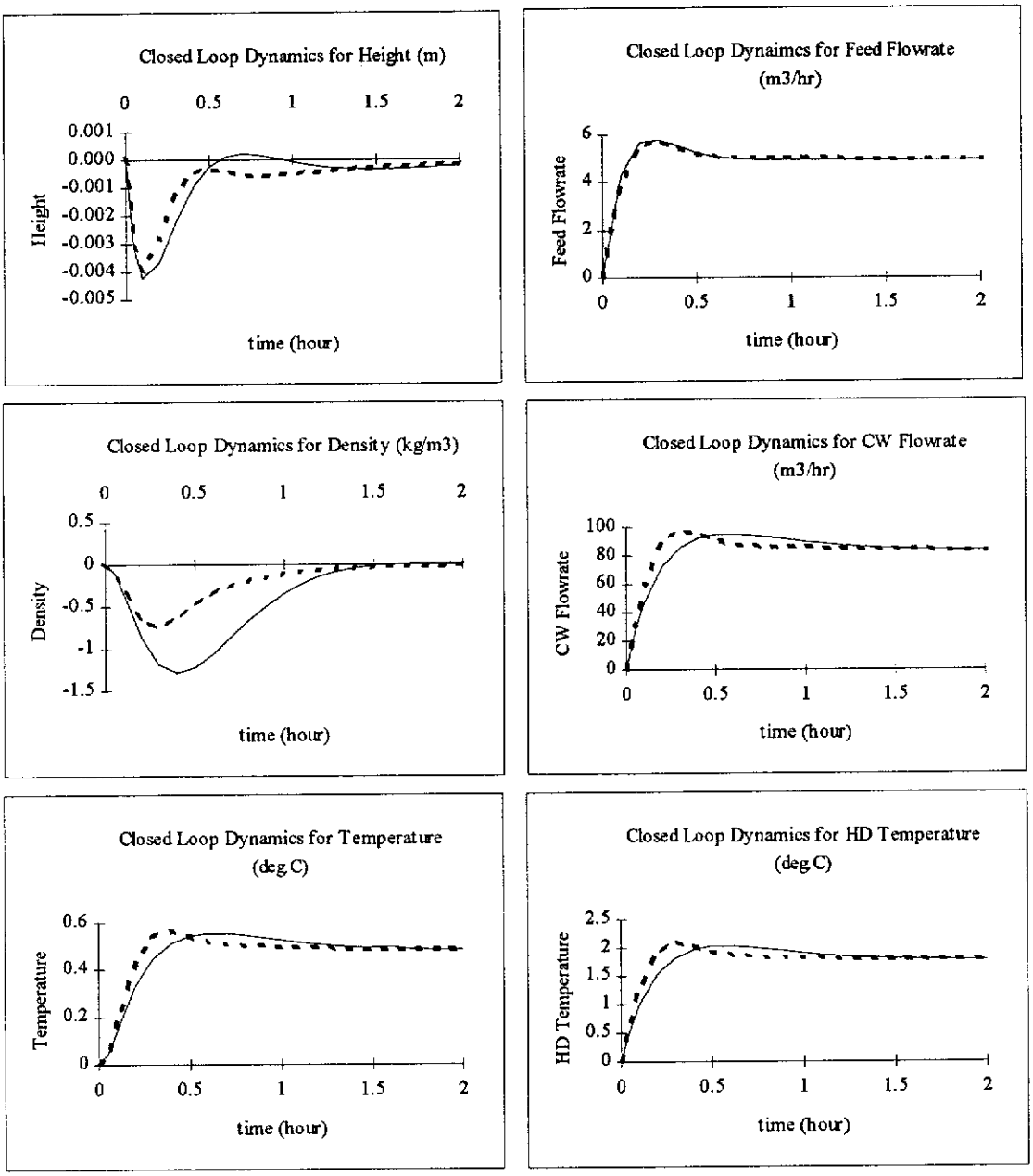


Figure 5.9: Closed Loop Dynamics for Case 1 using SHM with δ_p = (0, 1, 0)
(— — with UVA ——— without UVA)

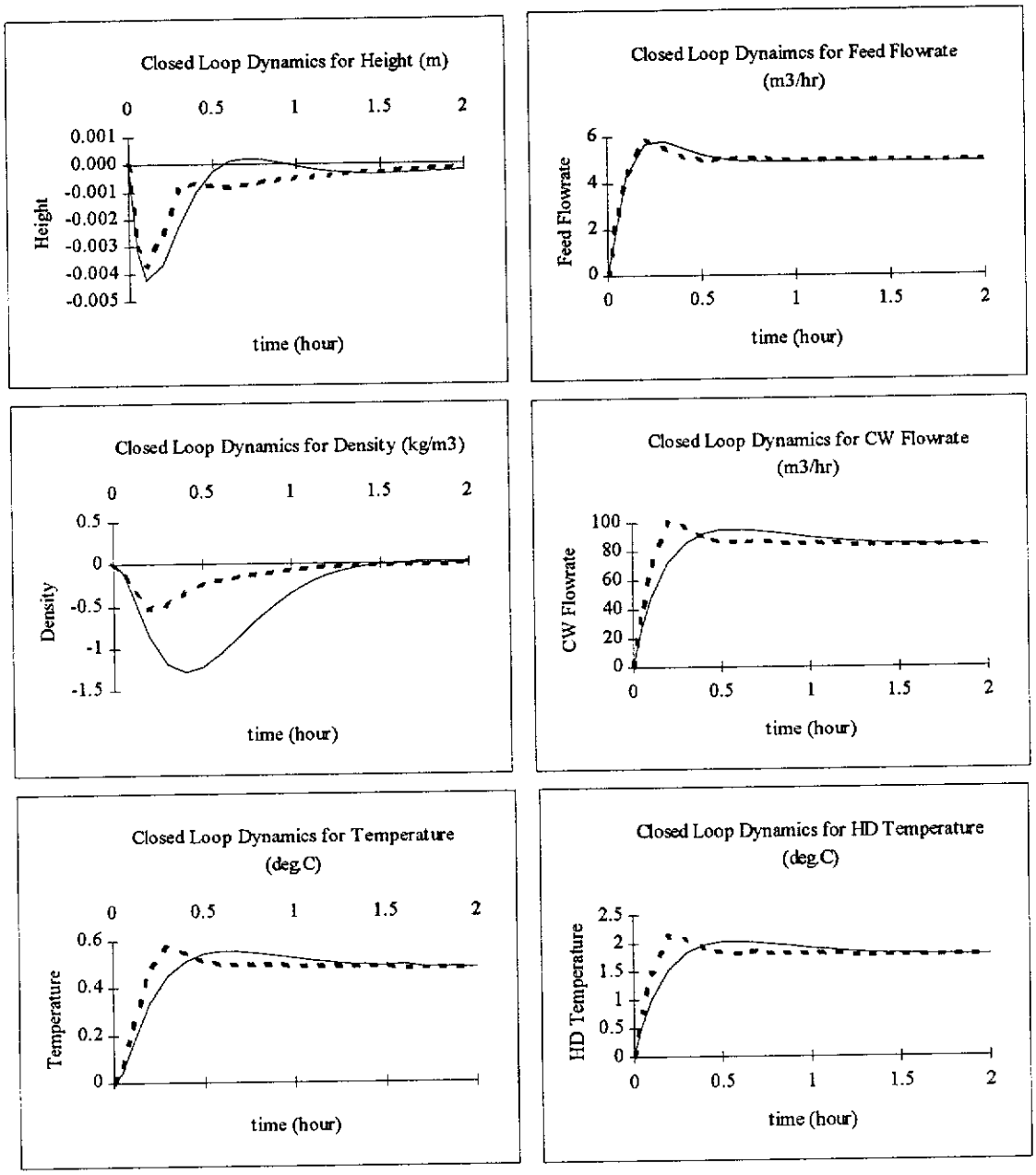


Figure 5.10: Closed Loop Dynamics for Case 1 using SHM with δ_p = (0, 2, 0)

(— — with UVA——— without UVA)

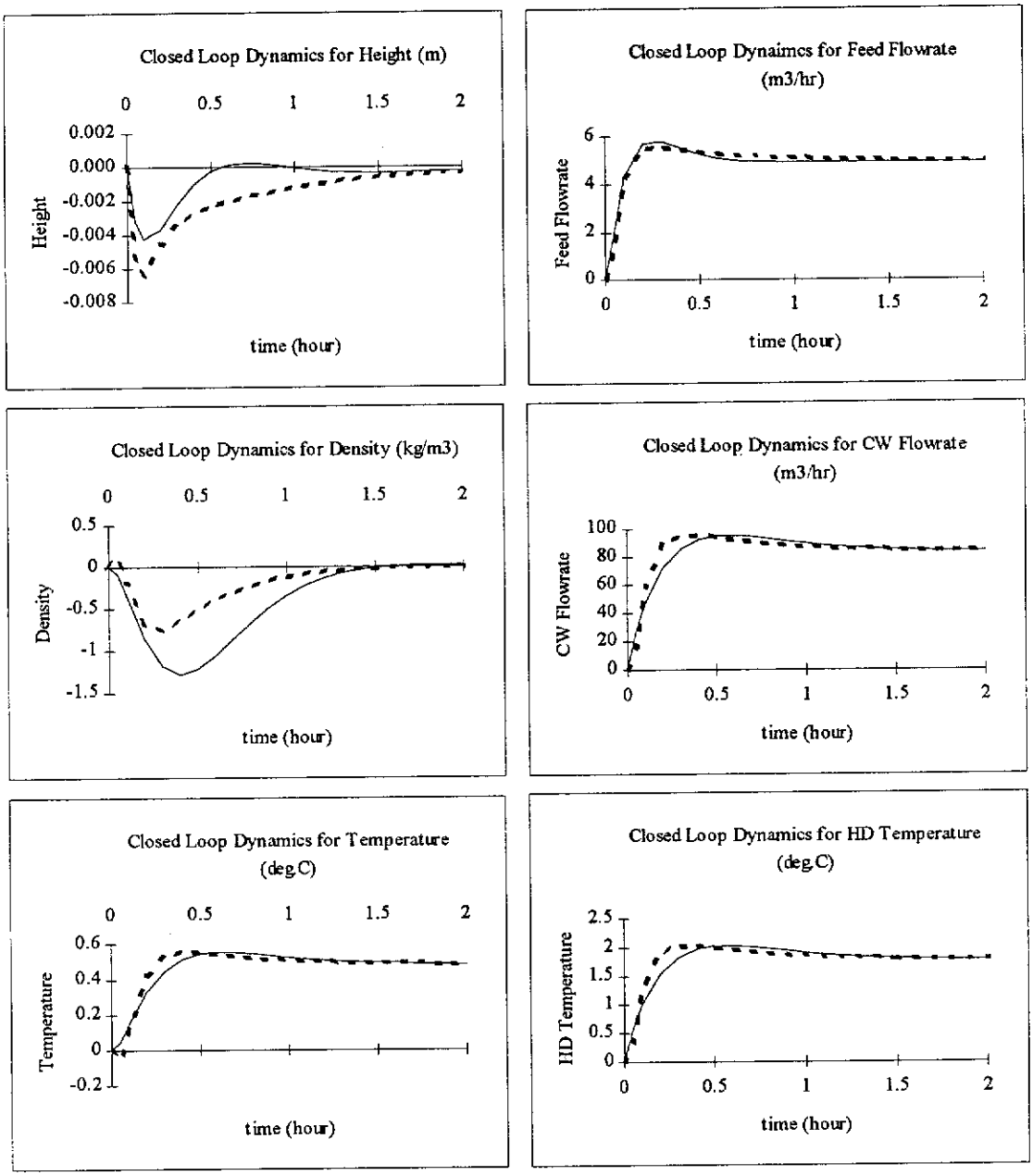


Figure 5.11: Closed Loop Dynamics for Case 1 using SHM with δ_p = (1, 1, 0)
(— — with UVA——without UVA)

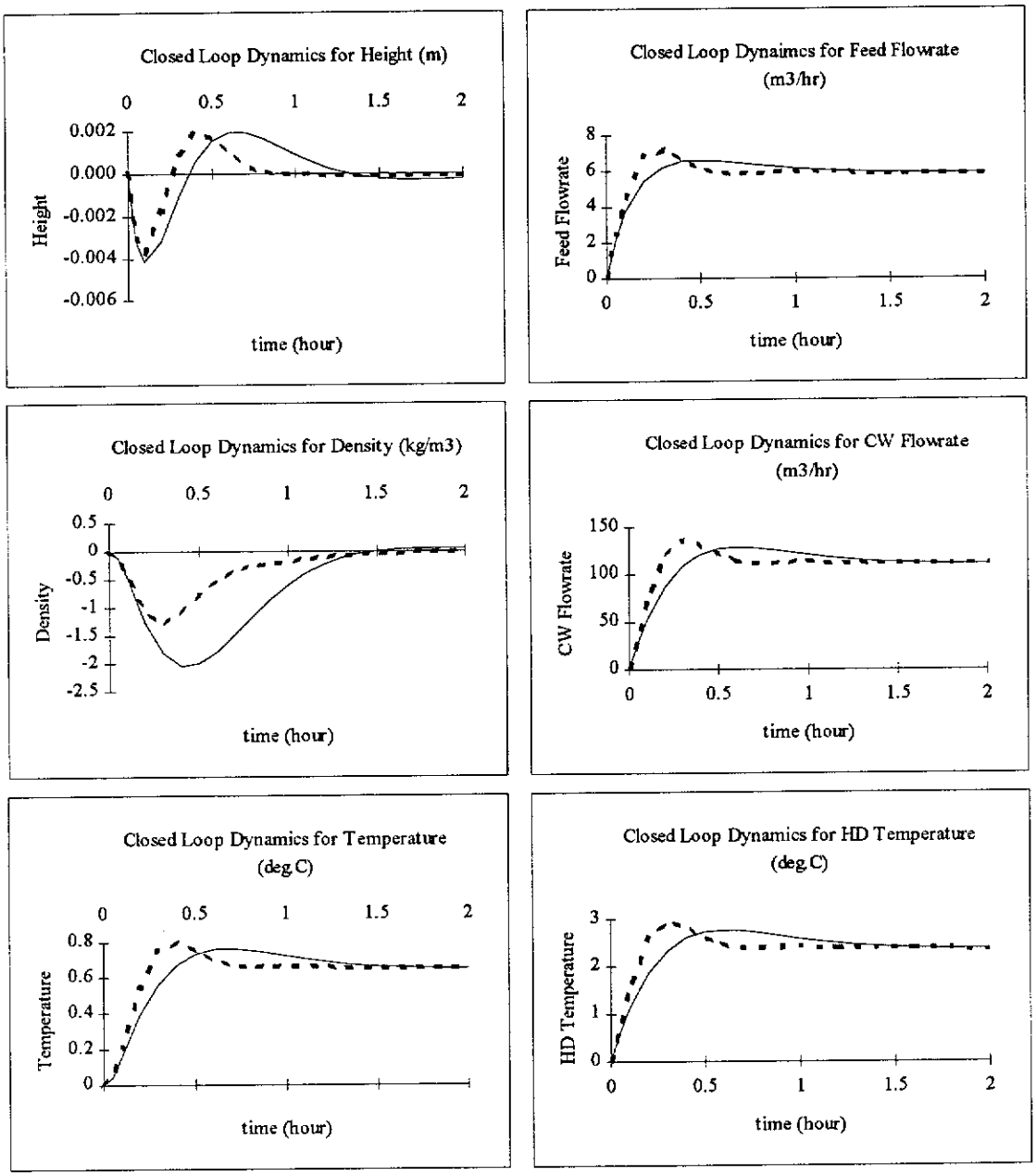


Figure 5.12: Closed Loop Dynamics for Case 2 using SHM with δ_p = (0, 1, 0)

(– – with UVA——without UVA)

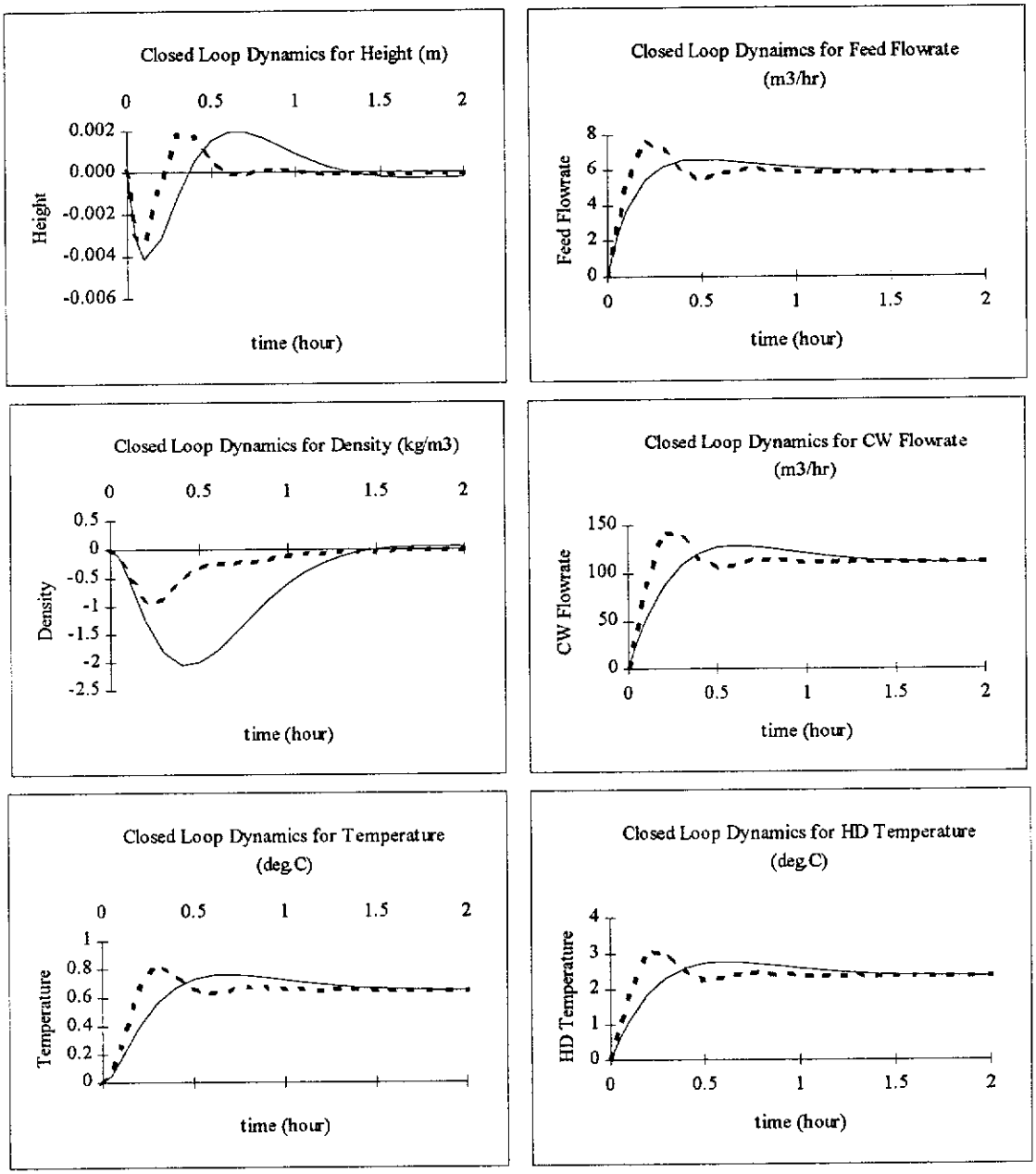


Figure 5.13: Closed Loop Dynamics for Case 2 using SHM with δ_p = (0, 2, 0)

(– – with UVA——— without UVA)

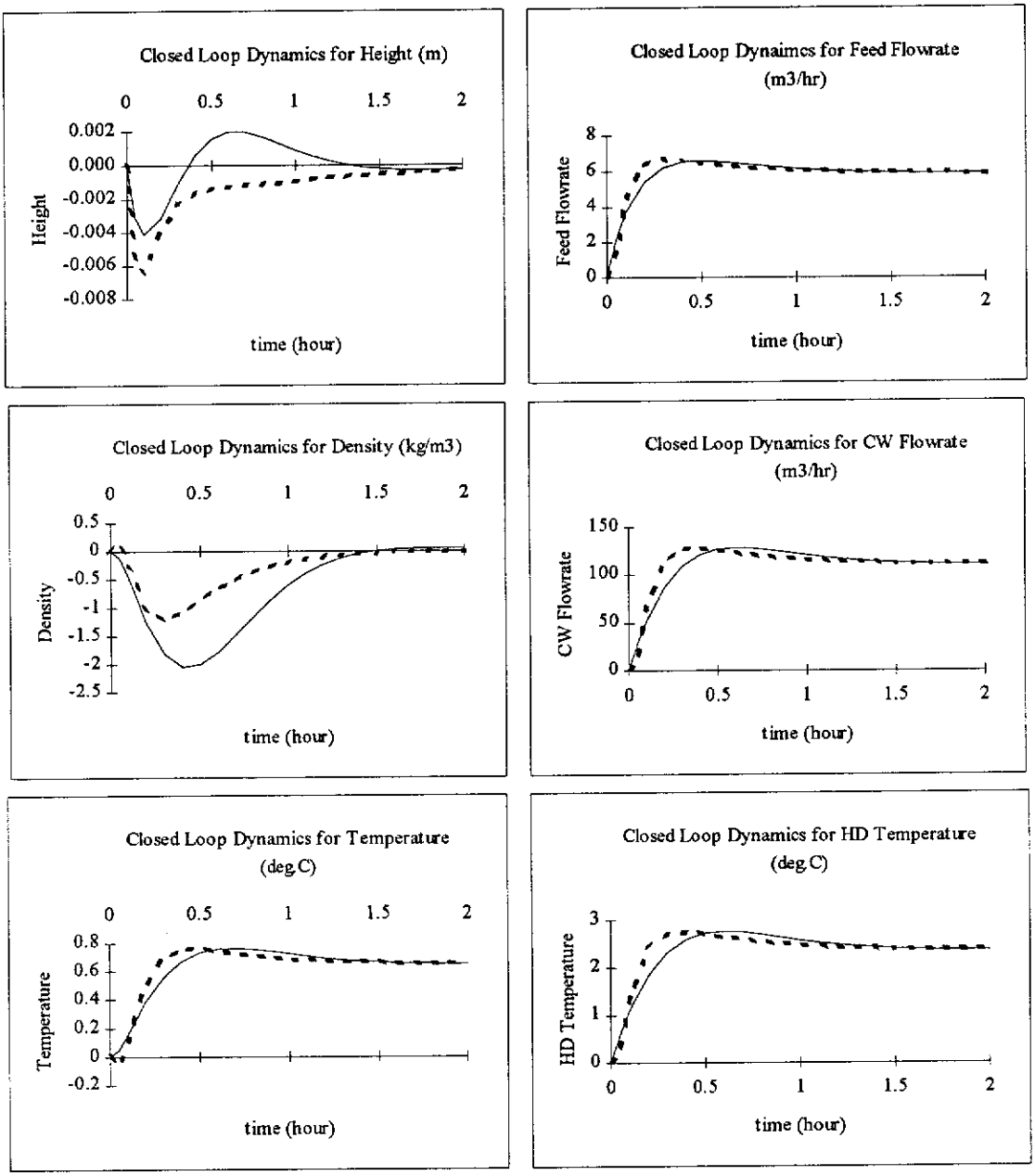


Figure 5.14: Closed Loop Dynamics for Case 2 using SHM with δ_p = (1, 1, 0)
(— —with UVA——without UVA)

5.7 Discussion On Robust Controller Synthesis Using Uncertainty Vector Adjustment

The UVA has been applied to two nonlinear control theories, I/O and SHM transformation, using the evaporative stage of the liquor burning process. The effectiveness of the adjustment is demonstrated by the significant improvements observed in control performances in terms of ITAE indices and overshoots in response curves. Before discussing the results obtained in the above section, a few comments are made on the two problems associated with the original UVA discussed earlier. Simulations have been initially performed using the original UVA. No improvement in performance was observed. After the solution to problem one was introduced, a slight improvement in ITAE was obtained. An even better performance was achieved when the solution to both problems were implemented. Therefore, this confirmed the necessity of the proposed solutions to ensure reasonable controller performance in the presence of uncertainties in the process model.

The following sections provide discussion on the simulation results obtained in section 5.6. The overall effects on the process caused by the modelling errors were different for the two nonlinear control schemes. The differences are explained below and discussions is also provided on the selection of parameters for each strategy. It should be noted that the third parameter δ_{p3} was kept at zero in all cases because the control objectives are the density and the height. The control of the third output, the temperature of the liquor in the flash tank, was deemed to be of no primary importance by plant personnel.

5.7.1 Case 1: An Increase in Temperature Factor for Input Output Linearization and Su-Hunt-Meyer Transformation

The error introduced in Case 1 was an increase of 9°C in the temperature factor. The following discussions equally apply to both nonlinear strategies studied here. Based on the ITAE indices in the nominal situation, the simulation results showed that the error improved the control performance for the level control loop, but degraded the

density control performance. Therefore, no adjustment was necessary for the height control while a full adjustment was required for the density control. Hence the suitable uncertainty vector parameters were (0, 1, 0). The appropriateness of the parameters was checked with two other sets of parameters which were (1, 1, 0) and (0, 2, 0).

Firstly, the performance obtained with parameters (0, 1, 0) was compared with that obtained with (1, 1, 0). The objective was to show that any attempt to compensate an error which has a positive effect on the closed loop dynamics will lead to a poorer control performance. In the case of I/O, the ITAE obtained with (1, 1, 0) was slightly better than that without UVA, that is, (0, 0, 0). However, a significant overshoot was observed in the first half hour which was unsatisfactory (*Figure 5.5*). When the parameters (0, 1, 0) were used, an excellent control performance was obtained in all aspects (*Figure 5.3*). The degradation in control performances of the liquor level caused by using (1, 1, 0) is even more serious for the SHM transformation and improvements were obvious when (0, 1, 0) was applied. The response curves are shown in *Figures 5.11* and *5.9*, respectively.

Once the uncertainty vector parameter for the level control was determined to be zero, an investigation was carried out to determine the effect of the increase of the other parameter. Thus, the uncertainty vector parameter for the density control was increased from 1 to 2. Improvements in control performance were observed for both methods, as shown in *Tables 5.3* and *5.7*. Therefore, with the modelling error of an increase of 9°C in temperature factor, the density control was improved as its associated uncertainty vector parameter was increased. However, the dynamics of the manipulated inputs appeared to become more vigorous and less stable as the parameter increased. Since the flow controller dynamics is not considered in this context, a further increase of the parameter is not recommended.

With the two parameter sets (1, 1, 0) and (0, 2, 0) simulated, it was demonstrated that the parameters (0, 1, 0) were most suitable for the situation here. It should be noted

that instability in performance was observed when negative values of parameters were used during the simulation to determine the best set of parameters.

5.7.2 Case 2: An Increase in Temperature Factor and a Decrease in Feed Density

Two errors were introduced into the dynamic model, that is, an increase in the temperature factor and a decrease in the feed density. Unlike case 1, the overall effects on the process caused by these errors were different for the two nonlinear control schemes. The following sections explain the differences and provide discussions on the selections of parameters for each scheme.

5.7.2.1 Input Output Linearization

Compared with the nominal situation, on the one hand, the overall effects of these errors deteriorated the density control; on the other hand, the nominal ITAE index for the height is almost the same as the ITAE when no UVA was used. Therefore, the suitable uncertainty vector parameters could be either (0, 1, 0) or (1, 1, 0). The simulation results in *Figure 5.8* show that the parameters (1, 1, 0) improved control performances for both density and height control. The ITAE index in *Table 5.4* for the height response is about three quarter of the value obtained with parameters (0, 1, 0). However, the parameters (0, 1, 0) provide a slightly better density control and no overshoot in the response curve of the height (*Figure 5.6*). The simulation also shows that a better density control will automatically lead to a better height control, which is due to interactions between the two variables. In addition, it was observed that both height and density control were further improved when the parameters were increased from (0, 1, 0) to (0, 2, 0) (*Figure 5.7*). Once again it appears that (0, 2, 0) are the most suitable parameters, based on the ITAE indices. However, with the same argument regarding the stability of the input dynamics as for case 1, the parameters (0, 1, 0) were considered to be the most suitable for this situation and hence a further increase of the parameters is not recommended.

5.7.2.2 Su-Hunt-Meyer Transformation

In the case of the SHM transformation (*Figures 5.12* to *5.14)*, the ITAE indices showed that the introduction of the two errors lead to an improvement in height control but a deterioration in density control. Therefore, parameters (0, 1, 0) were suitable, using the previous arguments. The closed loop response for this case is shown in *Figure 5.12*. When the parameters (1, 1, 0) were used, the ITAE index obtained for the level control was larger than that for no adjustment case (*see Table 5.4*). This showed that any attempt to compensate a positive effect error leads to a degradation in control performance. With the parameters increased to (0, 2, 0), the control performance for density was further improved as expected. However, using the same argument regarding the stability of inputs and the controller dynamics as above, the parameters (0, 1, 0) were considered to be more appropriate than the parameters (0, 2, 0) in this study.

5.8 Conclusions On Robust Controller Synthesis Using Uncertainty Vector Adjustment

The uncertainty vector approach was proposed to compensate for the deterioration in control performance caused by modelling errors. As a result, the two nonlinear control techniques, I/O and SHM transformation, became more robust and hence less sensitive to modelling errors. The target process studied was the evaporative stage of the liquor burning process. The development of an algorithm for the new approach was detailed. Concluding from the simulation results, this approach should only be applied when the overall effects of modelling errors degrade the performance. Any attempt to compensate any modelling error that has a positive effect will lead to degradation in performance. It is also important to note that the effects of modelling errors are very specific for the control schemes implemented on the process. Therefore, the overall effects of the modelling error should be investigated before any action is taken. However, if the situation is not allowed as in most industrial processes, a trial and error procedure will be necessary to determine suitable values of the parameters. The recommended starting values are *+1* since, in most cases, the performance will be degraded by the modelling errors. The magnitude of the

uncertainty vector parameters is also an important aspect. The performance can usually be further improved by increasing the parameters which are limited by the stability of the inputs and the controller dynamics.

CHAPTER SIX

LYAPUNOV FUNCTION ANALYSIS OF THE UNCERTAINTY VECTOR ADJUSTMENT

6.1 INTRODUCTION

The uncertainty vector approach has been developed and implemented successfully with the I/O and SHM transformation in the simulation of the evaporative stage of the liquor burning process. The basic definition, preliminary study and practical aspects of the approach have been established in the last chapter. In this chapter, the investigation will continue on the theoretical aspects of the UVA. The analytical tool used here is the direct method of Lyapunov stability theory. The stability properties of the closed loop dynamics of the evaporative system obtained using the two nonlinear control theories are examined. Using Lyapunov function, the theoretical bound of the uncertainty vector and the modelling errors are determined for each nonlinear control strategy to guarantee the asymptotic stability of the closed loop system.

6.2 BACKGROUND

Stability theory draws conclusions about the behaviour of a system without actually computing its solution trajectories. The first stability theory for the analysis of arbitrary differential equations was developed by a Russian mathematician, Alexandr Mikhailovich Lyapunov. In 1892, Lyapunov presented two methods for stability analysis, which were the linearization method and direct method, in his PhD thesis entitled *The General Problem of Motion Stability* (Lyapunov (English translated version), 1992). The linearization method draws conclusions about the local stability of a nonlinear system in the close vicinity of its steady state from the stability

properties of its linear approximation. The direct method determines the stability properties of a nonlinear system by constructing a scalar energy-like function (a Lyapunov function) for the system and examining the time variation of the function. Lyapunov not only introduced the basic definitions of stability, but he also proved many of the fundamental theorems. However, his work was only brought to the attention of the control engineering community in the Western World by the publication of a book by La Salle and Lefschetz in 1961. Today Lyapunov's linearization method has established the theoretical justification of linear control, while the direct method is an indispensable tool in the analysis and synthesis of nonlinear systems.

Apart from being a method of stability analysis, an important application of Lyapunov function is to design a robust and stable nonlinear controller with the existence of a stable Lyapunov function as the design criteria. Many researchers have also used Lyapunov's direct method to estimate the performance of a closed loop system and study its robustness. This chapter will focus on the latter application.

6.3 BRIEF LITERATURE REVIEW

6.3.1 Construction of a Lyapunov Function

One hindrance of using the Lyapunov's direct method in stability analysis of a specific nonlinear system is the construction of a Lyapunov function for that system. Methods for linear system have been well developed. However, the methods developed for autonomous nonlinear systems are usually iterative and mathematically involved, for example, Zubov (1964), Hahn (1967) and Loparo and Blankenship (1978). Michel et al (1982) developed a computer software to generate Lyapunov functions. An iterative procedure by Vannelli and Vidyasagar, 1985 yielded a maximal Lyapunov function that is a rational function rather than polynomials as proposed in aforementioned methods. Several authors have constructed Lyapunov function candidates by studying and analysing nonlinear feedback systems, for instance, Brokett (1966), Walker and McClamroch (1967), Weissenberger (1968) and Willems

(1971).

6.3.2 Robust Controller Design by Lyapunov Functions

Robust nonlinear control for systems with uncertainty such as modelling errors and unknown disturbances has received extensive attention in the past few years. Many studies have been performed in the area of robust controller synthesis, for example, Gutman (1979); Corless and Leitmann (1981); Chen and Leitmann (1987); Chen (1988); and Qu and Dorsey (1992), but only a number of control schemes was developed for the differential geometry based nonlinear control theory to operate against specific types of uncertainties. Of those approaches mentioned, the Lyapunov function method is of central importance.

For SISO systems, Kravaris and Palanki (1988) have proposed a robust controller design for input output linearizable system with uncertainties satisfying some matching conditions. In addition, Arkun and Calvet (1992) developed a stabilisation method which loosen the matching conditions, and Slotine and Hedrick (1993) and Pleau and McLellan (1994) provided systematic designs which required no matching conditions for the I/O. For feedback linearization in input-state sense, three recent papers have been written on robust stabilisation which are Khorasani (1989) and Kim and Groves (1993) for autonomous systems, and Marino and Tomei (1993) for non-autonomous systems. All these methods are conservative; require restrictive conditions on the uncertainties; and are mathematically complicated limiting their applications.

The literature on the robust control of MIMO feedback linearizable systems is very limited. Zhou and Lee (1992) synthesised a robust generic model controller for MIMO cases with no matching conditions required. However, it can only be applied to first order systems and requires the estimation of the time derivatives of the state variables. Therefore, the application of the algorithm is limited.

6.4 MATHEMATICAL PRELIMINARY

In this section, the intuitive notions of asymptotic stability in local and global sense for autonomous systems are introduced. The concepts are formally defined and their practical meanings are explained. After addressing the background issue of stability, the basic philosophy of Lyapunov's direct method is explained and some relevant theorems are presented. All the definitions and theorems presented in the following sections are referenced to Slotine and Li (1991) and Vidyasagar (1993).

6.4.1 Concepts of Stability

Lyapunov's theory abounds in a variety of notions of stability. In this study, the focus is only on asymptotic stability in terms of local and global behaviour.

6.4.1.1 Local Asymptotic Stability

A system is said to be asymptotically stable in a local sense if there exists an equilibrium state, such as the origin, and its system trajectory satisfies *Definition 6.1*. *Local* means that the initial state has to be within a *certain* distance from the equilibrium state.

Definition 6.1:

> The equilibrium state $\mathbf{x}(t) = 0$ is said to be *asymptotically stable* if, for any $R > 0$, there exists $0 < r < R$, such that if $\|\mathbf{x}(0)\| < r$, then $\|\mathbf{x}(t)\| < R$ for all $t \geq 0$ and $\mathbf{x}(t) \to 0$ as $t \to \infty$.

Essentially, local asymptotic stability means that the system trajectory can be kept arbitrarily close, and eventually converges, to the origin by starting sufficiently close to it.

6.4.1.2 Global Asymptotic Stability

The following definition characterises the global behaviour of an asymptotically stable system.

Definition 6.2:

If asymptotic stability holds for *any* initial state, the equilibrium point is said to be *globally asymptotically stable*.

The concept of *global stability* allows the initial state to be at *any* distance away from the equilibrium. For linear systems, the asymptotic stability is always global. However, a nonlinear system which is locally asymptotically stable does not imply that the system is globally stable.

6.4.2 Lyapunov's Direct Method

The basic philosophy of Lyapunov's direct method relates to the total energy of a system. If the system dynamics are such that the total energy of the system is monotonically decreasing with time, then the system, whether linear or nonlinear, must eventually settle down to its equilibrium state. Thus, the stability of a system can be concluded by simply examining the time variation of a single scalar energy function.

6.4.2.1 Lyapunov Function

The objective of the Lyapunov's direct method is to construct the scalar energy function for the nonlinear system. This energy function is called Lyapunov function. Before relating the Lyapunov function to the stability theory, the definition of a positive definite function has to be introduced.

Definition 6.3:

A scalar continuous function $V(\mathbf{x})$ is said to be *globally positive definite* if $V(\mathbf{0}) = 0$ and $\mathbf{x} \neq \mathbf{0} \Rightarrow V(\mathbf{x}) > 0$ holds over the *entire* state space. If the properties only hold when $\|\mathbf{x}(t)\| < R$, then $V(\mathbf{x})$ is a *locally* positive definite function.

The above definition implies that V has a unique minimum at the origin, in a certain ball of radius R if it is locally positive definite, or in the entire state space if it is

globally positive definite.

Lyapunov Function for Locally Asymptotic Stability

Let $\mathbf{B}_R$ denote the sphere or ball defined by $\|\mathbf{x}(t)\| < R$ in state space.

Theorem 6.1:

Assume that there exists a scalar function $V(\mathbf{x})$ with continuous first partial derivatives such that $V(\mathbf{x})$ is *locally positive definite* in $\mathbf{B}_R$, and its time derivative $\dot{V}(\mathbf{x})$ along any state trajectory of the nonlinear system is *locally negative definite* in $\mathbf{B}_R$. Then the equilibrium point at the origin is *locally asymptotically stable* in $\mathbf{B}_R$.

Lyapunov Function for Globally Asymptotic Stability

To assert global asymptotic stability of a system, the ball $\mathbf{B}_R$ in the local theorem has to be expanded to the *entire* state space. In addition, the scalar function $V(\mathbf{x})$ must be *radially unbounded*, that is, $V(\mathbf{x}) \to \infty$ as $\|\mathbf{x}\| \to \infty$.

Theorem 6.2:

Assume that there exists a scalar function $V(\mathbf{x})$ with continuous first order partial derivatives such that $V(\mathbf{x})$ is *globally positive definite*, $\dot{V}(\mathbf{x})$ is *globally negative definite* and $V(\mathbf{x}) \to \infty$ as $\|\mathbf{x}\| \to \infty$. Then the equilibrium point at the origin is *globally asymptotically stable*.

The proof for the above theorems can be found in Slotine and Li (1991).

Remark

For a given nonlinear system, many Lyapunov function candidates may exist and one Lyapunov function may yield more conclusive results than the others. If any of the conditions is not met for a particular Lyapunov function candidate, the stability property of the system will be inconclusive and a different Lyapunov function

candidate should be tested.

6.5 CONSTRUCTION OF LYAPUNOV FUNCTIONS THROUGH TRANSFORMED LINEAR SYSTEM

The Lyapunov function analysis requires one to search for a function which satisfies some pre-specified properties as described above. The main problem of the analysis is the construction of such a function for a specific system. As mentioned in section 6.3.1 of this chapter, there is no simple and unifying way of finding stable Lyapunov functions for nonlinear systems. This is the fundamental drawback of the direct method. However, systematic techniques for constructing Lyapunov functions for linear system are well-established and can be found in literature such as Kailath (1980), Slotine and Li (1991) and Vidyasagar (1993). This chapter proposes a general method for constructing valid Lyapunov functions for a state feedback linearizable nonlinear system making use of the existing techniques from the linear control theory. Using the nonlinear control theory, the nonlinear system is mapped onto another coordinate system or space in which the system behaves linearly (the transformed linear system) under feedback conditions. Vidyasagar (1993, p397) states that the integral curve of a transformed vector field in a new coordinate system is the same as the integral curve of the vector field in its original coordinate system. In other words, the following theorem can be stated. A proof of this theorem is also given for each nonlinear control scheme.

Theorem 6.3

The original nonlinear system is asymptotically stable if and only if the transformed linear system is asymptotically stable provided the following assumptions are satisified.

(1) $\| \mathbf{h}(\mathbf{x}) \| \to \infty$ as $\| \mathbf{x} \| \to \infty$;

(2) $\mathbf{x} = 0$ is the only solution for $\mathbf{h}(\mathbf{x}) = 0$ for I/O or $\mathbf{x} = 0$ is the only solution for $\mathbf{T}(\mathbf{x}) = 0$ for SHM.

Corollary 6.1

A strict Lyapunov function exists for the original nonlinear system if and only if a strict Lyapunov function exist for the transformed linear system. The two Lyapunov functions are related by the transformation relationships employed if assumptions (1) and (2) are satisfied.

Theorem 6.3 and *Corollary 6.1* can be simply proved by the fact that the transformed linear system is a subspace of the original nonlinear space. Therefore, suppose that the original nonlinear system is asymptotically stable, the transformed linear system must be asymptotically stable. This is true for any control system which satisfies assumptions (1) and (2).

The second assumption is a very strong condition which many systems are unlikely to meet. However, this is necessary for $V(\mathbf{h}(\mathbf{x}))$ to a Lyapunove function in $\mathbf{x}$.

The converse of *Theorem 6.3* and *Corollary 6.1* can be proved by showing that the Lyapunov function for a nonlinear system is actually a composite function of the Lyapunov function of the transformed system and the transformation relationships by assuming that the transformed linear system is asymptotically stable, or a Lyapunov function exists for the transformed linear system. The nonlinear control theories considered here are the I/O and the SHM transformation.

6.5.1 Proof of Theorem 6.3 for Input Output Linearization

Consider the decoupled transformed input output linear system in state space form obtained from the transformation of a nonlinear system using the I/O:

$$\dot{\mathbf{y}} = \mathbf{A}\mathbf{y} + \mathbf{B}\nu \quad \text{..... (6.1)}$$

where

$$\mathbf{A} = \begin{bmatrix} \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ \dfrac{-\beta_{10}}{\beta_{1 r_1}} & \cdots & \cdots & \cdots & \dfrac{-\beta_{1(r_1-1)}}{\beta_{1 r_1}} \end{matrix} \Big\}\, r_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ \dfrac{-\beta_{m0}}{\beta_{m r_m}} & \cdots & \cdots & \cdots & \dfrac{-\beta_{m(r_m-1)}}{\beta_{m r_m}} \end{matrix} \Big\}\, r_m \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} \begin{matrix} 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & & \ddots & 0 \\ \dfrac{1}{\beta_{1(r_1-1)}} & 0 & \cdots & 0 \end{matrix} \Big\}\, r_1 \\ \vdots \\ \begin{matrix} 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & & \ddots & 0 \\ 0 & \cdots & 0 & \dfrac{1}{\beta_{m(r_m-1)}} \end{matrix} \Big\}\, r_m \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_1^{(r_1-1)} \\ \hdashline \vdots \\ \hdashline y_m \\ \vdots \\ y_m^{(r_m-1)} \end{bmatrix} \text{ and } \boldsymbol{\nu} = \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_m \end{bmatrix}.$$

Suppose a linear feedback control is applied to the transformed linear system in (6.1) to asymptotically stabilise the system about the origin. For brevity, a proportional controller with a gain matrix K_C is used for the analysis here.

With $\nu = K_C\, \mathbf{y}$ substituted, (6.1) becomes

$$\begin{aligned} \dot{\mathbf{y}} &= \mathbf{A}\,\mathbf{y} + \mathbf{B}\left(K_C\,\mathbf{y}\right) \\ &= \left(\mathbf{A} + \mathbf{B}K_C\right)\mathbf{y} \\ &= \mathbf{C}\mathbf{y} \end{aligned} \qquad \text{.....(6.2)}$$

The eigenvalues of $\mathbf{C}$ in (6.2) must have negative real parts since the closed loop dynamics of the transformed linear system (6.2) is asymptotically stable about the origin.

The Lyapunov function can be constructed for (6.2) with techniques from linear control theory.

Choose a negative definite matrix **Q** such as **-I** for the following equation,

$$\mathbf{C}^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathbf{C} = \mathbf{Q} \quad \text{..... (6.3)}$$

so that a unique positive definite solution **P** is obtained.

A strict Lyapunov function for the transformed closed loop linear system (6.2) is

$$V(\mathbf{y}) = \mathbf{y}^{\mathrm{T}}\mathbf{P}\mathbf{y} \quad \text{..... (6.4)}$$

The time derivative of the above Lyapunov function is

$$\dot{V}(\mathbf{y}) = \sum_{j=1}^{m}\left(\frac{\partial V}{\partial y_j}\dot{y}_j\right) \quad \text{..... (6.5)}$$

which is negative definite as V is a strict Lyapunov function. And,

$$V(\mathbf{y}) \to \infty \text{ as } \|\mathbf{y}\| \to \infty \quad \text{..... (6.6)}$$

Since $\mathbf{y} = \mathbf{h}(\mathbf{x})$, therefore the Lyapunov function in (6.4) is also a function of **x**.

$$V(\mathbf{y}) = V(\mathbf{h}(\mathbf{x})) = V(\mathbf{x}) \quad \text{..... (6.7)}$$

provided that $\mathbf{x} = 0$ is the only solution for $\mathbf{h}(\mathbf{x}) = 0$.

When $\|\mathbf{x}\| \to \infty$, $\|\mathbf{y}\| \to \infty$, therefore

$$V(\mathbf{x}) \to \infty \text{ as } \|\mathbf{x}\| \to \infty \quad \text{..... (6.8)}$$

Substitute $\mathbf{y} = \mathbf{h}(\mathbf{x})$ into (6.5) to obtain

$$\begin{aligned}\dot{V}(\mathbf{y}) &= \sum_{j=1}^{m}\left(\frac{\partial V}{\partial y_j}\dot{y}_j\right)\\ &= \sum_{i=1}^{m}\sum_{j=1}^{m}\left(\frac{\partial V}{\partial y_j}\frac{\partial y_j}{\partial x_i}\right)\dot{x}_i \qquad \text{..... (6.9)}\\ &= \sum_{i=1}^{m}\left(\frac{\partial V}{\partial x_i}\dot{x}_i\right)\\ &= \dot{V}(\mathbf{x})\end{aligned}$$

Since $\dot{V}(\mathbf{y})$ is negative definite, $\dot{V}(\mathbf{x})$ must be negative definite. The implication of (6.7), (6.8) and (6.9) is that the strict Lyapunov function for the origin in the original nonlinear x-space is a composite function of the Lyapunov function in the transformed linear y-space and the transformation relationships which, in this case, transform the states to the outputs, that is, $\mathbf{y} = \mathbf{h}(\mathbf{x})$. Hence, the proof is completed for I/O.

6.5.2 Proof of Theorem 6.3 for Su-Hunt-Meyer Transformation

The proof for the SHM transformation is similar to that for the I/O. Similar proof can also be found in Su et al (1983). Suppose a linear feedback control is applied to the transformed linear system to stabilise the system asymptotically about the origin.

$$\dot{\xi} = \mathbf{C}\xi \qquad \text{..... (6.10)}$$

where $\mathbf{C} = \mathbf{A} + \mathbf{B}K_C$; $\mu = K_C\,\xi$ and the matrices **A** and **B** are shown in (2.12) of Chapter 2.

Since the transformed linear system is asymptotically stable by the gain matrix K_C, the eigenvalues of matrix **C** must have negative real parts. The positive definite matrix **P** is computed as explained above. Therefore, the strict Lyapunov function for the closed loop transformed linear system (6.10) is

$$V(\xi) = \xi^{\mathrm{T}}\mathbf{P}\xi \qquad \text{..... (6.11)}$$

The time derivative of the above Lyapunov function is

$$\dot{V}(\xi) = \sum_{j=1}^{m}\left(\frac{\partial V}{\partial \xi_j}\dot{\xi}_j\right) \quad \text{..... (6.12)}$$

which is negative definite as V is a strict Lyapunov function. Moreover,

$$V(\xi) \to \infty \text{ as } \|\xi\| \to \infty$$

By definition, $\xi_j = T_j$ for $j = 1, ..., n$ and T_j is a function of $\mathbf{x}$. Therefore,

$$V(\xi) = V(\mathbf{T}(\mathbf{x})) = V(\mathbf{x}) \quad \text{..... (6.13)}$$

provided that $\mathbf{x} = 0$ is the only solution for $\mathbf{T}(\mathbf{x}) = 0$.

Since $\|\mathbf{x}\| \to \infty$, $\|\xi\| \to \infty$ and hence, $\|\mathbf{T}\| \to \infty$, therefore $V(\mathbf{x}) \to \infty$. And,

$$\begin{aligned}
\dot{V}(\xi) &= \sum_{j=1}^{n}\left(\frac{\partial V}{\partial \xi_j}\dot{\xi}_j\right) \\
&= \sum_{j=1}^{n}\left(\frac{\partial V}{\partial T_j}\dot{T}_j\right) \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}\left(\frac{\partial V}{\partial T_j}\frac{\partial T_j}{\partial x_i}\right)\dot{x}_i \quad \text{..... (6.14)} \\
&= \sum_{i=1}^{n}\left(\frac{\partial V}{\partial x_i}\dot{x}_i\right) \\
&= \dot{V}(\mathbf{x})
\end{aligned}$$

Therefore, the Lyapunov function for the nonlinear system is a composite function of the Lyapunov function for the transformed linear system in ξ-space and the transformation relationships, $T_j(\mathbf{x})$ for $j = 1, ..., n$. Thus the proof is completed for SHM transformation.

6.6 THEORETICAL BOUNDS FOR UNCERTAINTY VECTOR AND MODELLING ERROR

Modelling errors are always present in the mathematical representation of a process. In the last chapter, an uncertainty vector was proposed to describe the overall effect of modelling errors. Apart from using the above Lyapunov analysis to predict the stability of a nonlinear system, the analysis can also be used to establish the theoretical bound of the uncertainty vector and the modelling error within which the system remains asymptotically stable at the origin. This section will substantiate the discussions and findings in the last chapter and provide a complete analysis on the proposed UVA method.

6.6.1 Theoretical Bounds for the Uncertainty Vector Using Input Output Linearization

Assume that the Lyapunov function for the nominal nonlinear system has been determined using Corollary 6.1,

$$V(\mathbf{y}) = \mathbf{y}^{\mathrm{T}}\mathbf{P}\mathbf{y} \text{ and } \mathbf{y} = \mathbf{h}(\mathbf{x})$$

Here we assume that there is no modelling error or uncertainty vector in $\mathbf{y} = \mathbf{h}(\mathbf{x})$. Therefore, V of the nominal system is also the Lyapunov function of the adjusted nonlinear system. The time derivative of V with $\mathbf{u}_{\text{adjusted}}$ is

$$\begin{aligned}\dot{V}_{\text{adjusted}} &= \left[\frac{\partial V}{\partial x_1} \quad \frac{\partial V}{\partial x_2} \quad \cdots \quad \frac{\partial V}{\partial x_n}\right]\dot{\mathbf{x}} \\ &= \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\mathbf{f} + \mathbf{g}\mathbf{u}_{\text{adjusted}}\right)\end{aligned} \quad \text{..... (6.15)}$$

Therefore, using (5.13), the above equation becomes

$$\dot{V}_{\text{adjusted}} = \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\mathbf{f} + \mathbf{g}\left(-\mathbf{A}_{\text{adjusted}}\right)^{-1}\left(\mathbf{B} + \mathbf{C} - \nu\right)\right) \quad \text{..... (6.16)}$$

where

$$\mathbf{A}_{\text{adjusted}} = \begin{bmatrix} \hat{\beta}_{1 r_1} L_{g_1} L_{f+\delta}^{r_1-1} h_1 & \cdots & \hat{\beta}_{1 r_1} L_{g_m} L_{f+\delta}^{r_1-1} h_1 \\ \vdots & & \vdots \\ \hat{\beta}_{m r_m} L_{g_1} L_{f+\delta}^{r_m-1} h_m & \cdots & \hat{\beta}_{m r_m} L_{g_m} L_{f+\delta}^{r_m-1} h_m \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} \sum_{k=0}^{r_1} \hat{\beta}_{1k} L_f^k h_1 \\ \vdots \\ \sum_{k=0}^{r_m} \hat{\beta}_{mk} L_f^k h_m \end{bmatrix} \text{ and } \mathbf{C} = \begin{bmatrix} \sum_{k=1}^{r_1} \hat{\beta}_{1k} L_\delta^k h_1 \\ \vdots \\ \sum_{k=1}^{r_m} \hat{\beta}_{mk} L_\delta^k h_m \end{bmatrix}$$

For the system to be asymptotically stable, $\dot{V}$ has to be negative definite. That is,

$$\left[\frac{\partial V}{\partial x}\right]\left(\mathbf{f} + \mathbf{g}\left(-\mathbf{A}_{\text{adjusted}}\right)^{-1}\left(\mathbf{B} + \mathbf{C} - \nu\right)\right) < 0 \quad \text{..... (6.17)}$$

(6.17) defines the boundary of the uncertainty vector *implicitly* for systems with well-defined relative order for each output. When the relative order of each output is greater than 1, the isolation of δ in (6.17), if possible, involves heavy algebra which leads to a mathematically complicated expression. However, when the relative order of each output is equal to 1, the bound of the uncertainty vector can be defined *explicitly*. That is,

$$\|\delta\| < \frac{|\dot{V}|}{\left\| \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g} \mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \right\|} \quad \text{..... (6.18)}$$

where $\|\,.\,\|$ indicates the Euclidean norm.

It should be noted that $\dot{V}$ in (6.18) is the time derivative of the Lyapunov function of the nominal system. That is,

$$\dot{V} = \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\mathbf{f} + \mathbf{g}\mathbf{u}\right)$$

The proof of (6.18) is given below. When $r_1 = r_2 = \cdots = r_m = 1$,

$$\mathbf{A}_{\text{adjusted}} = \mathbf{A} = \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \begin{bmatrix} L_{\mathbf{g}_1} h_1 & \cdots & L_{\mathbf{g}_m} h_1 \\ \vdots & & \vdots \\ L_{\mathbf{g}_1} h_m & \cdots & L_{\mathbf{g}_m} h_m \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \hat{\beta}_{10} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m0} \end{bmatrix} \mathbf{h} + \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \begin{bmatrix} L_{\mathbf{f}} h_1 \\ \vdots \\ L_{\mathbf{f}} h_m \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \begin{bmatrix} L_{\delta} h_1 \\ \vdots \\ L_{\delta} h_m \end{bmatrix} = \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathbf{h}}{\partial \mathbf{x}} \end{bmatrix} \delta$$

Substituting **A**, **B** and **C** into (6.17) gives

$$\begin{bmatrix} \dfrac{\partial V}{\partial \mathbf{x}} \end{bmatrix} \left(\mathbf{f} + \mathbf{g}(-\mathbf{A})^{-1} (\mathbf{B} + \mathbf{C} - \nu) \right)$$
$$+ \begin{bmatrix} \dfrac{\partial V}{\partial \mathbf{x}} \end{bmatrix} \mathbf{g}(-\mathbf{A}^{-1}) \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathbf{h}}{\partial \mathbf{x}} \end{bmatrix} \delta < 0$$

That is,

$$\begin{bmatrix} \dfrac{\partial V}{\partial \mathbf{x}} \end{bmatrix} (\dot{\mathbf{x}}) + \begin{bmatrix} \dfrac{\partial V}{\partial \mathbf{x}} \end{bmatrix} \mathbf{g}(-\mathbf{A}^{-1}) \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathbf{h}}{\partial \mathbf{x}} \end{bmatrix} \delta < 0$$

where the relation of $\dot{\mathbf{x}} = \mathbf{f} + \mathbf{g}(-\mathbf{A})^{-1} (\mathbf{B} - \nu)$ has been used. This implies that

$$\dot{V} - \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g}\mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \delta < 0$$

Therefore,

$$\dot{V} < \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g}\mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \delta \quad \text{..... (6.19)}$$

One of the sufficient conditions for (6.19) to be valid is when

$$|\dot{V}| > \left| \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g}\mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \delta \right|$$

where $|\,.\,|$ indicate the absolute value of a scalar quantity.

The issue here is that the above inequality is the sufficient condition for (6.19), therefore, the stability of the adjusted system. Hence, it is conservative.

Furthermore, one of the sufficient condition for (6.19) to be true is when

$$|\dot{V}| > \left\| \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g}\mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \right\| \|\delta\|$$

Therefore, further conservative condition is established. The final conservative upper bound of δ to guarantee the stability of the closed loop system is

$$\|\delta\| < \frac{|\dot{V}|}{\left\| \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g}\mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \right\|} \text{ and (6.18) is proved.}$$

6.6.2 Theoretical Bounds for the Uncertainty Vector Using Su-Hunt-Meyer Transformation

Consider the nonlinear system with $n = m$, the expression for the adjusted manipulated inputs is shown below:

$$\mathbf{u}_{\text{adjusted}} = \mathbf{G}^{-1}\left(\begin{bmatrix} T_{n+1} \\ \vdots \\ T_{n+m} \end{bmatrix} - \begin{bmatrix} L_{\mathbf{f}}\, T_1 \\ \vdots \\ L_{\mathbf{f}}\, T_n \end{bmatrix} - \begin{bmatrix} L_{\delta}\, T_1 \\ \vdots \\ L_{\delta}\, T_n \end{bmatrix}\right) \quad \text{..... (6.20)}$$

where

$$\mathbf{G} = \begin{bmatrix} L_{\mathbf{g}_1} T_1 & \cdots & L_{\mathbf{g}_m} T_1 \\ \vdots & & \vdots \\ L_{\mathbf{g}_1} T_n & \cdots & L_{\mathbf{g}_m} T_n \end{bmatrix}$$

It is assumed that the Lyapunov function for the nominal nonlinear system has been determined using *Corollary 6.1*. The Lyapunov function candidate in (6.11) depends on T_1 ,..., T_n which are not functions of the uncertainty vector when $n = m$. Therefore, the Lyapunov function for the nominal system is the same as that for the adjusted system if the same gain matrix is used for both cases. It is important to note that the Lyapunov function for the adjusted system will be different from the nominal one for system with $n > m$ since the adjusted transformed states $T_{1,\text{adjsuted}}$,..., $T_{n,\text{adjusted}}$ are functions of not only the original states, but also of the uncertainty vector.

With the Lyapunov function constructed, the bound of the uncertainty vector for the system with $n = m$ is,

$$\|\delta\| < \frac{\left|\left[\dfrac{\partial V}{\partial \mathbf{x}}\right]\dot{\mathbf{x}}\right|}{\left\|\left[\dfrac{\partial V}{\partial \mathbf{x}}\right]\right\|} = \frac{|\dot{V}|}{\left\|\left[\dfrac{\partial V}{\partial \mathbf{x}}\right]\right\|} \quad \text{..... (6.21)}$$

As indicated in (6.19), $\dot{V}$ is the time derivative of the Lyapunov function of the nominal system.

Proof:

With $\mathbf{u}_{\text{adjusted}}$ in (6.20) substituted into(6.14),

$$\dot{V}_{\text{adjusted}} = \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\mathbf{f} + \mathbf{g}\,\mathbf{G}^{-1}\left(\begin{bmatrix} T_{n+1} \\ \vdots \\ T_{n+m} \end{bmatrix} - \begin{bmatrix} L_{\mathbf{f}}\,T_1 \\ \vdots \\ L_{\mathbf{f}}\,T_n \end{bmatrix} - \begin{bmatrix} L_{\delta}\,T_1 \\ \vdots \\ L_{\delta}\,T_n \end{bmatrix}\right)\right) \quad \text{..... (6.22)}$$

Now, it can be seen that

$$\mathbf{G} = \begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n} \end{bmatrix}\mathbf{g} \text{ and } \begin{bmatrix} L_{\delta}\,T_1 \\ \vdots \\ L_{\delta}\,T_n \end{bmatrix} = \begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n} \end{bmatrix}\delta$$

Substituting the above equations into (6.22) gives

$$\dot{V}_{\text{adjusted}} = \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\begin{array}{l} \mathbf{f} + \mathbf{g}\,\mathbf{G}^{-1}\left(\begin{bmatrix} T_{n+1} \\ \vdots \\ T_{n+m} \end{bmatrix} - \begin{bmatrix} L_{\mathbf{f}}\,T_1 \\ \vdots \\ L_{\mathbf{f}}\,T_n \end{bmatrix}\right) \\ -\left(\mathbf{g}\left(\begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n} \end{bmatrix}\mathbf{g}\right)^{-1}\begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n} \end{bmatrix}\delta\right) \end{array}\right)$$

Since $n = m$, $\mathbf{g}$ and $\begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n} \end{bmatrix}$ are square matrices. Furthermore, the existence of transformation guarantees that the two matrices are invertible. Hence,

$$\dot{V}_{\text{adjusted}} = \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\begin{array}{l}\mathbf{f}+\mathbf{g}\,\mathbf{G}^{-1}\left(\begin{bmatrix}T_{n+1}\\ \vdots \\ T_{n+m}\end{bmatrix}-\begin{bmatrix}L_{\mathbf{f}}\,T_1\\ \vdots \\ L_{\mathbf{f}}\,T_n\end{bmatrix}\right) \\ -\left(\mathbf{g}\,\mathbf{g}^{-1}\begin{bmatrix}\frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n}\\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n}\end{bmatrix}^{-1}\begin{bmatrix}\frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n}\\ \vdots & & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n}\end{bmatrix}\delta\right)\end{array}\right)$$

That is,

$$\begin{aligned}\dot{V}_{\text{adjusted}} &= \left[\frac{\partial V}{\partial \mathbf{x}}\right]\left(\mathbf{f}+\mathbf{g}\,\mathbf{G}^{-1}\left(\begin{bmatrix}T_{n+1}\\ \vdots \\ T_{n+m}\end{bmatrix}-\begin{bmatrix}L_{\mathbf{f}}\,T_1\\ \vdots \\ L_{\mathbf{f}}\,T_n\end{bmatrix}\right)-\delta\right)\\ &= \left[\frac{\partial V}{\partial \mathbf{x}}\right](\dot{\mathbf{x}}-\delta)\end{aligned}$$

where the relation of $\dot{\mathbf{x}} = \mathbf{f} + \mathbf{g}\,\mathbf{G}^{-1}\left(\begin{bmatrix}T_{n+1}\\ \vdots \\ T_{n+m}\end{bmatrix}-\begin{bmatrix}L_{\mathbf{f}}\,T_1\\ \vdots \\ L_{\mathbf{f}}\,T_n\end{bmatrix}\right)$ has been used.

$\dot{V}_{\text{adjusted}}$ must be negative definite to guarantee the system to be asymptotically stable. Therefore,

$$\left[\frac{\partial V}{\partial \mathbf{x}}\right](\dot{\mathbf{x}}-\delta) < 0 \qquad \text{(6.23)}$$

With the same conditions and derviations of (6.18), a conservative upper bound for δ can be found and (6.21) is proved.

6.6.3 Existence of the Bound for the Uncertainty Vector

Using the Lyapunov function of the nominal system, the bound of the uncertainty vector for both I/O and SHM transformation are defined. The following informal proofs are performed regarding the existence of the bound defined in (6.18) and (6.21).

The norm used here is the Euclidean norm in order to guarantee the continuous property of the bound. The existence of the bound in (6.18) and (6.21) can be proved by the fact that the system is asymptotically stable; hence, the states are continuous and bounded. Therefore, the bounds must exist for the norms in the two equations. That is, the norm of the uncertainty vector in both cases must be less than some values, $\mathbf{L}_{I/O}$ for the closed loop system using I/O, and $\mathbf{L}_{SHM}$ for systems employing SHM transformation, respectively.

6.6.4 Theoretical Bound for Modelling Errors

The definition of the uncertainty vector in (5.10) of Chapter 5 implies that the Euclidean norm of the uncertainty vector is an estimate of the norm of the modelling errors. Therefore, the upper bound of the uncertainty vector is an esitmate of the upper boundary of the modelling errors. That is, in order to guarantee the asymptotic stability of the closed loop system,

$$\left\| \begin{matrix}\text{Overall Effect of} \\ \text{Modelling Error}\end{matrix} \right\| < \frac{|\dot{V}|}{\left\| \left[\frac{\partial V}{\partial \mathbf{x}}\right] \mathbf{g}\mathbf{A}^{-1} \begin{bmatrix} \hat{\beta}_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\beta}_{m1} \end{bmatrix} \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right] \right\|} \quad \text{..... (6.24)}$$

for I/O with $r_1 = r_2 = \cdots = r_m = 1$, and

$$\left\| \begin{matrix}\text{Overall Effect of} \\ \text{Modelling Error}\end{matrix} \right\| < \frac{|\dot{V}|}{\left\| \left[\frac{\partial V}{\partial \mathbf{x}}\right] \right\|} \quad \text{..... (6.25)}$$

for SHM transformation with $n = m$.

6.6.5 Theoretical Bound for Uncertainty Vector Parameters

The theoretical bound for the uncertainty vector parameters can be determined from (6.18) and (6.21) for I/O and SHM transformation, respectively. By the definition in (5.10) of Chapter 5,

$$\begin{bmatrix}\delta_1\\ \delta_2\\ \vdots\\ \delta_n\end{bmatrix} \cong \begin{bmatrix}\delta_{p1}(-\dot{x}_1)\\ \delta_{p2}(-\dot{x}_2)\\ \vdots\\ \delta_{pn}(-\dot{x}_n)\end{bmatrix} \equiv \begin{bmatrix}\delta_{p1} & 0 & \cdots & 0\\ 0 & \delta_{p2} & \ddots & \vdots\\ \vdots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & \delta_{pn}\end{bmatrix}\begin{bmatrix}-\dot{x}_1\\ -\dot{x}_2\\ \vdots\\ -\dot{x}_n\end{bmatrix} \equiv \delta_p(-\dot{\mathbf{x}})$$

Taking norm on both sides,

$$\therefore \quad \|\delta\| \cong \|\delta_p(-\dot{\mathbf{x}})\| \qquad (6.26)$$

As a result of the norm in (6.18), (6.21) and (6.26), it is too complicated to explicitly determine the bounds for the uncertainty vector parameters. Therefore, instead of establishing a complicated mathematical relationships, (6.18) and (6.21) will be used as conditions for the selection of parameters.

6.6.6 Remarks

Even though (6.18) is only valid for the uncertainty vector in I/O with $r_i = 1$, the above discussions of (6.18) are applicable to system with $r_i > 1$ since (6.18) is derived from (6.17) which is a general expression for any system. The same argument is valid for (6.21) describing the uncertainty vector in SHM transformation. In other words, despite the difference in the algebra involved, the implications of (6.21) are also true for system with $n > m$.

6.7 SIMULATION RESULTS AND DISCUSSIONS

6.7.1 Construction of Lyapunov Functions

The evaporative unit of the liquor burning process associated with the Bayer process is used here as a continuation of the previous investigation. Two simulation programs have been written with MAPLE V.3 to generate the Lyapunov function for the evaporative unit when the I/O and the SHM transformation were used as the control strategy. The output of the two simulation programs are detailed in To et al (1996b). The Lyapunov functions constructed with the simulation program are:

$$V = 0.4167\,dx_1^{\,2} + 0.4167\,dx_2^{\,2} \quad \text{..... (6.30)}$$

for the I/O, and

$$V = 0.1051\times10^{7} + 267.2672\,dx_1^{\,2} + 0.1422\times10^{8}\ln\left(0.762+dx_2\right)^2 + 0.7733\times10^{7}\ln\left(0.762+dx_2\right) + 0.0001019\,dx_3^{\,2} \quad \text{..... (6.31)}$$

for the SHM transformation. The gain matrices used are:

$$K_C = \begin{bmatrix} -5 & 0 \\ 0 & -5 \end{bmatrix} \text{ for I/O, and } K_C = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -0.06 & 0 \\ 0 & 0 & -20 \end{bmatrix} \text{ for SHM.}$$

It can be easily seen that the Lyapunov function in (6.30) is positive definite for any value of dx_1 and dx_2. However, (6.31) is less obvious because of the third and the forth term involving the term dx_2. Since V in (6.30) is positive definite for any value of dx_1 and dx_3, a plot of V with $dx_1 = dx_3 = 0$ is included in *Figure 6.1* to show that V is also positive definite within the operating range of dx_2. Therefore, V in (6.31) is positive definite within the entire operating range or locally and hence is a Lyapunov function candidate for the system.

The time derivative of the two Lyapunov function candidates in (6.30) and (6.31) are too complicated to be shown here. They can be found in To et al. (1996d). However, the plots of $\dot{V}$ at $dx_2 = 0$ for both nonlinear control theories are provided in *Figures 6.2* and *6.3* to show that $\dot{V}$ in both cases are negative definite. It is noted that the level surfaces of $\dot{V}$ at other values of dx_2 have the same shape but at different levels and none of the surfaces is found above the plane $\dot{V} = 0$.

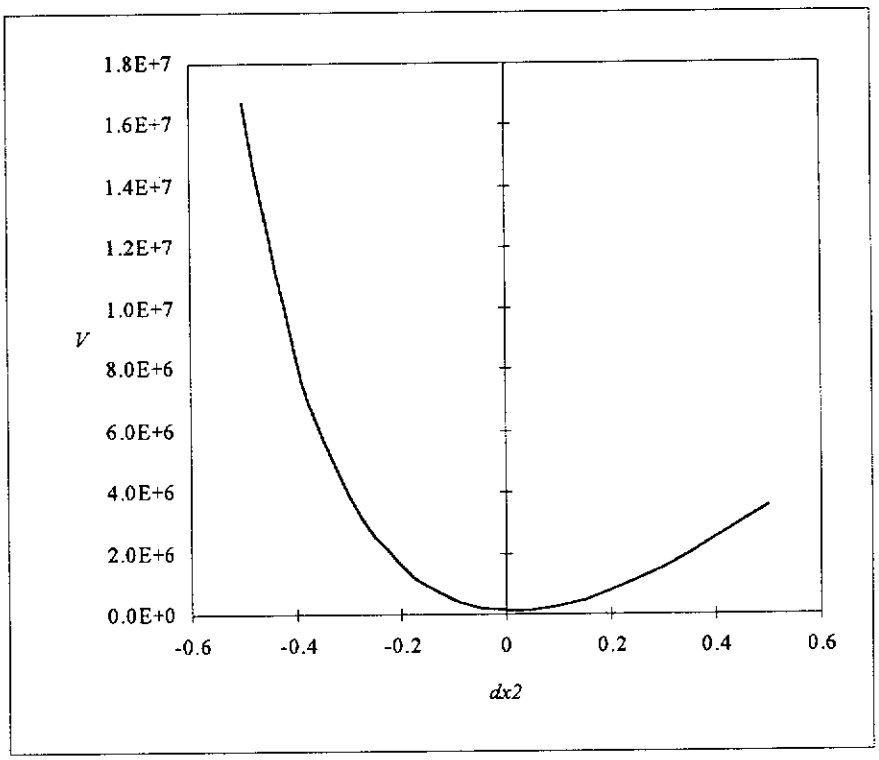


Figure 6.1: V versus dx_2 with $dx_1 = dx_3 = 0$ for SHM

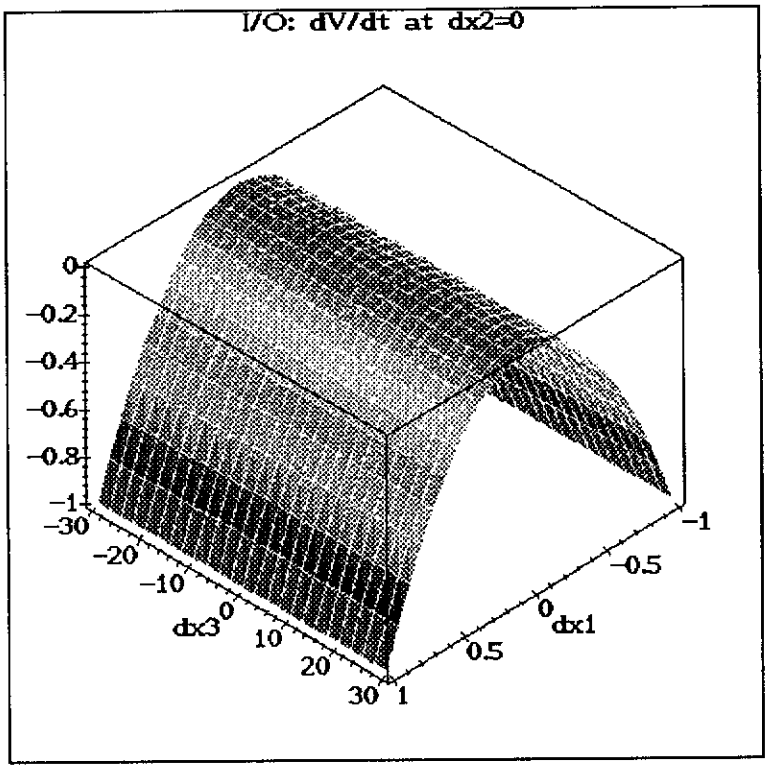


Figure 6.2: Level Curve of $\dot{V}$ at $dx_2 = 0$ for I/O

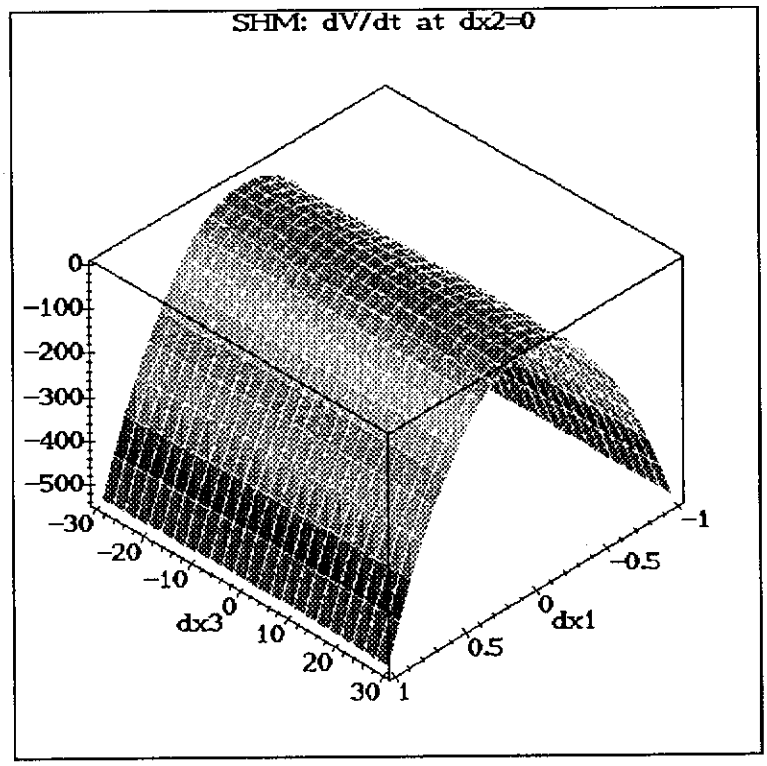


Figure 6.3: Level Curve of $\dot{V}$ at $dx_2 = 0$ for SHM

In short, a global Lyapunov function (6.30) is constructed for the closed loop dynamics using the I/O while the Lyapunov function (6.31) determined for the SHM transformation is only valid in a local sense because the transformation for dx_2 is only valid when $dx_2 > -0.762$. However, it should be noted that the steady state value of the density of the liquor in the flash tank x_2 is nominated to be 1.762 tonne/m^3. Therefore, it would only be under very abnormal situations that the density would fall below 1 tonne/m^3 so that the locality of the Lyapunov function for the SHM transformation is reasonable.

6.7.2 Determination of the Bound for the Uncertainty Vector

With the Lyapunov functions constructed, the bound of the uncertainty vector or the modelling error can be established using (6.18) and (6.21) for I/O and SHM transformation, respectively. The calculation of the norms encountered no difficulty, despite the nonlinearities and complexities in $\dot{V}$. However, in order to determine $\mathbf{L}_{I/O}$ and $\mathbf{L}_{SHM}$, it was necessary to locate the minimum or the lower bound of the denominators in (6.18) and (6.21). As a result of the nonlinearities, especially the square roots in the Euclidean norms, the determinations of these bounds become very difficult for MAPLE. However, since the establishment of the existence of $\mathbf{L}_{I/O}$ and $\mathbf{L}_{SHM}$ is more important than the actual determinations of $\mathbf{L}_{I/O}$ and $\mathbf{L}_{SHM}$ and also because these values have virtually no practical consequence, it was decided that the bounds of the uncertainty vector or the modelling errors should be expressed as a function of $\mathbf{x}$.

6.8 CONCLUSIONS

The theoretical bounds of the uncertainty vector and the modelling errors have been established for both I/O and SHM transformation using Lyapunov function analysis. The analysis showed that an upper bound exists for the uncertainty vector in order to guarantee the stability of the closed loop system, and this bound is also the upper bound of the modelling errors by the definition of the uncertainty vector.

Besides the above robustness analysis, it was proved in this chapter that the Lyapunov function of the nonlinear system is a composite function of the Lyapunov function of the transformed linear system and the transformation relationships.

With the theoretical analysis here and the practical aspects in the previous study, the study of the new UVA procedure is considered to be completed for the I/O and SHM transformation schemes. It can be concluded that the uncertainty vector is simple and effective. Most important of all, the adjustment requires no restrictive conditions as other existing techniques. The uncertainty vector exists provided the nonlinear system is feedback transformable to a linear system by the I/O and the SHM transformation under nominal situations.

CHAPTER SEVEN

SIMULATION AND CONTROL OF A TRIPLE EFFECTS EVAPORATOR

7.1 INTRODUCTION

In the previous chapters, two nonlinear control strategies were applied to the simulation of a single effect industrial evaporator. The results showed that the control performance obtained using the nonlinear control strategies, especially the I/O, were better than the traditional SISO linear controller. Robustness studies of the nonlinear control strategies led to the development of UVA; thereby, more appropriate control action, computed based on the effect of modelling errors, can be applied to the system affected by modelling errors. The effectiveness of the new procedure was corroborated by the improvements observed in the simulation of the single effect evaporator. In this chapter, a more complicated problem is proposed to challenge the nonlinear control theories and the new UVA procedure. The target process selected is a triple effects evaporator designed on the basis of a new five effects evaporation unit associated with a new liquor burning process in Alcoa's Wagerup alumina refinery. The five effects evaporator could not be used due to proprietary reasons.

The model of the triple effects evaporation process was developed using mass and energy balances and then the simulation was implemented using MAPLE V.3. The control performances were investigated using both linear and nonlinear control schemes. The UVA procedure was applied to the nonlinear control system to demonstrate its effectiveness against modelling errors.

The literature on the control of multiple effect evaporators in the chemical engineering application is limited. Shinskey (1988) discussed the linear control of a double effect

evaporator and Burdett and Holland (1971) modelled the dynamics of a multiple effect evaporator. Other studies related to multiple effect evaporators are in the area of food engineering, for example, Tonelli et al (1990), Quaak and Gerritsen (1990), Runyon et al (1991) and Driscoll et al (1995). Therefore, the contribution of this chapter is significant in providing insights into the nonlinear control of a multiple effects evaporator in chemical engineering applications.

7.2 TRIPLE EFFECTS EVAPORATOR DESIGN

7.2.1 Process Description

The basic function and structure of the triple effects evaporator are the same as the single effect evaporation unit described in Chapter 3. The two designs treat approximately the same amount of feed liquor. The difference is that the steam requirements of the triple effects design is significantly less than the single effect evaporator since the triple effects design is more energy efficient. The simplified process flow diagram is shown in *Figure 7.1*. *Tables 7.1 to 7.4* tabulate the steady state values of each stream. Some of the values have employed more than three significant figures in order to ensure that the time derivative of each state variable in the dynamic model equations presented in the following section is kept as close to zero as numerically possible at the specified steady state conditions. The product density and the inventory in each flash tank are still the prime control objectives. The triple effects evaporator employs the countercurrent feed design where the feed enters the effect operating at the lowest pressure and is pumped from one effect to the other in the direction opposite to that in which the vapour is flowing. Thus the product leaves the evaporator in the effect that functions at the highest pressure. The main characteristics of the triple effects evaporator design in this context which are different from the typical countercurrent feed design are that the second and third effect operate at the same pressure and the vapour exiting from these two effects provides the energy requirement for the first effect. Thus the heat source for both second and third effects is the fresh steam and no steam is required for the first stage under normal operating circumstances. The unusual arrangement here is able to

provide additional degrees of freedom in control. The reason will be clear in the later section which describes the proposed control scheme for the operation.

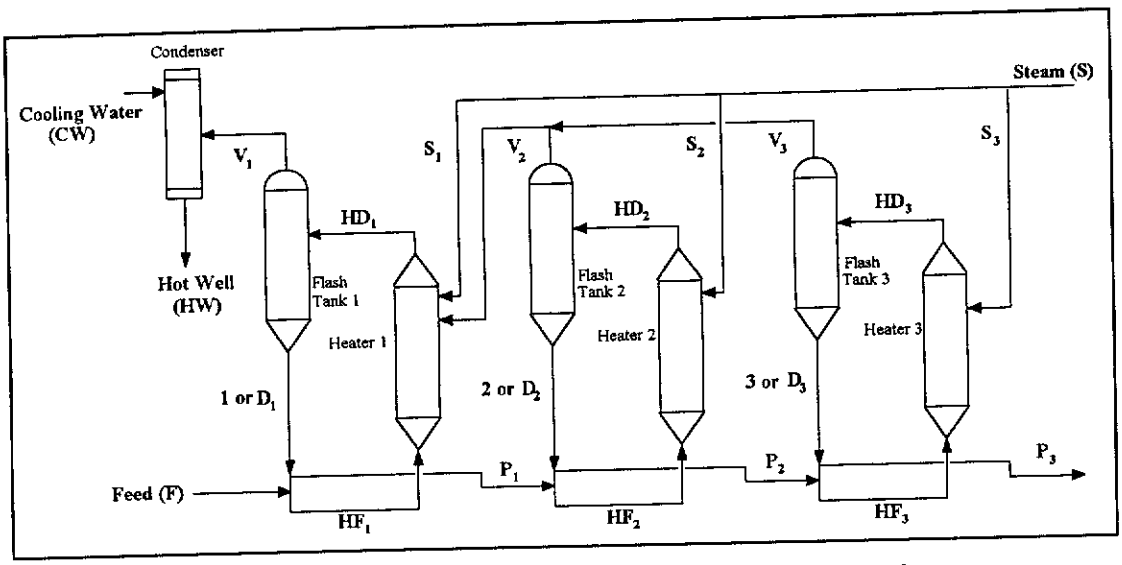


Figure 7.1: Flowsheet of Triple effects Evaporator Design

7.2.2 Dynamic Modelling

The dynamic equations used to model the triple effects evaporator design were derived from mass and energy balances around the flowsheet in *Figure 7.1*. The model is for a six-input six-output system representing a mathematically complex and highly nonlinear unit operation. The strict validity of the equations was not established because the unit is yet to be constructed. Nevertheless, the model is considered to give an adequate description of the operation for the purpose of the evaluation of control strategies, based on the experience gained from the single effect evaporator operation. Moreover, the simulation will definitely provide some insight and useful information for the future operation.

Table 7.1: Steady State Values of Process Variables for Evaporator Stage 1

Stream		F	D_1	V_1
Description		Feed to Effect 1	Underflow from Flash Tank 1	Flash Vapour ex Flash Tank 1
Avg. Flow at Temp.	m^3/hr	29.7	2290	29499.92
Mass Flow at Temp.	tonne/hr	41	3298	≅ 7.059981
Temperature	°C	66	90.6	74.6
SG at Temperature	kg/m^3	1380	1440	0.24
Heat Capacity	kJ/kg	3.29	3.25	
Boiling Point Elevation	°C		16	
Pressure (Gauge)	kPa			38.1
Latent Heat	kJ/kg			2234.1

Stream		HF_1	HD_1
Description		Feed to Heater 1	Discharge from Heater 1
Avg. Flow at Temp.	m^3/hr	≅ 2296	≅ 2296
Mass Flow at Temp.	tonne/hr	3304.9	3304.9
Temperature	°C	≅ 90.3	≅ 91.87509747
SG at Temperature	kg/m^3	≅ 1439	≅ 1439
Heat Capacity	kJ/kg	≅ 3.25	≅ 3.25

Table 7.2: Steady State Values of Process Variables for Evaporator Stage 2

Stream		P_1	D_2	V_2
Description		Feed to Effect 2	Underflow from Flash Tank 2	Flash Vapour ex Flash Tank 2
Avg. Flow at Temp.	m^3/hr	23.7	2300	7136.23
Mass Flow at Temp.	tonne/hr	34.1	3427	≅ 4.924000055
Temperature	°C	90.6	129	105
SG at Temperature	kg/m^3	1440	1490	0.69
Heat Capacity	kJ/kg	3.25	3.32	
Boiling Point Elevation	°C		24	
Pressure (Gauge)	kPa			126
Latent Heat	kJ/kg			2190.7

Stream		HF_2	HD_2
Description		Feed to Heater 2	Discharge from Heater 2
Avg. Flow at Temp.	m^3/hr	≅ 2305	≅ 2305
Mass Flow at Temp.	tonne/hr	3431.9	3431.9
Temperature	°C	≅ 128.6	≅ 129.7613460
SG at Temperature	kg/m^3	≅ 1489	≅ 1489
Heat Capacity	kJ/kg	≅ 3.32	≅ 3.32

Table 7.3: Steady State Values of Process Variables for Evaporator Stage 3

Stream		P_2	D_3	V_3
Description		Feed to Effect 3	Underflow from Flash Tank 3	Flash Vapour ex Flash Tank 3
Avg. Flow at Temp.	m^3/hr	19.6	2300	4019.3
Mass Flow at Temp.	tonne/hr	29.2	3542	≅ 2.773321582
Temperature	°C	129	135	105
SG at Temperature	kg/m^3	1490	1540	0.69
Heat Capacity	kJ/kg	3.32	3.41	
Boiling Point Elevation	°C		30	
Pressure (Gauge)	kPa			126
Latent Heat	kJ/kg			2190.7

Stream		HF_3	HD_3
Description		Feed to Heater 3	Discharge from Heater 3
Avg. Flow at Temp.	m^3/hr	≅ 2302.5	≅ 2302.5
Mass Flow at Temp.	tonne/hr	3544.9	3544.9
Temperature	°C	≅ 134.9	≅ 135.3933081
SG at Temperature	kg/m^3	≅ 1539.6	≅ 1539.6
Heat Capacity	kJ/kg	3.41	3.41

Table 7.4: Steady State Values of Product Stream and Utilities

Stream		P_3	CW	S
Description		Product	Cooling Water to Condenser	Total Live Steam
Avg. Flow at Temp.	m^3/hr	17.1	≅ 77.4681	4153.92
Mass Flow at Temp.	tonne/hr	26.3	≅ 77.4681	≅ 8.97246615
Temperature	°C	135	27	144
SG at Temperature	kg/m^3	1540	1000	2.16
Heat Capacity	kJ/kg	3.41		
Pressure (Gauge)	kPa			406
Latent Heat	kJ/kg			2110.5

Stream		S_1	S_2	S_3
Description		Live Steam to Heater 1	Live Steam to Heater 2	Live Steam to Heater 3
Avg. Flow at Temp.	m^3/hr	0	2905.5574	1248.343287
Mass Flow at Temp.	tonne/hr	0	≅ 6.276004	≅ 2.69646215
Temperature	°C	144	144	144
SG at Temperature	kg/m^3	2.16	2.16	2.16
Pressure (Gauge)	kPa	406	406	406
Latent Heat	kJ/kg	2110.5	2110.5	2110.5

7.2.2.1 Assumptions

The assumptions involved in the derivation of the dynamic equations are similar to those for the single effect evaporator listed in section 3.1.2.1 of Chapter 3 and they are summarised as follow:

1. Total liquor volume in heat exchangers and piping is equivalent to a tank of area of 23.12 m^2, and height of 2.21 m. This capacity is added to the inventory of each flash tank. It is also assumed that the tank size is the same for all effects.
2. The hot well approach temperature is assumed to be constant at 5 °C.
3. The liquor boiling point elevation is assumed to be constant.
4. The cooling water temperature is assumed to be constant.
5. The specific heat capacity of liquor is assumed to be constant.
6. Flow controller dynamics are not simulated.
7. All temperature equations are solved consistently in °C unit.
8. The process is assumed to be adiabatic.
9. The volume of the liquor remains constant in all ancillary pipe works since the pipes run full.
10. The evaporation rate is calculated on liquid water basis.
11. The setpoint is assumed to be constant at the corresponding steady state value.
12. The heat transfer efficiency is usually larger than 98% according to the plant personnel. Therefore, the efficiency can be assumed to be 100%.

7.2.2.2 Dynamic Equations

The model which is composed of nine state equations with six inputs and six outputs is summarised below. The manipulations of the equations were performed by MAPLE V.3 and the print-out in *Appendix D* shows the combined model equations.

1. Stage 1

$$\frac{dh_1}{dt} = \frac{Q_{HF_1} - \dot{m}_{V_1} - Q_{D_1}}{A_1} \quad \text{.....(7.1)}$$

$$\frac{d\rho_1}{dt} = \frac{Q_{HF_1}\rho_{HF_1} - \dot{m}_{V_1} + \left(m_{V_1} - Q_{HF_1}\right)\rho_1}{A_1 h_1} \quad \text{.....(7.2)}$$

$$\frac{dT_1}{dt} = \frac{Q_{HF_1}\rho_{HF_1} C_{P,HD_1} T_{HD_1} - \dot{m}_{V_1} H_{V_1} - C_{P,1} T_1 \left(Q_{HF_1}\rho_{HF_1} - \dot{m}_{V_1}\right)}{C_{P,1} A_1 h_1 \rho_1} \quad \text{.....(7.3)}$$

where

$$Q_{D_1} = Q_{P_1} + Q_{HF_1} - Q_F \quad \text{.....(7.4)}$$

$$\dot{m}_{V_1} = \frac{Q_{CW} C_{P,CW}\left(T_1 - \left(T_a + T_{e_1}\right) - T_{CW}\right)}{H_{V_1} - C_{P,CW}\left(T_1 - \left(T_a + T_{e_1}\right)\right)} \quad \text{.....(7.5)}$$

$$T_{HD_1} = T_{HF_1} + \frac{\dot{m}_{S_1} H_{S_1} + \dot{m}_{V_2} H_{V_2} + \dot{m}_{V_3} H_{V_3}}{Q_{HF_1}\rho_{HF_1} C_{P,HF_1}} \quad \text{.....(7.6)}$$

$$T_{HF_1} = \frac{\rho_F Q_F C_{P,F} T_F + \rho_1 Q_{D_1} C_{P,1} T_1 - \rho_1 Q_{P_1} C_{P,1} T_1}{\rho_{HF_1} Q_{HF_1} C_{P,HF_1}} \quad \text{.....(7.7)}$$

$$\rho_{HF_1} = \frac{\rho_F Q_F + \rho_1 Q_{D_1} - \rho_1 Q_{P_1}}{Q_{HF_1}} \quad \text{.....(7.8)}$$

2. Stage 2

$$\frac{dh_2}{dt} = \frac{Q_{HF_2} - \dot{m}_{V_2} - Q_{D_2}}{A_2} \quad \text{.....(7.9)}$$

$$\frac{d\rho_2}{dt} = \frac{Q_{HF_2}\rho_{HF_2} - \dot{m}_{V_2} + \left(m_{V_2} - Q_{HF_2}\right)\rho_2}{A_2 h_2} \quad \text{.....(7.10)}$$

$$\frac{dT_2}{dt} = \frac{Q_{HF_2}\,\rho_{HF_2}\,C_{P,HD_2}\,T_{HD_2} - \dot{m}_{V_2}\,H_{V_2} - C_{P,2}\,T_2\left(Q_{HF_2}\,\rho_{HF_2} - \dot{m}_{V_2}\right)}{C_{P,2}\,A_2\,h_2\,\rho_2} \quad \text{..... (7.11)}$$

where

$$Q_{D_2} = Q_{P_2} + Q_{HF_2} - Q_{P_1} \quad \text{..... (7.12)}$$

$$\dot{m}_{V_2} = \frac{Q_{P_1}\,\rho_1\,C_{P,1}\,T_1 + \dot{m}_{S_2}\,H_{S_2} - Q_{P_2}\,\rho_2\,C_{P_2}\,T_2}{H_{V_2}} \quad \text{..... (7.13)}$$

$$T_{HD_2} = T_{HF_2} + \frac{\dot{m}_{S_2}\,H_{S_2}}{Q_{HF_2}\,\rho_{HF_2}\,C_{P,HF_2}} \quad \text{..... (7.14)}$$

$$T_{HF_2} = \frac{\rho_1\,Q_{P_1}\,C_{P,1}\,T_1 + \rho_2\,Q_{D_2}\,C_{P,2}\,T_2 - \rho_2\,Q_{P_2}\,C_{P,2}\,T_2}{\rho_{HF_2}\,Q_{HF_2}\,C_{P,HF_2}} \quad \text{..... (7.15)}$$

$$\rho_{HF_2} = \frac{\rho_1\,Q_{P_1} + \rho_2\,Q_{D_2} - \rho_2\,Q_{P_2}}{Q_{HF_2}} \quad \text{..... (7.16)}$$

3. Stage 3

$$\frac{dh_3}{dt} = \frac{Q_{HF_3} - \dot{m}_{V_3} - Q_{D_3}}{A_3} \quad \text{..... (7.17)}$$

$$\frac{d\rho_3}{dt} = \frac{Q_{HF_3}\,\rho_{HF_3} - \dot{m}_{V_3} + \left(m_{V_3} - Q_{HF_3}\right)\rho_3}{A_3\,h_3} \quad \text{..... (7.18)}$$

$$\frac{dT_3}{dt} = \frac{Q_{HF_3}\,\rho_{HF_3}\,C_{P,HD_3}\,T_{HD_3} - \dot{m}_{V_3}\,H_{V_3} - C_{P,3}\,T_3\left(Q_{HF_3}\,\rho_{HF_3} - \dot{m}_{V_3}\right)}{C_{P,3}\,A_3\,h_3\,\rho_3} \quad \text{..... (7.19)}$$

where

$$Q_{D_3} = Q_{P_3} + Q_{HF_3} - Q_{P_2} \quad \text{..... (7.20)}$$

$$\dot{m}_{V_3} = \frac{Q_{P_2}\,\rho_2\,C_{P,2}\,T_2 + \dot{m}_{S_3}\,H_{S_3} - Q_{P_3}\,\rho_3\,C_{P_3}\,T_3}{H_{V_3}} \quad \text{..... (7.21)}$$

$$T_{HD_3} = T_{HF_3} + \frac{\dot{m}_{S_3}\,H_{S_3}}{Q_{HF_3}\,\rho_{HF_3}\,C_{P,HF_3}} \quad \text{..... (7.22)}$$

$$T_{HF_3} = \frac{\rho_2\,Q_{P_2}\,C_{P,2}\,T_2 + \rho_3\,Q_{D_3}\,C_{P,3}\,T_3 - \rho_3\,Q_{P_3}\,C_{P,3}\,T_3}{\rho_{HF_3}\,Q_{HF_3}\,C_{P,HF_3}} \quad \text{..... (7.23)}$$

$$\rho_{HF_3} = \frac{\rho_2\,Q_{P_2} + \rho_3\,Q_{D_3} - \rho_3\,Q_{P_3}}{Q_{HF_3}} \quad \text{..... (7.24)}$$

The input and output variables are listed in *Table 7.5*. All these variables were selected after consulting the plant personnel and based on the study of the single effect evaporator. It should be noted that all the state equations are input-linear except dT_2/dt and dT_3/dt.

Table 7.5: List of Variables for the Triple Effects Evaporator

Manipulated Inputs		Outputs	
u_1	Q_F	y_1	h_1
u_2	Q_{P_1}	y_2	h_2
u_3	Q_{P_2}	y_3	h_3
u_4	Q_{CW}	y_4	T_1
u_5	$\dot{m}_{S_2}$	y_5	$\rho_{P_2} = \rho_2$
u_6	$\dot{m}_{S_3}$	y_6	$\rho_{P_3} = \rho_3$

7.2.3 Selection of Control Strategies

The study of the single effect evaporator showed that the I/O provided the best control performance. Therefore, the control strategies used for the investigation of the control of the triple effects evaporator are the I/O with the classical linear controller using proportional and integral action as the bench mark. The SHM transformation is not applicable in this case because the state equations dT_2/dt and dT_3/dt are not input-linear and the algebra involved for either the simplified or original solution algorithm is too complicated to be implemented in MAPLE. The design method of I/O was modified in order to solve a more general class of process models because of the two input-nonlinear state equations present in the triple evaporator model. In addition, the tedious manipulations in isolating the **f** vector and the **g** matrix from the complicated state equations using MAPLE were proved to be non-trivial because of the zero-equivalent problem. Details can be found in *Appendix A*. The proposed modification is based on the results of the study of a SISO case for the I/O by Henson and Seborg (1990).

7.2.4 Proposed Input Output Linearization for a General Class of Process

The method for applying the I/O to a more general class of nonlinear process is developed in this section. In particular, input variables may appear nonlinearly and the models are not required to have a relative degree one.

Consider the following general class of autonomous models with n states, m inputs and w outputs,

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\mathbf{u}) \\ y_i &= h_i(\mathbf{x}) \qquad i = 1,\ldots,w \end{aligned} \qquad \text{.....(7.25)}$$

where **f** is a C^∞ vector field on $\mathbf{R}^n$ and $h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_w(\mathbf{x})$ are scalar fields on $\mathbf{R}^n$. In other words, the mapping **f** and the functions $h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_w(\mathbf{x})$ are smooth in their arguments, that is, all entries are real-valued functions of **x** with continuous

partial derivatives of any order. In addition, **y**, **x** and **u** represent vectors with $\mathbf{y} \in \mathbf{R}^w$, $\mathbf{x} \in \mathbf{R}^n$, and $\mathbf{u} \in \mathbf{R}^m$, respectively.

The algorithm proposed here is an extension of the I/O by Kravaris and Soroush (1990) and the discussion of the general SISO system by Henson and Seborg (1990). The I/O algorithm has already been detailed in Chapter 2. The proposed algorithm can be used to determine the static state feedback law of the I/O for more general systems, in which the inputs may appear nonlinearly. However, it is assumed that the process is minimum phase and all states can be accurately measured or estimated.

7.2.4.1 Relative Order

To obtain the static state feedback law, the relative order characterises how the inputs affect the outputs directly. The relative order r_i associated with the i^{th} output of the general system (7.25) is defined to be the smallest non-negative integer such that the following row matrix

$$\left[\frac{\partial\left(L_f^{r_i-1}h_i(\mathbf{x})\right)}{\partial u_1} \quad \frac{\partial\left(L_f^{r_i-1}h_i(\mathbf{x})\right)}{\partial u_2} \quad \cdots \quad \frac{\partial\left(L_f^{r_i-1}h_i(\mathbf{x})\right)}{\partial u_m}\right] \qquad (7.26)$$

has at least one nonzero element at the steady state $\mathbf{x}_o$. The total relative order of the system is defined to be the summation of the relative orders r_i for $1 \le i \le w$.

7.2.4.2 Design Method

Consider the minimum phase system (7.25) with equal number of inputs and outputs ($m = w$), the decoupled input output behaviour is described by the following equation.

$$\begin{aligned} \sum_{k=0}^{r_1}\hat{\beta}_{1k}\frac{d^k y_1}{dt^k} &= v_1 \\ &\vdots \\ \sum_{k=0}^{r_m}\hat{\beta}_{mk}\frac{d^k y_m}{dt^k} &= v_m \end{aligned} \qquad (7.27)$$

where $\dfrac{d^k y_i}{dt^k} = L_f^k h_i(\mathbf{x})$.

According to the implicit function theorem (Henson and Seborg, 1990), the static state feedback law to be used to determine the original nonlinear inputs **u**, that is, the solution of (7.27), exists if the following conditions are satisfied.

1. (7.27) must be a function of **u**. That is,

$$\left[\frac{\partial v_i}{\partial u_1} \quad \frac{\partial v_i}{\partial u_2} \quad \cdots \quad \frac{\partial v_i}{\partial u_m}\right] \text{ has at least one non-zero element for } 1 \le i \le m.$$

2. For every $\mathbf{x}_o$ and $\mathbf{v}_o$, there exists a unique $\mathbf{u}_o$.

The first condition is always satisfied if the system has well-defined relative orders. The second condition is to ensure that a unique inverse or solution exists for (7.27). This condition may not be satisfied when the system has multiple steady states. In this case, more than one solutions exists and the validity of each solution would depend on the operating region of the actual system and the numerical technique used to solve (7.27).

If the inverse of (7.27) or a unique state feedback law exists and the state variables are measured for the decoupled system, the MIMOGLC structure can be constructed as described in section 2.4.2.1. Using the same proportional and integral controller for each $(v_i - y_i)$ pair, the control law and the closed loop transfer function are exactly the same as (2.33) and (2.34) of Chapter 2, respectively.

7.3 Simulation Results And Discussions

7.3.1 Specifications for the Simulation

This section presents the relevant data used in the simulation for the closed loop dynamics of the triple effects evaporator. The closed loop responses were investigated with a disturbance of approximately 3 m^3/hr increase in the product flowrate of Stage 3. The study is divided into two sections: the study of the nominal situation with no modelling error and the study of robustness which demonstrates the

effectiveness of the UVA proposed in Chapter 5 by introducing modelling errors into the model.

The MAPLE simulation program for the I/O is written based on the procedures developed in *Appendix A* and is provided in *Appendix E*. The state feedback law and a sample of the output can also be found in *Appendix E*.

The relative order of each output is obviously 1 because the six output variables are in fact six of the nine state variables. The closed loop responses of the triple effects evaporator were obtained using local linearization (Linear) and I/O. The sampling time used was 0.02 hour. The PI controller tuning parameters and design parameters $\hat{\beta}_{ik}$ in I/O are listed in *Table 7.6*. The tuning parameters and $\hat{\beta}_{ik}$ were selected based on the results of Chapter 3 to achieve the most suitable responses in terms of the input dynamics. The ITAE performance indices for each control strategy were computed on the basis of 10 hours time span, since this was how long it took for the responses to completely settle down.

Table 7.6: PI Controller Tuning Parameters and Design Parameters $\hat{\beta}_{ik}$

Design Parameters, $\hat{\beta}_{ik}$				PI Controller Parameters			
$\hat{\beta}_{10}$	1	$\hat{\beta}_{11}$	0.5	K_{C1}	10	τ_{I1}	60
$\hat{\beta}_{20}$	1	$\hat{\beta}_{21}$	0.5	K_{C2}	10	τ_{I2}	60
$\hat{\beta}_{30}$	1	$\hat{\beta}_{31}$	0.5	K_{C3}	10	τ_{I3}	60
$\hat{\beta}_{40}$	1	$\hat{\beta}_{41}$	-0.5	K_{C4}	-500	τ_{I4}	60
$\hat{\beta}_{50}$	1	$\hat{\beta}_{51}$	0.5	K_{C5}	10	τ_{I5}	60
$\hat{\beta}_{60}$	1	$\hat{\beta}_{61}$	0.5	K_{C6}	10	τ_{I6}	60

The simulation of the closed loop dynamics of the triple effects evaporator in presence of a specific disturbance was successfully implemented using MAPLE V.3. However, several problems were encountered and solutions were proposed to redress each problem.

MAPLE was unable to compute the term exp(Δ(t-r)) in the analytical solution (A.2) in *Appendix A* of the linearized model equations (A.1). This could be due to memory limitations caused by the fact that Δ is a 9 by 9 matrix. Therefore, instead of using the linear analytical solution, the state trajectory was obtained using the simple Euler method. However, it was found that the temperature equations were very sensitive to a small change in the input variables using this method. Therefore, it was concluded that the system of nonlinear differential equations was ill-conditioned and suitable scaling of the temperature equations were necessary. The scaling factors used for the equations $\dot{T}_1$, $\dot{T}_2$ and $\dot{T}_3$ were 100, 10 and 10, respectively.

Despite the fact that MAPLE cannot compute the term exp(Δ(t-r)), MAPLE was able to compute the eigenvalues of the characteristic matrix Δ. The matrices Δ, Ψ and Γ are the Jacobian matrices with respect to the vector of states **x**, inputs **u** and disturbances **d**, correspondingly, and can be found in *Appendix F*. The last three rows of the matrix Δ confirmed that scaling should be applied to the equations $\dot{T}_1$, $\dot{T}_2$ and $\dot{T}_3$ since some of the elements in the corresponding rows are 10 to 10000 times larger than the rest.

The eigenvalues of the characteristic matrix were determined using MAPLE and recorded in *Appendix F*. All the eigenvalues except the one for T_1 which is virtually zero are negative or have negative real parts. This implies that all states or outputs of the process are open loop stable except T_1. Therefore, T_1 was included in the control scheme even though the product temperature of the single effect evaporator can be left under the open loop control.

The zero elements in disturbance matrix Γ imply that the specified disturbance has no effect on the states: h_1 ,h_2, ρ_1 ,ρ_2 and T_2, in terms of the open loop dynamics. However, the simulation using the Euler method showed that the disturbance affected all the states. The implication is that part of the nonlinearity of the model is lost in the process of linearization using the first order Taylor series and, unlike the single effect

evaporator model, the use of a linearized model is not sufficient to represent the triple effects evaporation process. Apart from the Euler method, other numerical methods can be used to approximate the solution trajectory of the model such as modified Euler method, Runge-Kutta method, Adams method and Adams-Moulton method. All these methods are detailed in Gerald and Wheatley (1989). However, the Euler method was selected for this study because of its simplicity in implementation. The simulation results showed that the method is sufficient for the purpose of comparing different control strategies and is able to provide better approximations than the linearization method.

7.3.2 Study of the Nominal Situation

The study of the nominal situation presents the closed loop response of the triple effects evaporator against the specific disturbance described above assuming that there is no modelling error. The ITAE indices for each output using local linearization and I/O are tabulated in *Table 7.7*. The response curves of the output, state and input variables are shown in *Figures 7.2* and *7.3* for local linearization and for I/O, respectively.

Table 7.7: ITAE for Triple Effects Evaporator (Nominal Case)

Control Strategy	Local Linearization	Input Output Linearization
h_1	0.0640	0.0000488
h_2	0.0213	0.0000379
h_3	0.0264	0.00465
T_1	0.0461	0.00702
$\rho_{P_2} = \rho_2$	0.00664	0.0000209
$\rho_{P_3} = \rho_3$	0.00268	0.000452

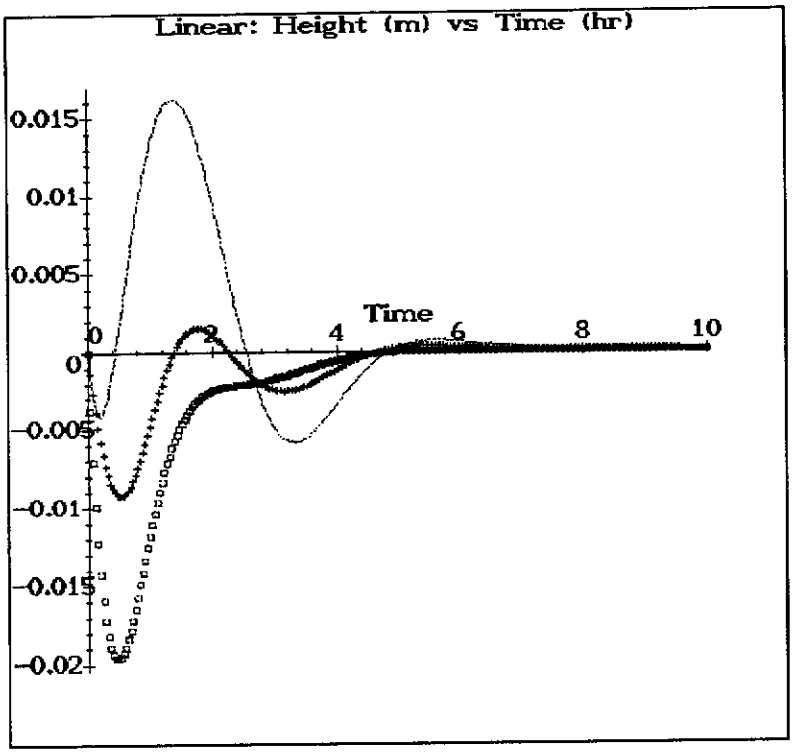


Figure 7.2 (a): — h_1 + h_2 □ h_3

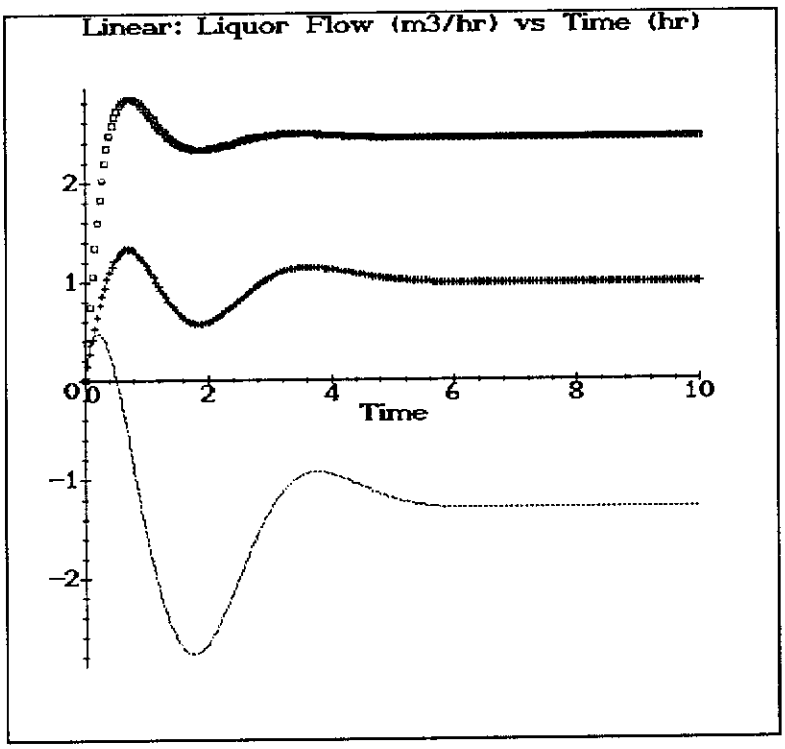


Figure 7.2 (b): — Q_F + Q_{P_1} □ Q_{P_2}

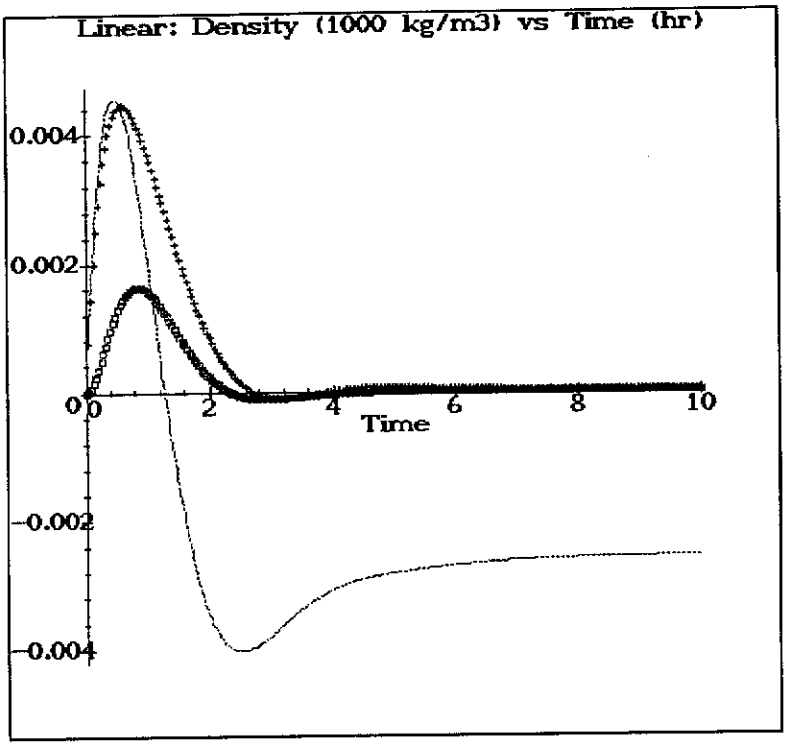


Figure 7.2 (c): — ρ_1 + ρ_2 □ ρ_3

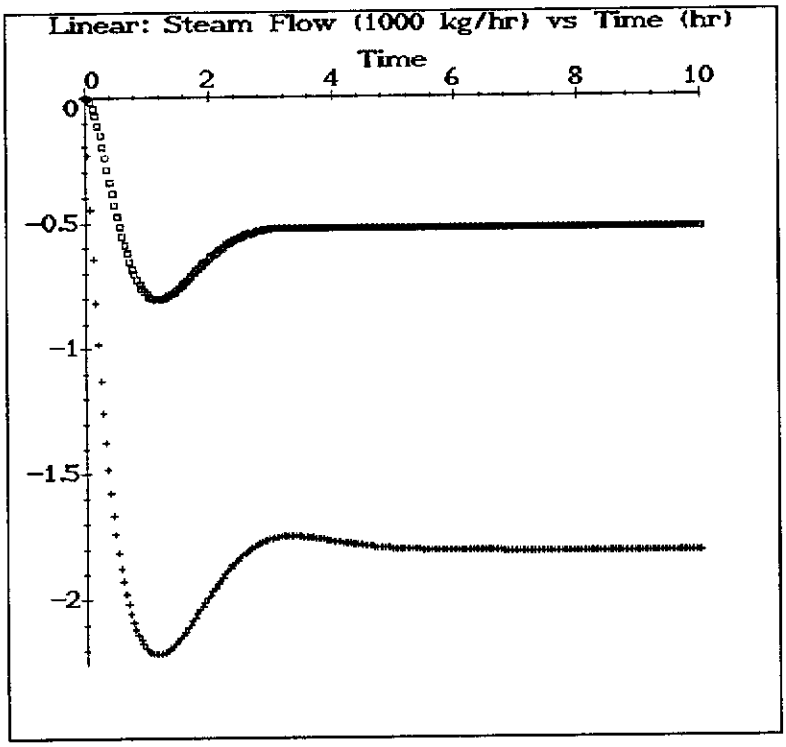


Figure 7.2 (d): + $\dot{m}_{S_2}$ □ $\dot{m}_{S_3}$

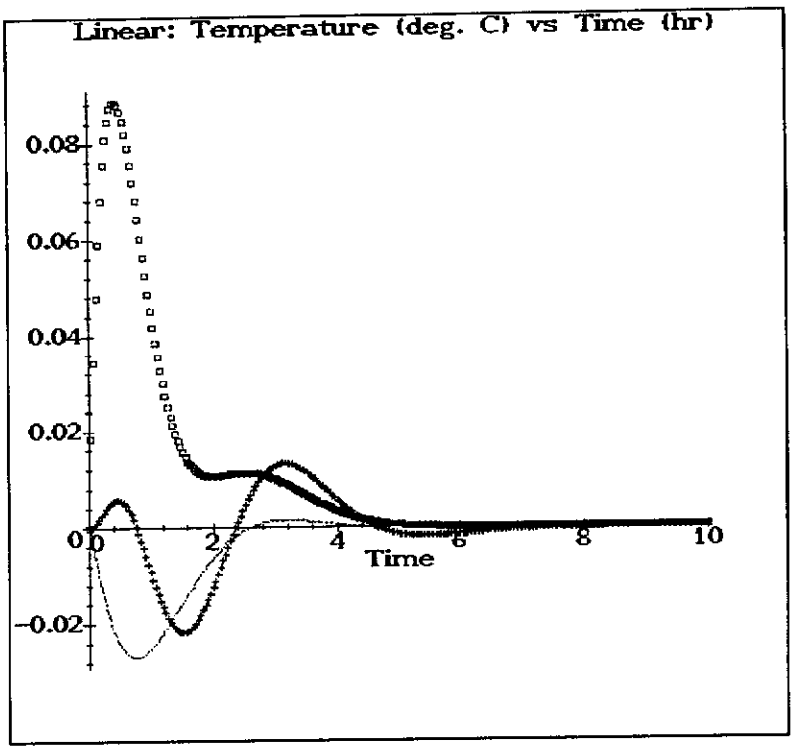


Figure 7.2 (e): — T_1 + T_2 □ T_3

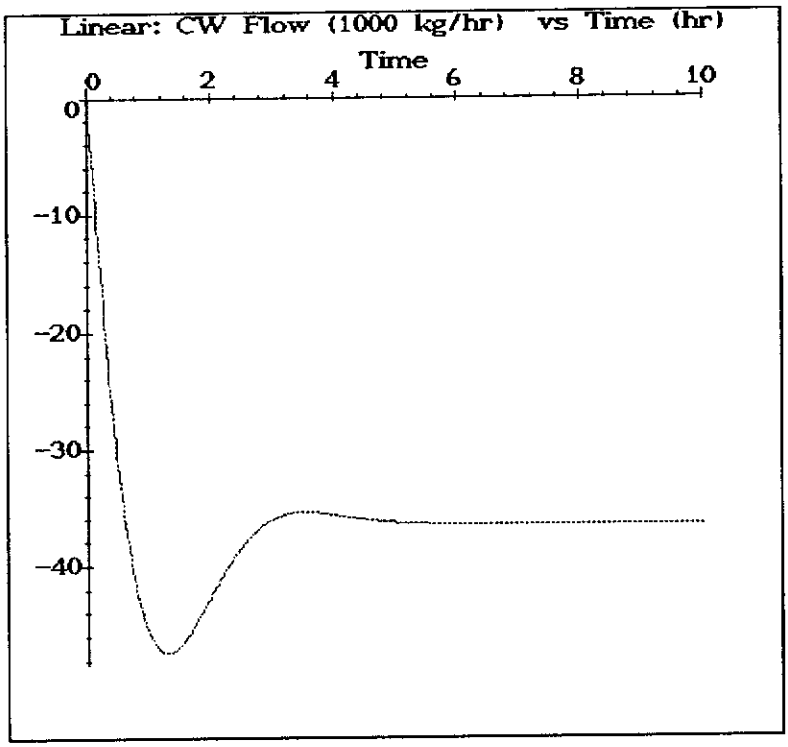


Figure 7.2 (f): — Q_{CW}

Figure 7.2: Closed Loop Dynamics of Triple Effects Evaporator using Linear Control (Nominal Case)

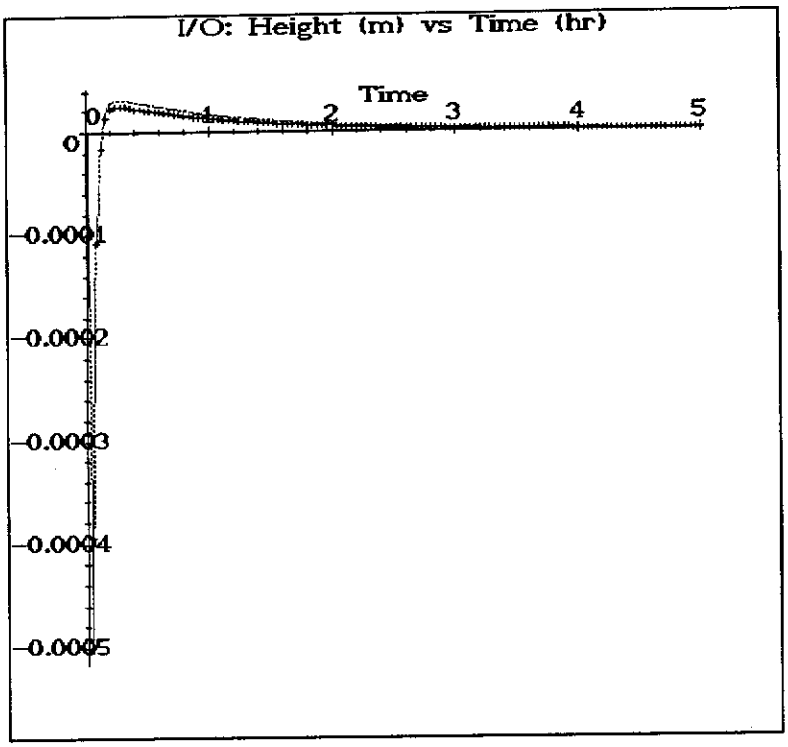


Figure 7.3 (a): $-h_1 \quad +h_2$

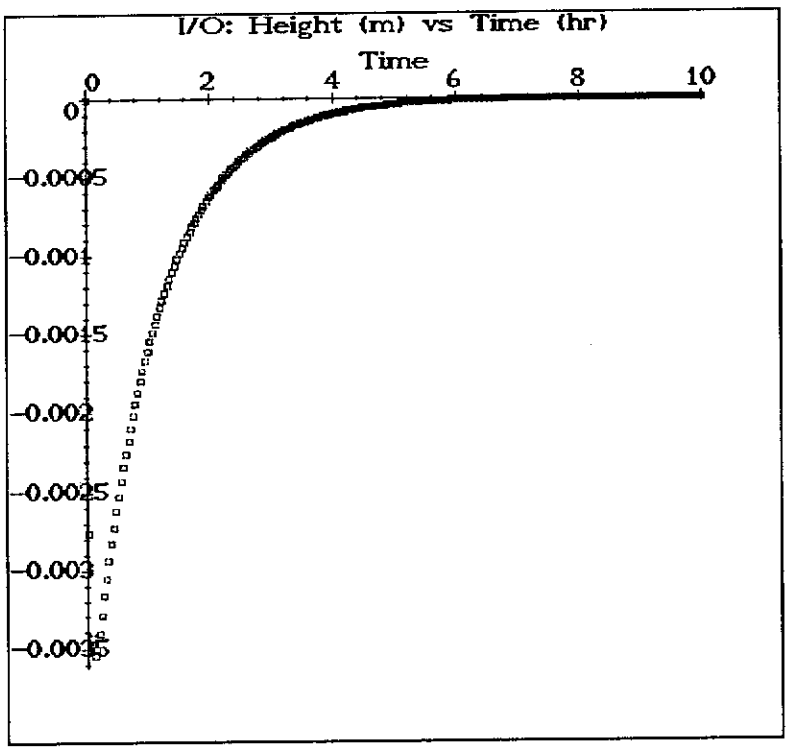


Figure 7.3 (b): $\square\, h_3$

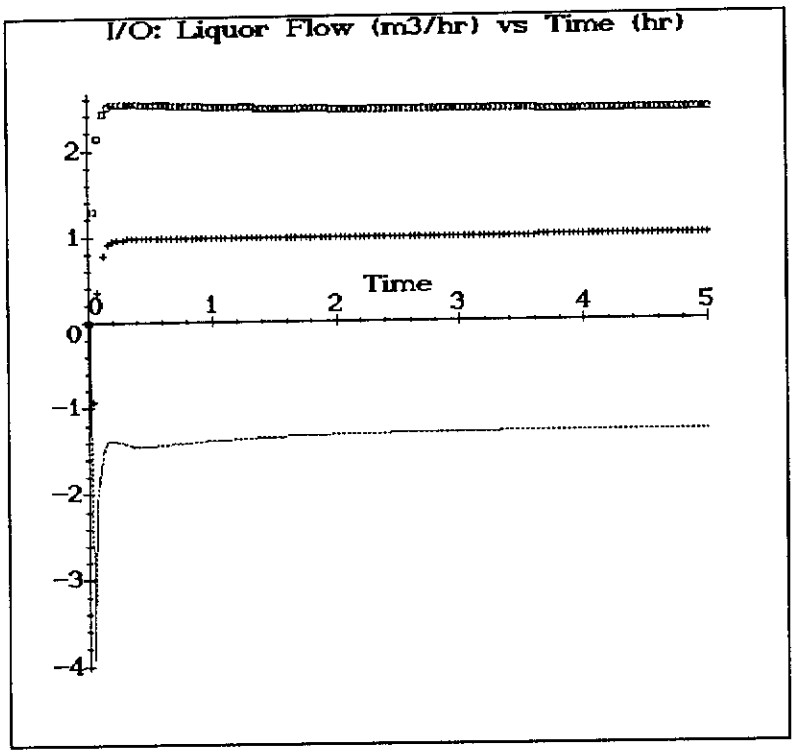


Figure 7.3 (c): —Q_F +Q_{P_1} □Q_{P_2}

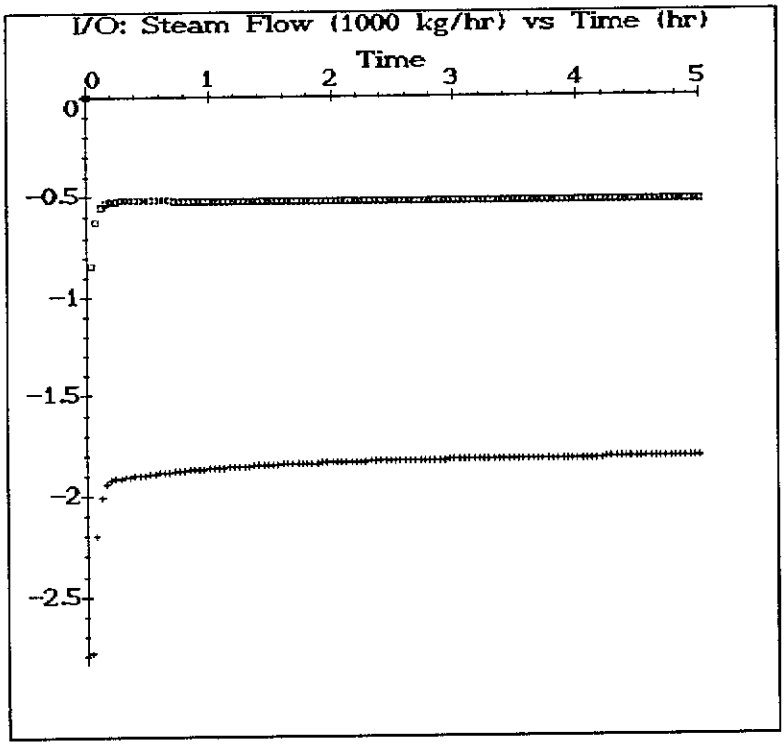


Figure 7.3 (d): + $\dot{m}_{S_2}$ □ $\dot{m}_{S_3}$

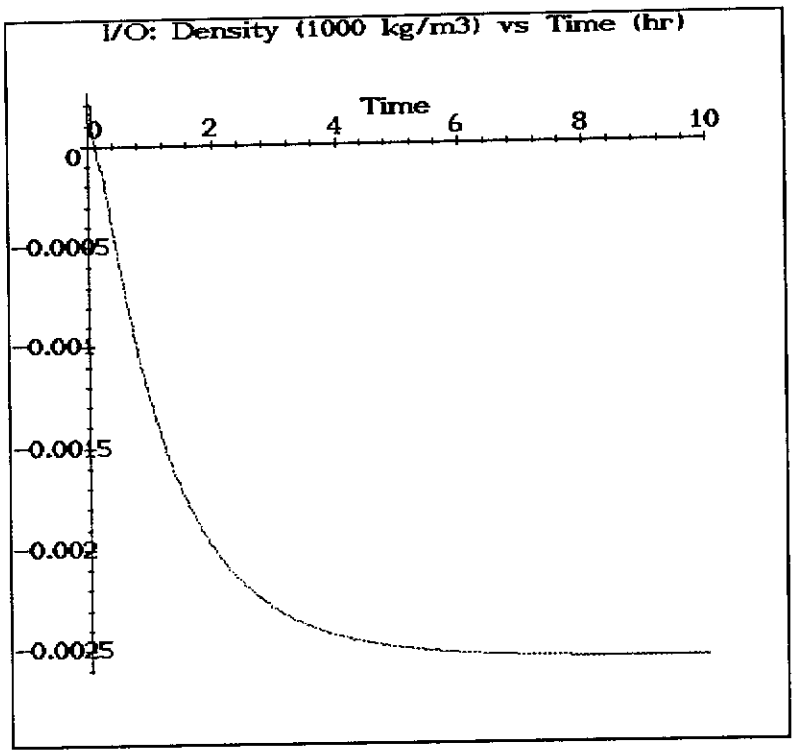


Figure 7.3 (e): — ρ_1

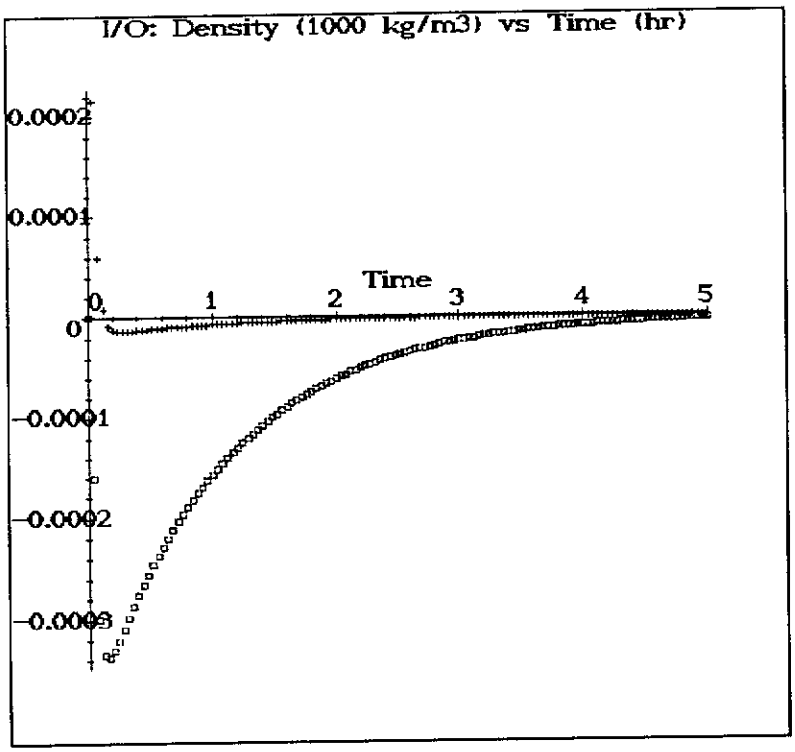


Figure 7.3 (f): + ρ_2 □ ρ_3

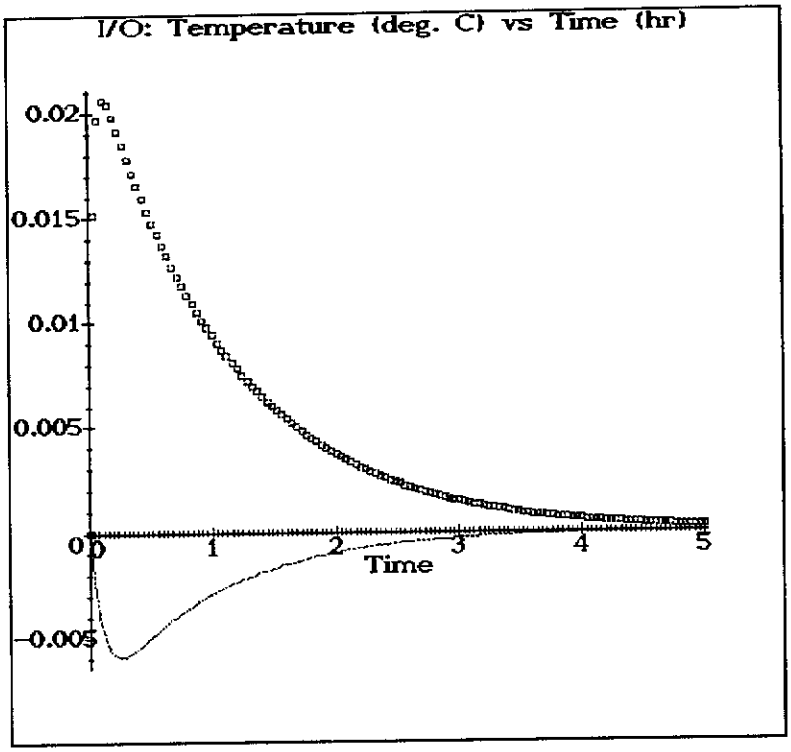


Figure 7.3 (g): — T_1 + T_2 □ T_3

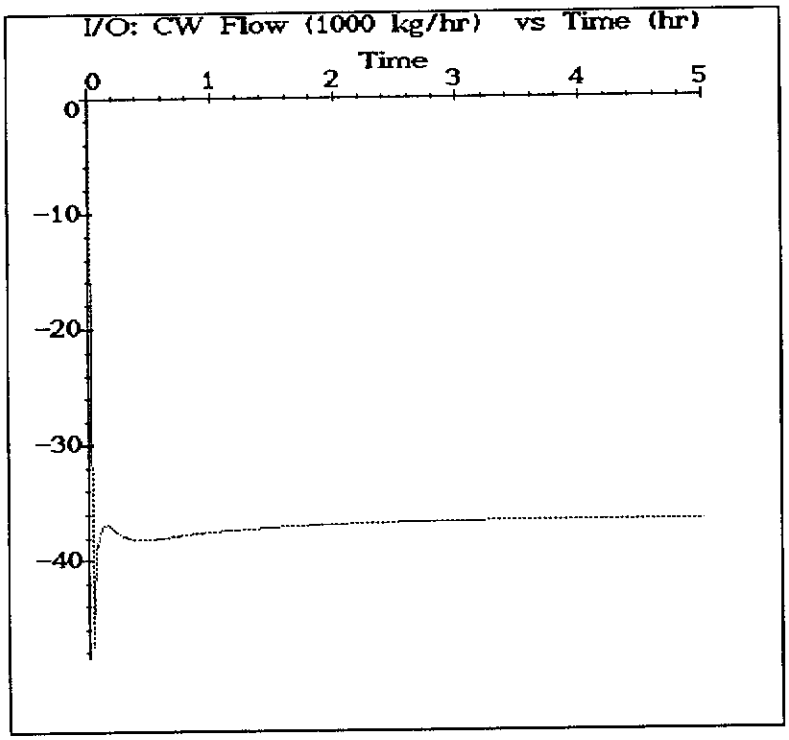


Figure 7.3 (h): — Q_{CW}

Figure 7.3: Closed Loop Dynamics of Triple Effects Evaporator using I/O Control (Nominal Case)

The control performance of the triple effects evaporator for the specific disturbance of a 3 m^3/hr increase in product flowrate obtained using the linear proportional and integral controller were found to be better than the case for the single effect evaporator, using the ITAE indices and the observed response curves (*Figures 7.2* and *7.3*). However, comparing the linear and the nonlinear controller, the performance of the nonlinear controller was about 10 to 1000 times better than the linear counterpart in terms of the ITAE indices shown in *Table 7.7*. The most obvious difference in the two control performances can be found in the response curves of the state variable T_2 (*Figures 7.2(e)* and *7.3(g)*). In both cases, the state variable T_2 was under open loop control. However, T_2 remained at approximately zero at all time in the case of the nonlinear controller while large offsets were observed in the linear control. The control actions provided by the nonlinear controller (*Figures 7.3(c), (d)* and *(h)*) were faster and more accurate in reaching set point values than the linear ones (*Figures 7.2(b), (d)* and *(f)*). The nonlinear controller counteracted the effect of the disturbance within the first sampling time while the linear one took more than 5 sampling times to respond. The responses obtained using the input output linearization can be considered as dead-beat responses. However, since the input dynamics were not considered here, dead-beat responses might not be desirable. The plant personnel confirmed that all the simulated control actions obtained here were feasible and implementable on the plant site.

7.3.3 Study of Robustness

The robustness of the linear and the nonlinear controller was studied by introducing parametric errors into the model. The results given below were obtained by introducing two parametric errors which were an increase of 9°C in the temperature approach T_a of the condenser and a decrease of 80 kg/m^3 in the feed density ρ_F. The UVA procedure was applied to the I/O to correct the degraded control performance. *Table 7.8* show the ITAE indices for local linearization and I/O with and without UVA. *Figures 7.4* and *7.5* provide the closed loop dynamics of the triple effects evaporator for the linear and nonlinear controllers, respectively. The control performance obtained using the I/O with uncertainty vector parameters of (10, 0, 0, 0, 0, 0, 10, 0, 0) is shown in *Figure 7.6*.

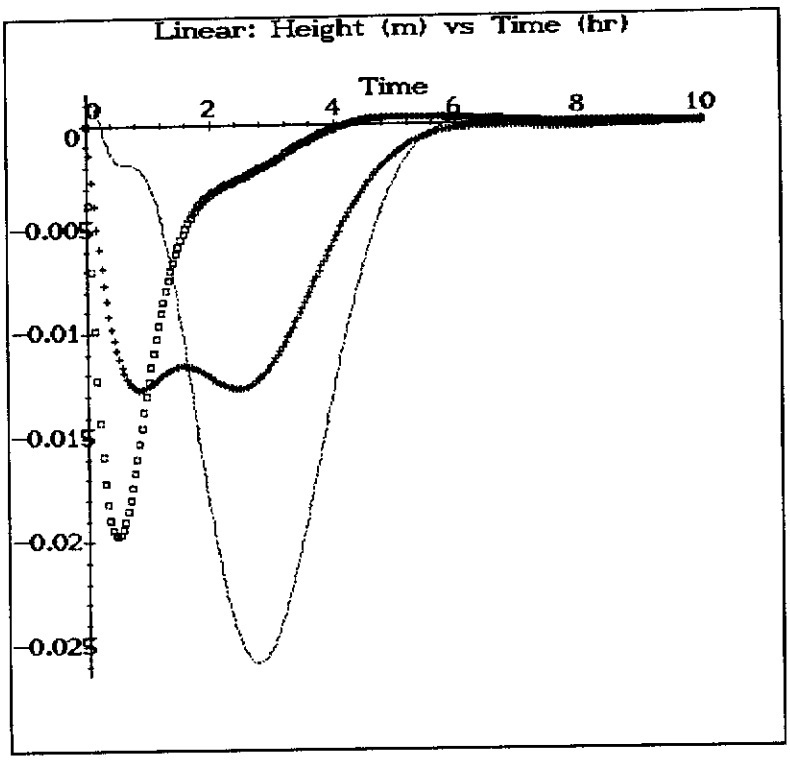


Figure 7.4 (a): —h_1 +h_2 □h_3

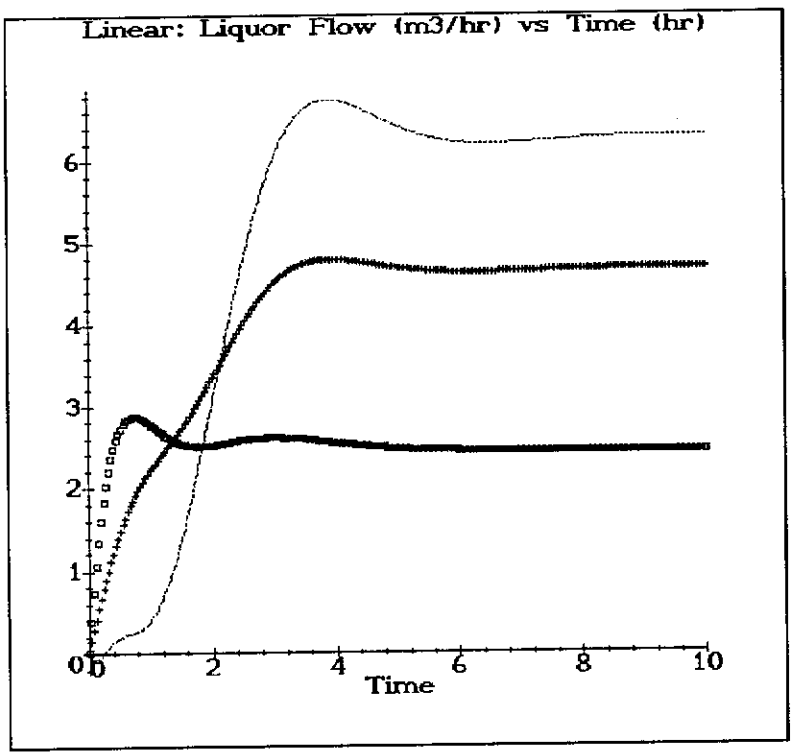


Figure 7.4 (b): —Q_F +Q_{P_1} □Q_{P_2}

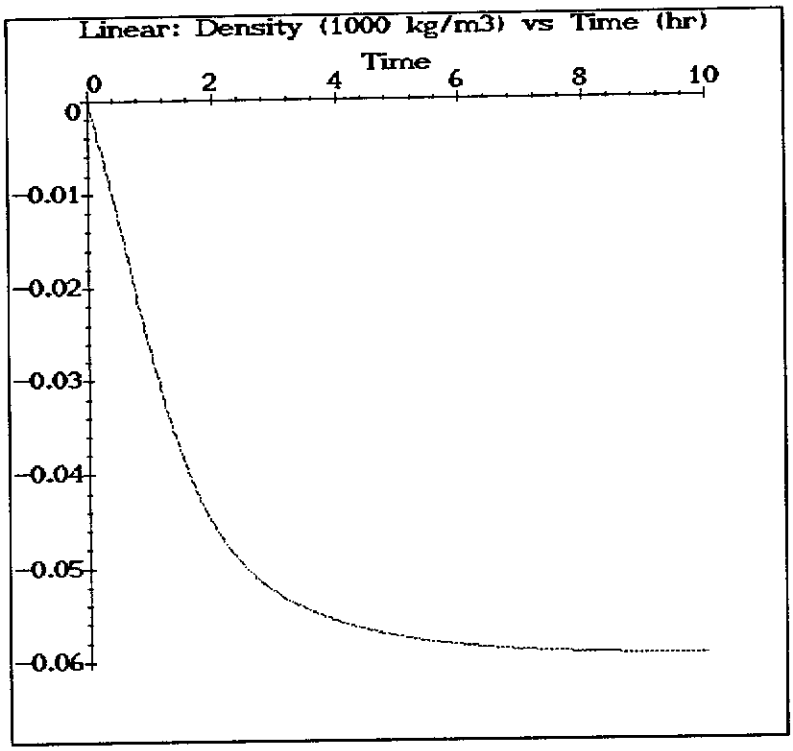


Figure 7.4 (c): — ρ_1

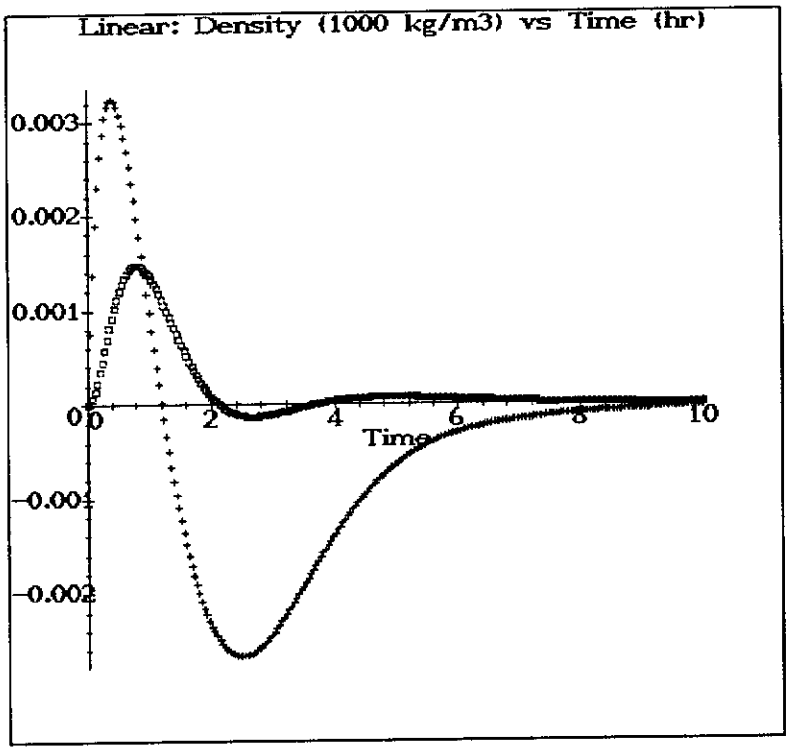


Figure 7.4 (d): + ρ_2 □ ρ_3

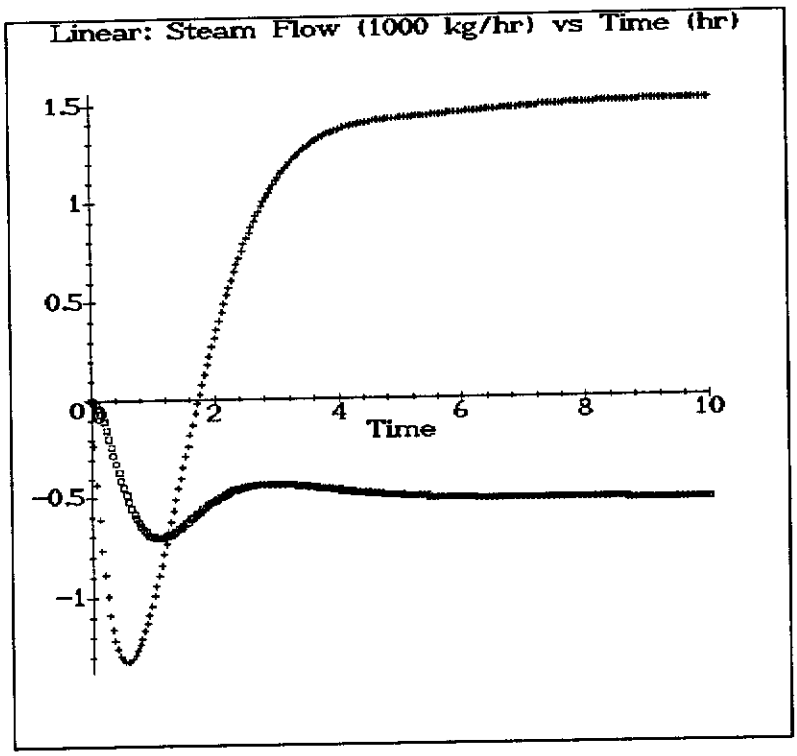


Figure 7.4 (e): + $\dot{m}_{S_2}$ □ $\dot{m}_{S_3}$

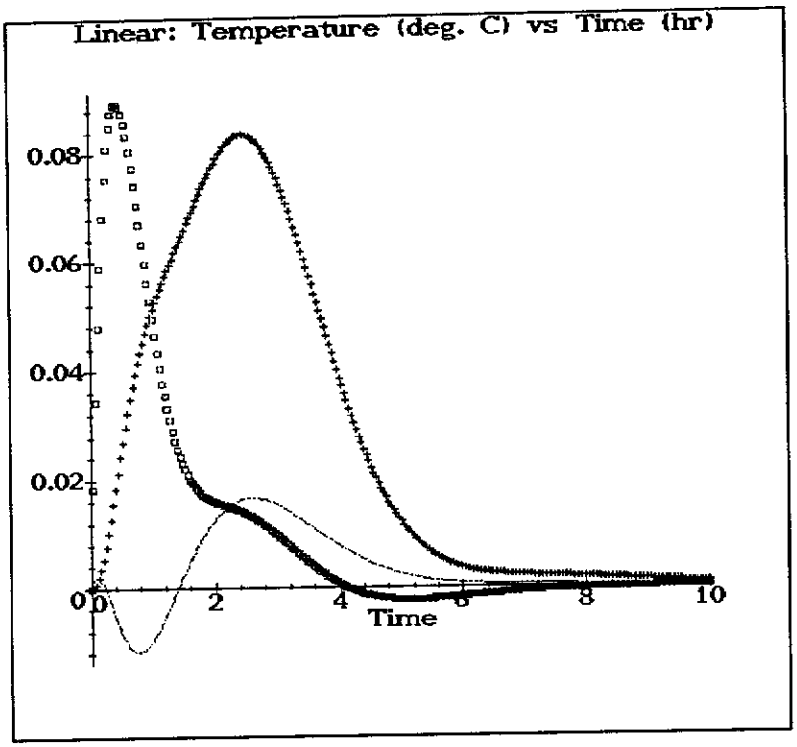


Figure 7.4 (f): — T_1 + T_2 □ T_3

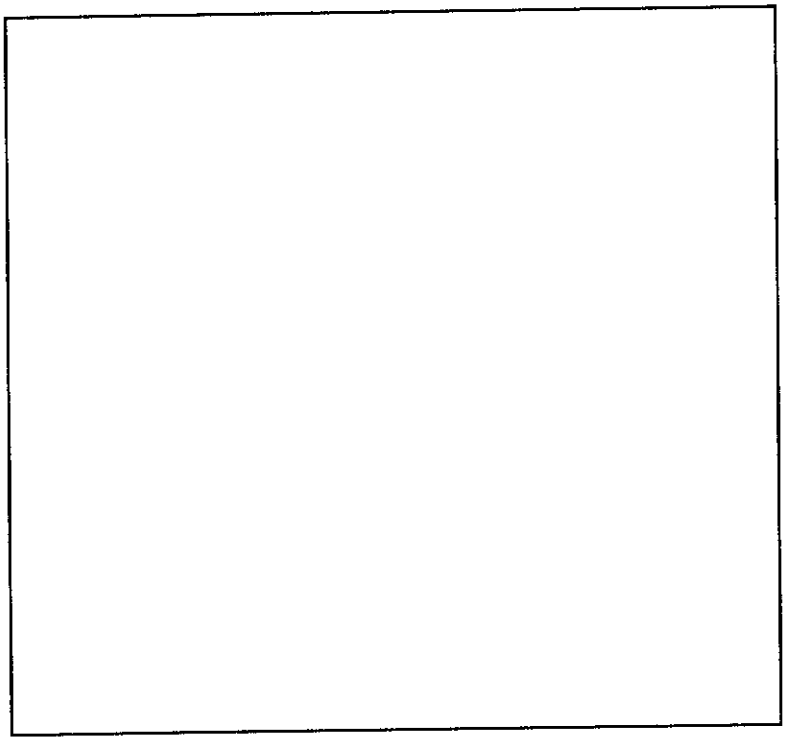


Figure 7.4 (g): — Q_{CW}

Figure 7.4: Closed Loop Dynamics of Triple Effects Evaporator using Linear Control (Robustness Study)

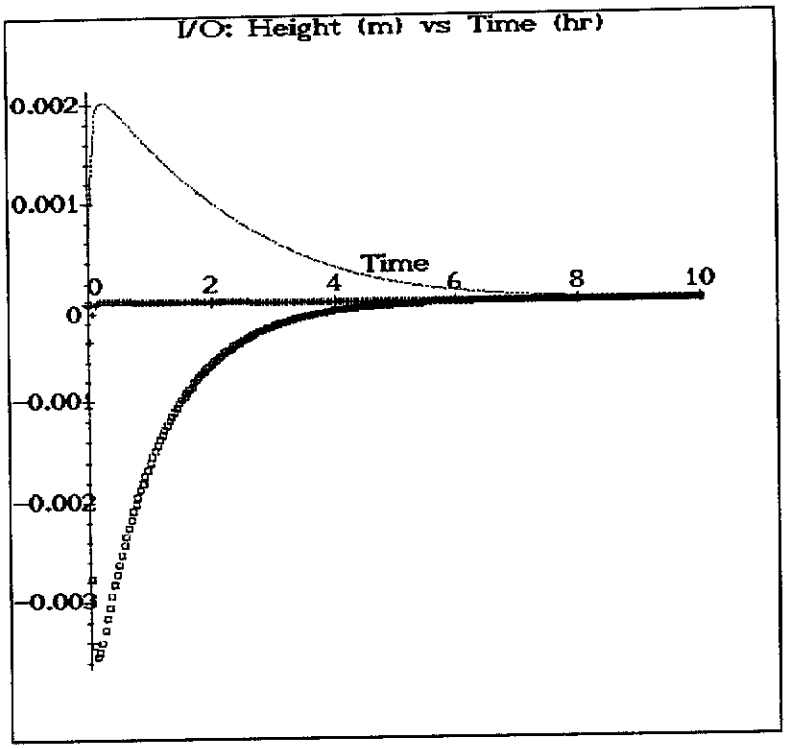


Figure 7.5 (a): — h_1 + h_2 □ h_3

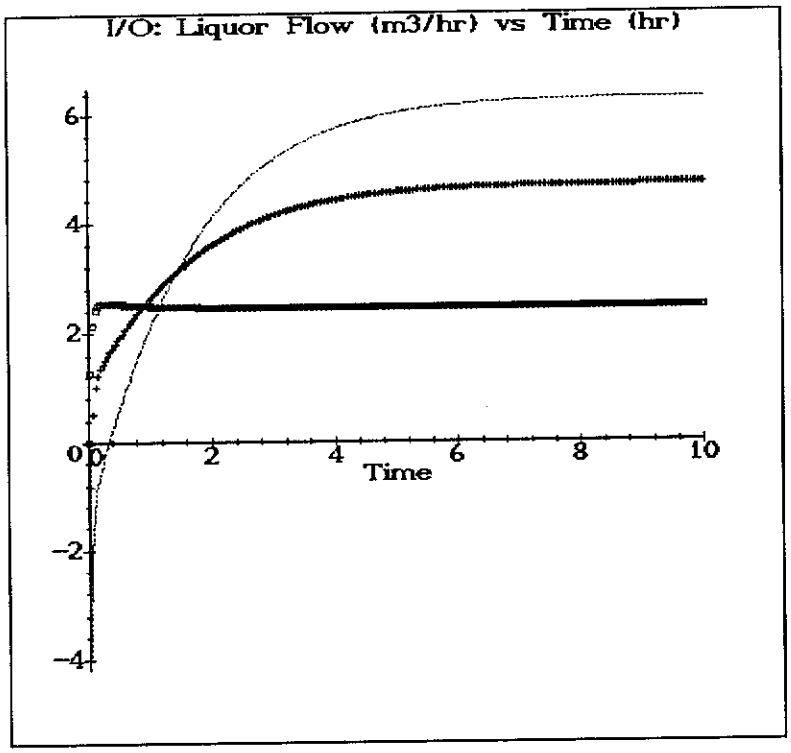


Figure 7.5 (b): — Q_F + Q_{P_1} □ Q_{P_2}

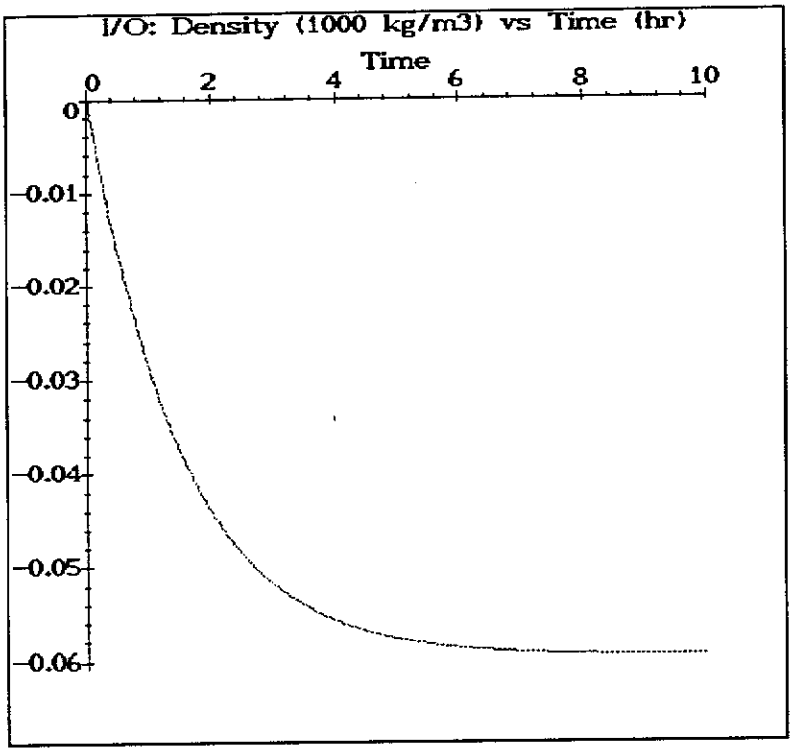


Figure 7.5 (c): — ρ_1

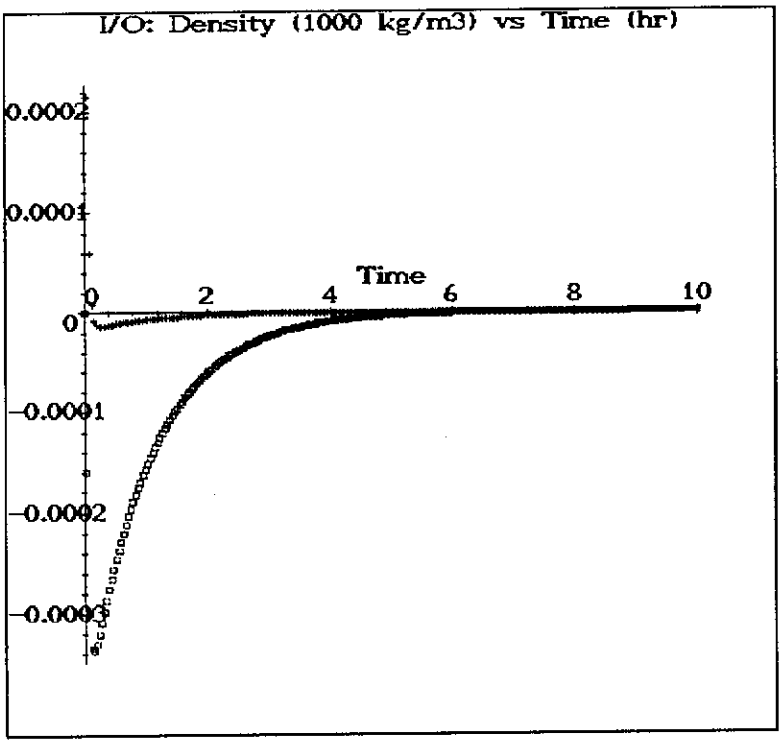


Figure 7.5 (d): + ρ_2 □ ρ_3

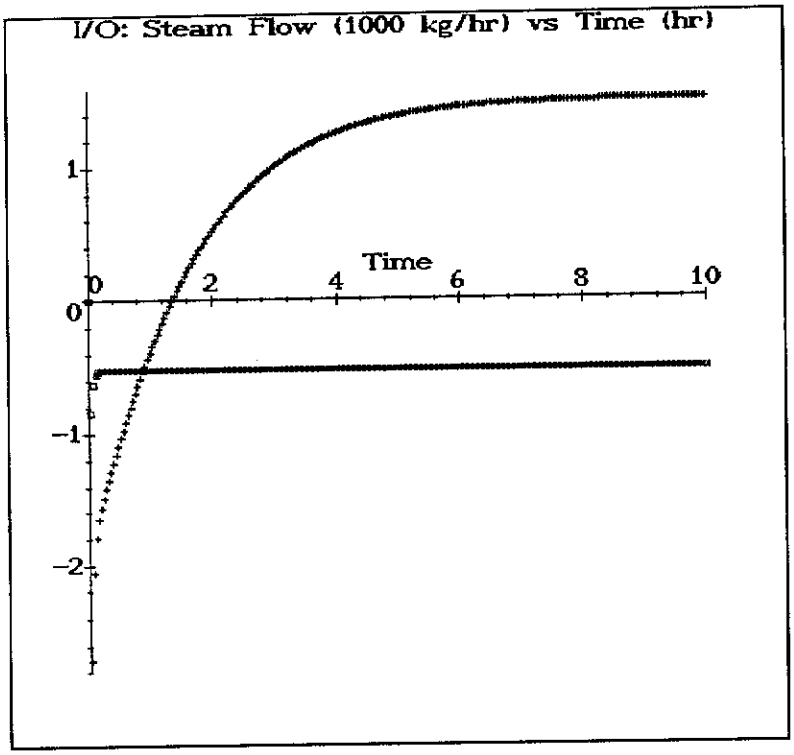


Figure 7.5 (e): + $\dot{m}_{S_2}$ □ $\dot{m}_{S_3}$

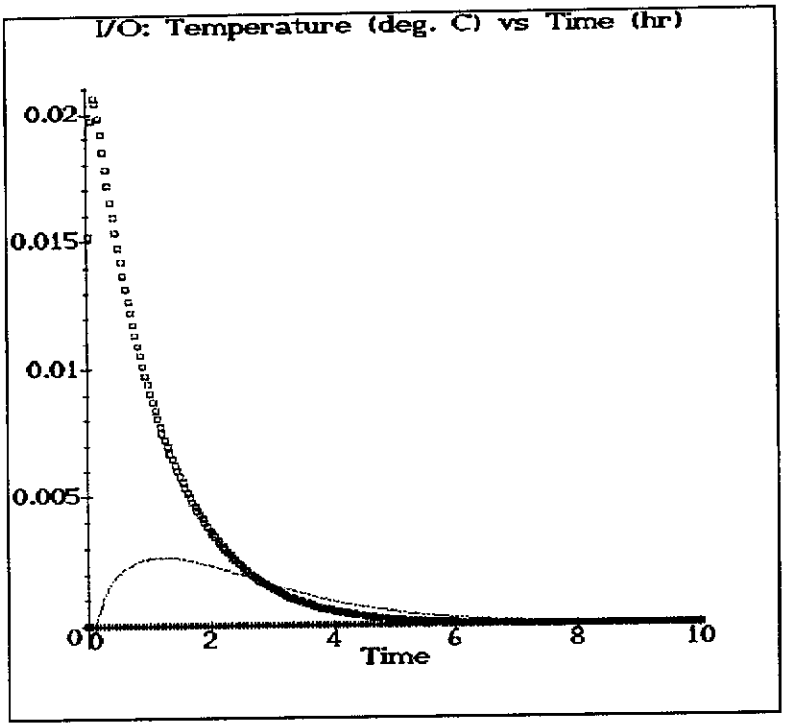


Figure 7.5 (f): — T_1 + T_2 □ T_3

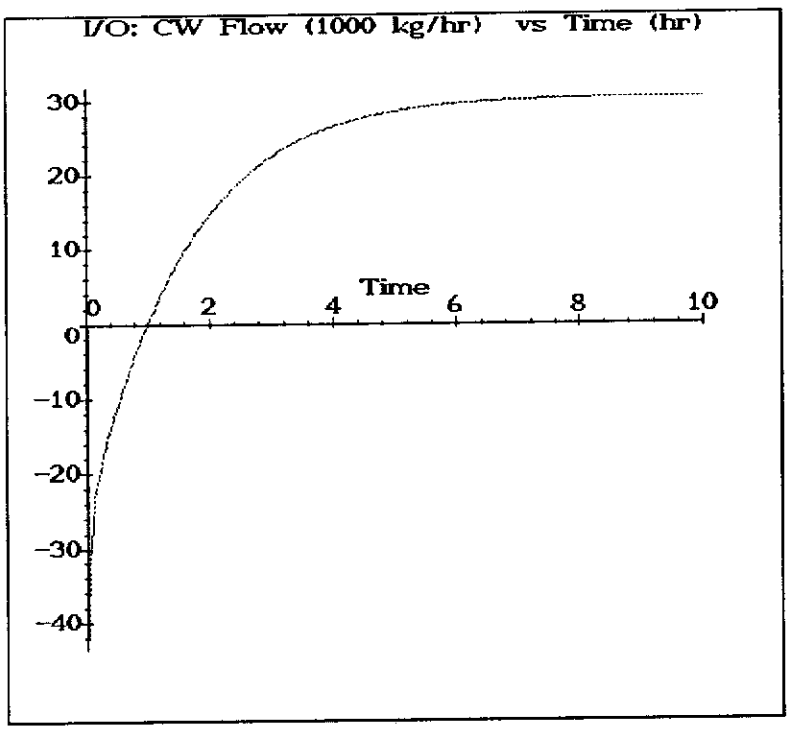


Figure 7.5 (g): — Q_{CW}

Figure 7.5: Closed Loop Dynamics of Triple Effects Evaporator using I/O Control (Robustness Study without UVA)

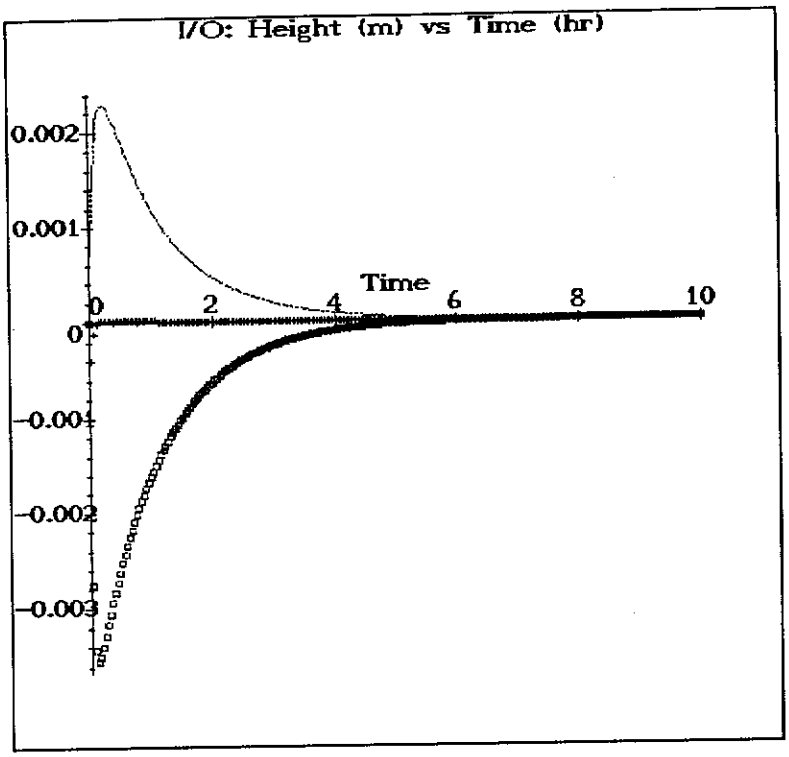


Figure 7.6 (a): — h_1 + h_2 □ h_3

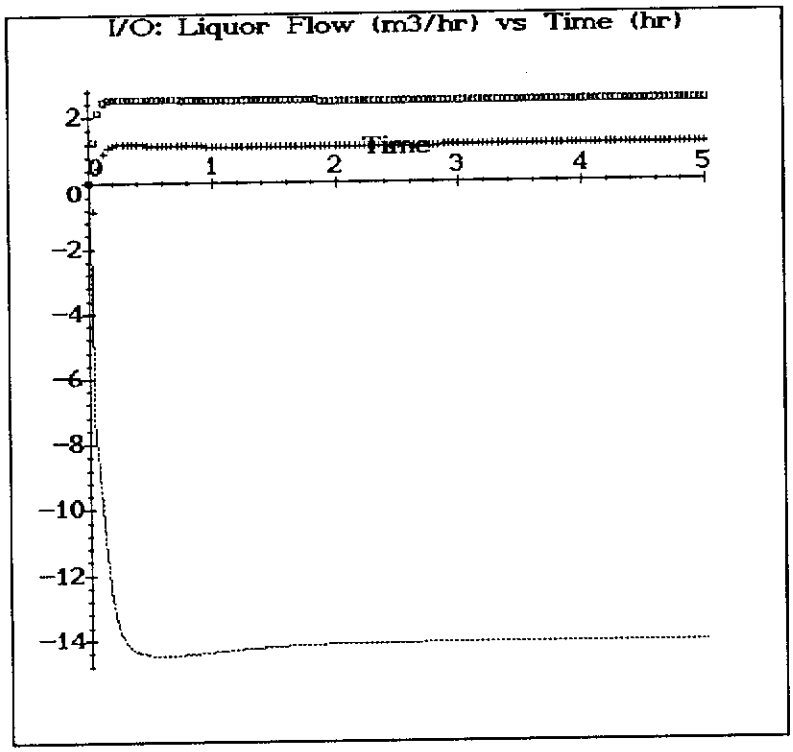


Figure 7.6 (b): — Q_F + Q_{P_1} □ Q_{P_2}

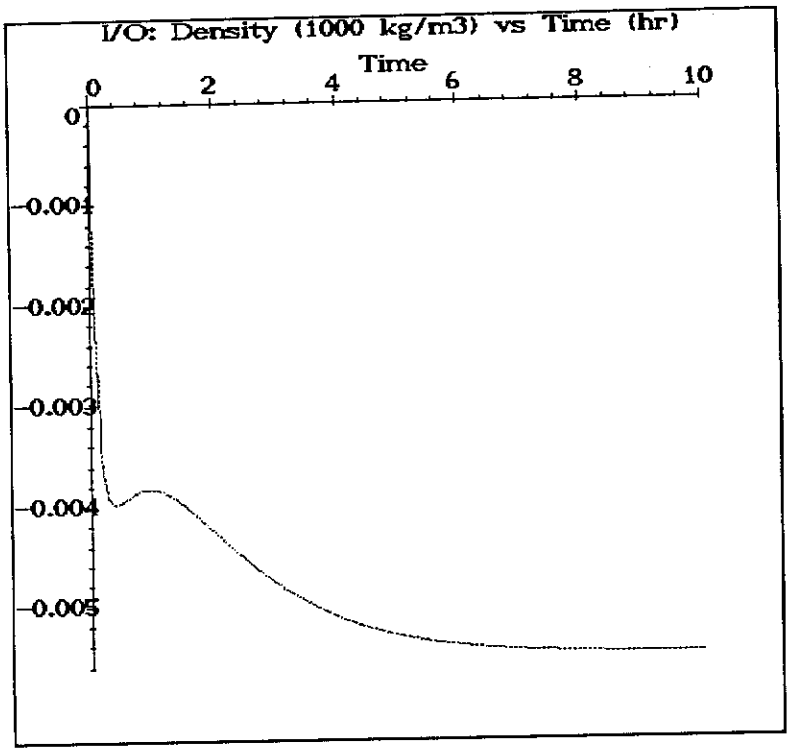


Figure 7.6 (c): — ρ_1

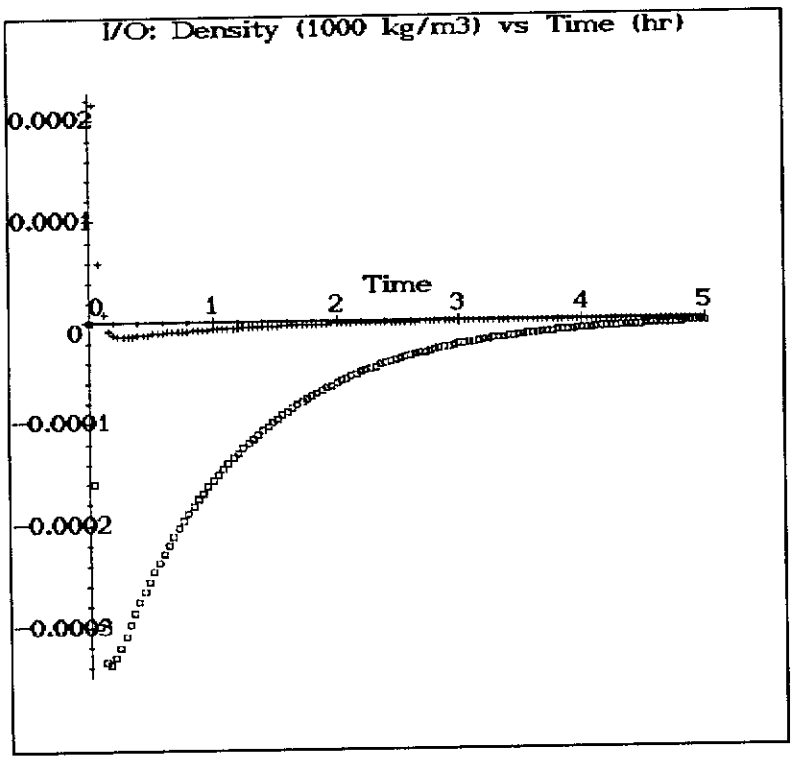


Figure 7.6 (d): + ρ_2 □ ρ_3

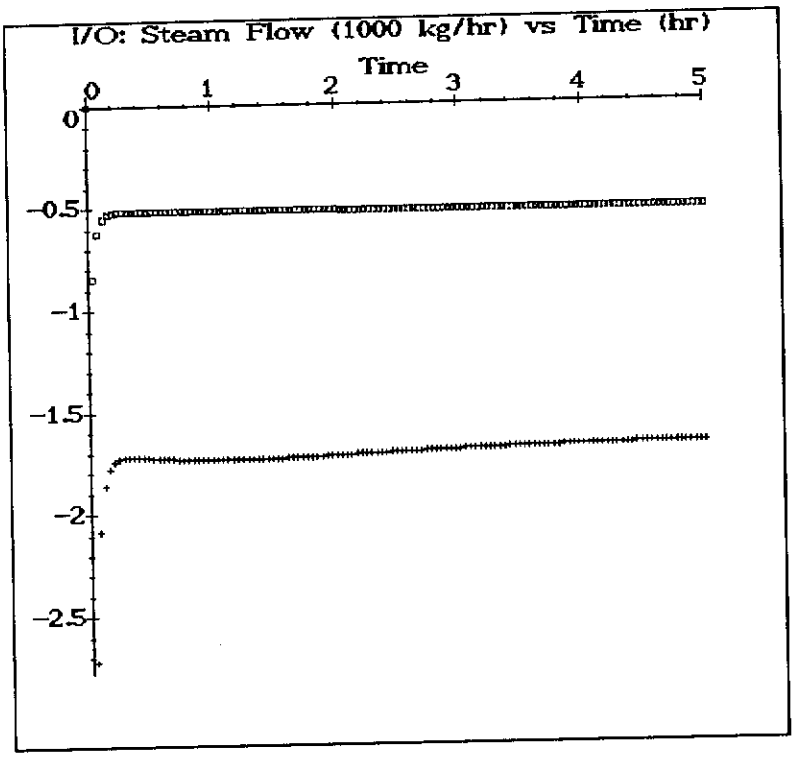


Figure 7.6 (e): + $\dot{m}_{S_2}$ □ $\dot{m}_{S_3}$

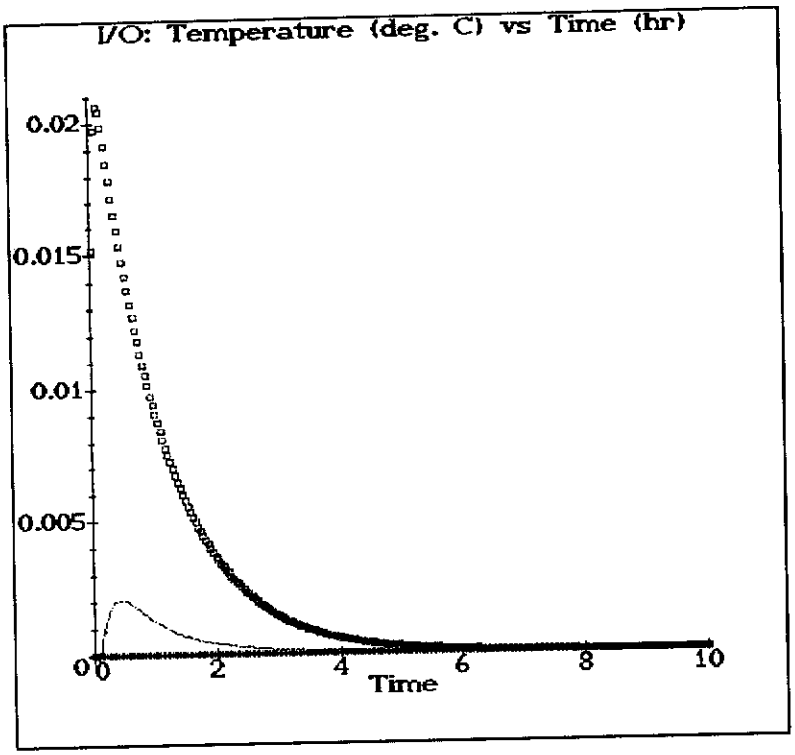


Figure 7.6 (f): — T_1 + T_2 □ T_3

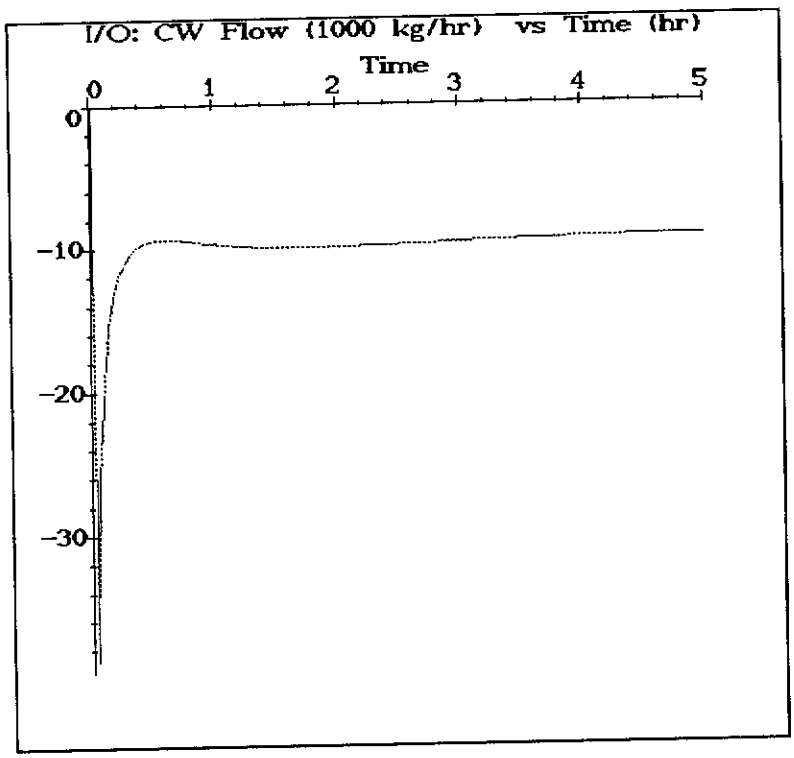


Figure 7.6 (g): — Q_{CW}

Figure 7.6: Closed Loop Dynamics of Triple Effects Evaporator using I/O Control (Robustness Study with UVA)

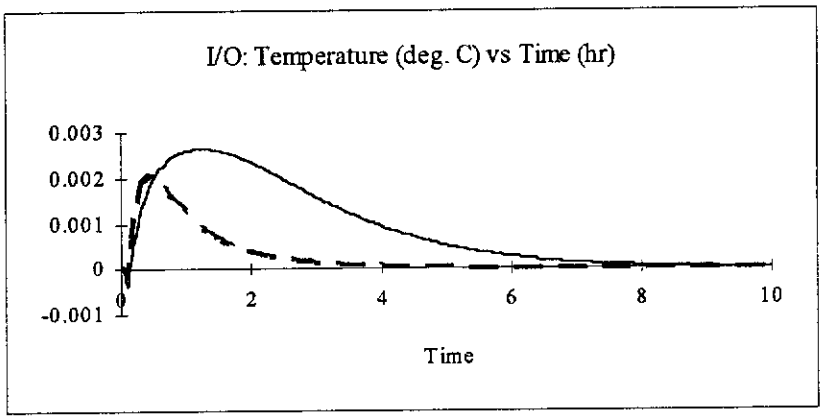


Figure 7.7 (a): — — with UVA—— without UVA

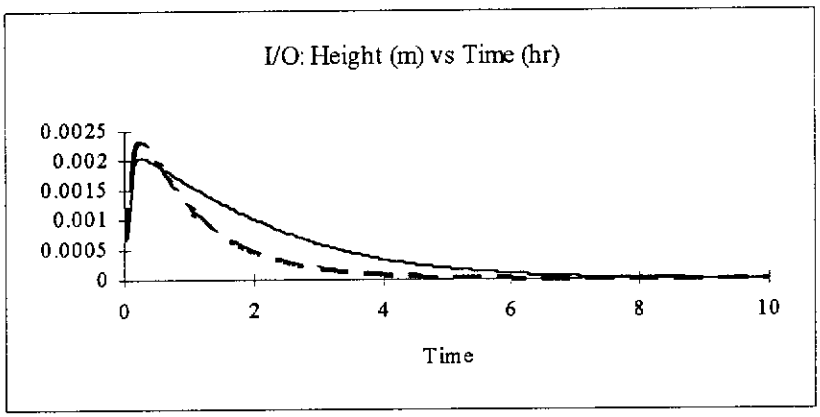


Figure 7.7 (b): — — with UVA—— without UVA

Figure 7.7: Closed Loop Responses of T_1 and h_1 of Triple Effects Evaporator

Table 7.8: ITAE for Robustness Study of Triple Effects Evaporator with Two Modelling Errors

Control Strategy	Local Linearization	I/O	
		without UVA	with UVA
h_1	0.186	0.00922	0.00366
h_2	0.106	0.0000379	0.0000379
h_3	0.0314	0.00465	0.00465
T_1	0.131	0.0218	0.00345
$\rho_{P_2} = \rho_2$	0.0260	0.0000209	0.0000209
$\rho_{P_3} = \rho_3$	0.00314	0.000452	0.000452

7.3.3.1 Discussions on Robustness Study without Uncertainty Vector Adjustment

For the linear controller, the deterioration in control performance was minimal when the parametric errors were introduced. The new ITAE indices were approximately 0 to 5 times higher than the nominal values (see *Tables 7.7* and *7.8*), which agrees with the previous conclusion that the single effect evaporator using the linear controller is more robust than the nonlinear one. The deteriorations in controller performances using I/O were only observed in the output variables h_1 and T_1 *(Figures 7.3(a), 7.3(g), 7.5(a)* and *7.5(f)*). The ITAE indices for h_1 and T_1 were approximately 10 and 100 times higher than the nominal situation (see *Tables 7.7* and *7.8*). In both situations of linear and nonlinear control schemes, the responses of the product density remained unchanged in spite of the introduction of the two parametric errors. This implied that the triple effects evaporator design is more robust to the parametric modelling errors tested here than the single effect design.

7.3.3.2 Discussions on Robustness Study Using Uncertainty Vector Adjustment

The deterioration observed in the control performances of the output variables h_1 and T_1 can be improved by applying the UVA procedure proposed in Chapter 5. The results in the previous section clearly demonstrate the effectiveness of the proposed adjustment. Since only the control performances of h_1 and T_1 deteriorated, the uncertainty vector parameters used were (10, 0, 0, 0, 0, 0, 10, 0, 0). Also, because the model equations for the triple effects evaporator design are less susceptible to changes, it was necessary to use larger values of adjustment parameters, for example, a value of 10, to correct the degraded control performances. *Table 7.8* shows that the ITAE indices for h_1 and T_1 obtained using the UVA were approximately 39% and 15% of the values without UVA, respectively. The response curves of h_1 and T_1 for the cases with and without the UVA are given in *Figure 7.7*, and the improvements in performances can be clearly seen in this figure. The uncertainty vector adjusted control actions, shown in *Figures 7.6*, were completely different from the situations without the UVA (*Figures 7.5*). Despite of the great similarity between the adjusted controls and the control action determined for the nominal situation (*Figures 7.3*),

the final steady state in *Figure 7.6* is different from those in the other figures. It implies that there is a possiblity of multiple steady states existed in the system. Further investigation is necessary to study the steady state and dynamic behaviours of the system.

7.4 FURTHER TEST FOR UNCERTAINTY VECTOR ADJUSTMENT

The effectiveness of the UVA was further examined by introducing more parametric errors into the system with the uncertainty vector parameters fixed at (10, 0, 0, 0, 0, 0, 10, 0, 0). This test demonstrates that the proposed procedure can actually improve the robustness of a nonlinear control system.

The additional modelling errors were a 4°C increase in feed temperature T_F, a 3°C increase in cooling water temperature T_{CW}, and a 5 J/kg/°C increase in the specific heat capacity of the product stream $C_{P,3}$. Therefore, there were five modelling errors in total. The simulations were carried out in the same way as described in section 7.3 of this chapter. The response curves of the output, state and input variables with and without the UVA are very similar to those observed in the case of two modelling errors (*Figures 7.5* and *7.6*), and therefore, they are omitted to avoid unnecessary repetitions. The ITAE indices for the scenario with and without the UVA are given in *Table 7.9* below. Once again, only the control performances of h_1 and T_1 deteriorated further due to the three additional modelling errors. The ITAE indices of the rest of the output variables remained unaffected and were about the same as their nominal values. When the UVA with (10, 0, 0, 0, 0, 0, 10, 0, 0) as the parameters was applied, the performances greatly improved. Comparing this with the case without the adjustment, the percentages of improvements were approximately 80% and 90%, respectively, which were even higher than in the case for the two modelling errors. The reason is that the performances in this case deteriorated more than before and hence there was a bigger margin for the adjustment to improve.

Since there was no change in the simulation conditions and the same uncertainty vector parameters were used, the introduction of the UVA to the control system has not only corrected the modelling errors and has thereby improved the control

performance, but has also successfully improved the robustness of the I/O.

Table 7.9: ITAE for Robustness Study of Triple Effects Evaporator with Five Modelling Errors

Control Strategy	I/O without UVA	I/O with UVA
h_1	0.0132	0.00292
h_2	0.0000379	0.0000379
h_3	0.00463	0.00463
T_1	0.0370	0.00412
$\rho_{P_2} = \rho_2$	0.0000209	0.0000209
$\rho_{P_3} = \rho_3$	0.000456	0.000456

7.5 CONCLUSIONS

The study in this chapter has extended the results of previous investigations to a higher level by designing a nonlinear control scheme for a triple effects evaporator. The triple effects evaporator was designed on the basis of a new five effects evaporator in Alcoa's Wagerup refinery. A dynamic model for the triple effects evaporator was constructed by performing mass and energy balances around the system. The control schemes used were the classic PI linear controller and the modified I/O for a general class of nonlinear processes. The SHM transformation is not applicable in this case because it is restricted to input-linear systems only. The input and output variables were selected based on: the results of the single effect evaporator presented in Chapter 3; the advice from plant personnel; and the analysis of the characteristic matrix of the linearized model. The simulation results showed that the control obtained using the I/O was superior than the one using the linear controller, based on the ITAE indices. The robustness study demonstrated the effectiveness of the proposed UVA by individually introducing two and five modelling errors into the dynamic model of the triple effects evaporator system. Overall, compared with the single effect evaporator design, the triple effects evaporator is a better design in terms of energy saving, control and robustness. Furthermore, the use of a more complicated model in this chapter has successfully proved that the proposed UVA procedure of Chapter 5 has the ability to correct the adverse effects of

modelling errors and to improve the robustness of a nonlinear control strategy.

CHAPTER EIGHT

USE OF STATE OBSERVERS IN NONLINEAR CONTROL

8.1 INTRODUCTION

Geometric nonlinear control strategies explored in this thesis assume the complete accessibility of the states for the computation of the transformation relationships. Since this is not usually the case in real plant situations (due to one reason or the other), the design of an observer to generate an estimate that converges asymptotically to the actual states is necessary. The state estimates can then be used to implement the nonlinear control algorithm. This chapter evaluates two types of state observers, which are the Geometric Nonlinear Filter (GNF) developed by Bestle and Zeitz (1983) and the Extended Kalman Filter (EKF) (Watanabe, 1992) for chemical engineering applications.

The target system used in this study is the single effect evaporation unit of the liquor burning process described in Chapter 3. MAPLE V.3 is once again used as the computing environment. According to Kailath (1980), it is convenient to set the input at any time to zero for the state observer design since the effect of a nonzero (but known) input is merely to change the values of the state vector. This assumption will be used here. Therefore, the open loop situation will be investigated to briefly illustrate the idea of GNF and EKF. The results obtained here using the open loop state observer are equally applicable for the closed loop situation.

8.2 BRIEF LITERATURE REVIEW ON GEOMETRIC NONLINEAR FILTERS

The basic idea of the design method of GNF is the same as that of the nonlinear controller itself. It transforms the nonlinear model to a linear form, known as

observer canonical form, through the use of a nonlinear state transformation and output injection so that the well-developed linear theory can be used in the filter gain computation.

The existence of transformations to the observer canonical form has been addressed by Thau (1973) and Bestle and Zeitz (1983). Thau (1973) gave a sufficient condition for the asymptotic stability of the origin of the error differential equation. Assuming that a transformation exists, Bestle and Zeitz (1983) provided an alternative approach to obtain the GNF design without actually computing the transformation. This alternative GNF design is used in this study. A discussion of the GNF design by Bestle and Zeitz (1983) is given below and it can also be found in Walcott et al (1987) and Isidori (1995).

Another GNF design introduced by Krener (1984) and Frezza et al (1988) is the asymptotic GNF technique which uses the modified and approximate observer canonical form. This technique is a combined structure of the Bestle and Zeitz's GNF design and Kalman filter. As a result, it accounts for state and measurement perturbations and results in solutions very similar in nature to the Kalman filter. Unlike the Bestle and Zeitz's GNF, this design requires explicit determination of the transformation which could present a problem in chemical engineering application because of the complex mathematical manipulations of the necessary equations which are usually highly nonlinear. To the best of the author's knowledge, the GNF has been mostly used in the electrical and mechanical engineering areas and this is the first time that this technique will be applied to a chemical process system.

8.3 GEOMETRIC NONLINEAR FILTER

The following sections briefly describe the method of GNF for a MIMO nonlinear system. The relevant equations for obtaining the transformation relationships which transforms the nonlinear system into the observer canonical form are given. The derivation of these equations are omitted since further details are available in Walcott et al (1987). The observer design is discussed in section 8.3.2.

8.3.1 Transformation of Nonlinear Systems into Observer Form

We consider the following multivariable autonomous nonlinear system with w outputs,

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}) = \begin{bmatrix} h_1(\mathbf{x}) \\ \vdots \\ h_w(\mathbf{x}) \end{bmatrix} \end{aligned} \quad \text{..... (8.1)}$$

where $\mathbf{f}$ is a C^∞ vector field on $\mathbf{R}^n$ and $h_1(\mathbf{x}),\ldots,h_w(\mathbf{x})$ are scalar fields on $\mathbf{R}^n$. In other words, the mapping $\mathbf{f}$ and the functions $h_1(\mathbf{x}),\ldots,h_w(\mathbf{x})$ are smooth in their arguments. $\mathbf{y}$ and $\mathbf{x}$ are vectors with $\mathbf{y} \in \mathbf{R}^w$ and $\mathbf{x} \in \mathbf{R}^n$, respectively. It is desired to determine a nonlinear transformation $\mathbf{T}$: $\mathbf{R}^n \rightarrow \mathbf{R}^n$, where

$$\mathbf{x} = \mathbf{T}(\xi) \quad \text{..... (8.2)}$$

such that (8.1) is transformed into the MIMO observer canonical form shown below.

$$\begin{aligned} \dot{\xi} &= \begin{bmatrix} \mathbf{B}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{B}_w \end{bmatrix} \xi - \begin{bmatrix} \phi_1(\mathbf{y}) \\ \vdots \\ \phi_n(\mathbf{y}) \end{bmatrix} \\ \mathbf{y} &= \begin{bmatrix} \xi_{l_1} \\ \xi_{l_1+l_2} \\ \vdots \\ \xi_{l_1+l_2+\ldots+l_w} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \\ \vdots \\ \mathbf{C}_w \end{bmatrix} \xi \end{aligned} \quad \text{..... (8.3)}$$

where the square matrices $\mathbf{B}_i \in \mathbf{R}^{l_i \times l_i}$ and the row matrices $\mathbf{C}_i \in \mathbf{R}^{1 \times n}$ are

$$\mathbf{B}_i = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & & & \vdots \\ & \ddots & & 0 \\ 0 & & 1 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{C}_i = \begin{bmatrix} \overbrace{0 \quad \cdots \quad 0 \quad 1}^{l_1+\cdots+l_i} \quad 0 \quad \cdots \quad 0 \\ \underbrace{\qquad\qquad\qquad\qquad\qquad\qquad}_{n} \end{bmatrix}$$

with $i = 1,\cdots,w$. The integers $l_1,\cdots,l_w$ are the observability indices defined below.

Observability

The nonlinear system (8.1) is observable if the following observability matrix is non-singular on $\mathbf{R}^n$ near the steady state.

$$\overline{\mathbf{O}} = \begin{bmatrix} d\left(L_f^0 h_1\right) \\ d\left(L_f^1 h_1\right) \\ \vdots \\ d\left(L_f^{l_1-1} h_1\right) \\ \hdashline \vdots \\ \hdashline d\left(L_f^0 h_w\right) \\ d\left(L_f^1 h_w\right) \\ \vdots \\ d\left(L_f^{l_w-1} h_w\right) \end{bmatrix} \quad \text{..... (8.4)}$$

where $l_1,\cdots,l_w$ are non-negative integers and $l_1 + l_2 + \cdots + l_w = n$ and the functional operator d denotes the gradient of a scalar field.

Starting Vectors

The starting vectors for generating the Jacobian matrix of the transformation can be determined using the observability indices and matrix. The proof can be found in Walcott et al (1987) and Isidori (1995). The w starting vectors are:

$$\begin{bmatrix} \dfrac{\partial \mathbf{T}}{\partial \xi_1} & \dfrac{\partial \mathbf{T}}{\partial \xi_{1+l_1}} & \cdots & \dfrac{\partial \mathbf{T}}{\partial \xi_{1+l_1+\cdots+l_{w-1}}} \end{bmatrix} = \overline{\mathbf{O}}\,\mathbf{C}^{\mathrm{T}} \quad \text{..... (8.5)}$$

where T denotes the matrix transpose.

Jacobian of Transformation

With the w starting vectors determined, one can proceed to obtain the following equation for the Jacobian of the desired transformation:

$$\frac{\partial \mathbf{T}}{\partial \xi} = \left[ad^0\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \xi_1} \quad \cdots \quad ad^{l_1-1}\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \xi_1} \;\middle|\; ad^0\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \xi_{1+l_1}} \quad \cdots \quad ad^{l_2-1}\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \xi_{1+l_1}} \;\middle|\; \cdots \;\middle|\; ad^0\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \xi_{1+l_1+\cdots+l_{w-1}}} \quad \cdots \quad ad^{l_w-1}\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \xi_{1+l_1+\cdots+l_{w-1}}} \right] \quad \text{..... (8.6)}$$

The transformation (8.2) is obtained by integrating the above Jacobian matrix.

8.3.2 Observer Design

For the MIMO system (8.1), the observer is governed by the following equation.

$$\dot{\hat{\xi}} = \begin{bmatrix} \mathbf{B}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{B}_w \end{bmatrix} \hat{\xi} - \begin{bmatrix} \phi_1(\mathbf{y}) \\ \vdots \\ \phi_n(\mathbf{y}) \end{bmatrix} - \mathbf{K}(\hat{\mathbf{y}} - \mathbf{y}) \quad \text{..... (8.7)}$$

where ^ denotes the estimated value and

$$\mathbf{K} = \begin{bmatrix} k_{1,0} & 0 & \cdots & 0 \\ \vdots & \vdots & & \\ k_{1,l_1-1} & 0 & & \\ 0 & k_{2,0} & & \vdots \\ & \vdots & \cdots & \\ & k_{2,l_2-1} & & \\ \vdots & & & 0 \\ & 0 & & k_{w,0} \\ & \vdots & & \vdots \\ 0 & 0 & \cdots & k_{w,l_w-1} \end{bmatrix}$$

Now, we define the error as

$$\mathbf{e} = \hat{\xi} - \xi \quad \text{..... (8.8)}$$

Hence, the error satisfies the linear differential equation:

$$\dot{\mathbf{e}} = \begin{bmatrix} \hat{\mathbf{B}}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \hat{\mathbf{B}}_w \end{bmatrix} \mathbf{e} \qquad (8.9)$$

where

$$\mathbf{B}_i = \begin{bmatrix} 0 & 0 & \cdots & -k_{i,0} \\ 1 & & & -k_{i,1} \\ & \ddots & & \vdots \\ 0 & & 1 & -k_{i,l_i-1} \end{bmatrix} \qquad (i = 1,\ldots,w)$$

The characteristic polynomial for the error differential equation (8.9) is

$$P(s) = \prod_{i=1}^{w} \left(k_{i,0} + k_{i,1}\, s + \cdots + k_{i,l_i-1}\, s^{l_i-1} + s^{l_i} \right) \qquad (8.10)$$

Hence, the convergence behaviour of (8.9) can be easily assigned by the appropriate choice of the elements of **K**. The condition for stability is that all the roots of (8.10) must have negative real parts.

8.3.3 Alternative Observer Design

As indicated in the above section, the determination of the transformation involves integrating the Jacobian matrix with the nonlinear equations as its elements. In other words, we need to solve a system of partial differential equations. The current version of MAPLE V.3 is unable to handle partial differential equations. Therefore, an approximation of the transformation is necessary. Bestle and Zeitz (1987) suggested an alternative approach based on linearizing the observer equation (8.7) about the estimates $\hat{\xi}_i$ where $i = l_1, l_1 + l_2, \cdots, l_1 + l_2 + \cdots + l_w$. For (8.1), the alternative design is given by the following equation:

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}) + \mathbf{g}(\hat{\mathbf{x}})\left(\mathbf{y} - \mathbf{h}(\hat{\mathbf{x}})\right) \qquad (8.11)$$

where

$$\mathbf{g}(\hat{\mathbf{x}}) = \frac{\partial \mathbf{T}}{\partial \hat{\xi}}\mathbf{K} - \left[\left[\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \hat{\xi}_{l_1}}\right] \left[\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \hat{\xi}_{l_1+l_2}}\right] \cdots \left[\mathbf{f}, \frac{\partial \mathbf{T}}{\partial \hat{\xi}_{l_1+l_2+\cdots+l_w}}\right]\right]$$

Equation (8.11) will be used to obtain the observer design for the single effect evaporator given in the later section.

8.4 EXTENDED KALMAN FILTER

This section reviews the design method of the discrete-time extended Kalman filter for a nonlinear system having Gaussian white noise in its measurement. Further discussions on EKF can be found in Watanabe (1992) and Ray (1981). The basic idea of EKF is to linearize the nonlinearity by applying the first order Taylor series about the current state estimate.

Consider an autonomous nonlinear model of the discrete-time form

$$\mathbf{x}(t+t_s) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \omega(t), t) \quad \text{..... (8.12)}$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), \upsilon(t), t) \quad \text{..... (8.13)}$$

where **f** and **h** are C^{∞} vector fields on $\mathbf{R}^n$. In other words, the mapping **f** and **h** are smooth in their arguments, that is, all entries are real-valued functions of **x** with continuous partial derivatives of any order. **y**, **x** and **u** represent the deviation vectors with $\mathbf{y} \in \mathbf{R}^w$, $\mathbf{x} \in \mathbf{R}^n$, and $\mathbf{u} \in \mathbf{R}^m$, respectively. ω and υ are the zero-mean Gaussian white noises having covariance matrices **Q** and **R**.

By taking the Taylor expansion of (8.12) at $\mathbf{x}(t) = \hat{\mathbf{x}}(t|t), \omega(t) = 0$ and of (8.13) around $\mathbf{x}(t) = \hat{\mathbf{x}}(t|t-t_s), \upsilon(t) = 0$, where $\hat{\mathbf{x}}(t|t-t_s)$ denotes $\hat{\mathbf{x}}(t)$ given $\hat{\mathbf{x}}(t-t_s)$, (8.12) and (8.13) become

$$\mathbf{x}(t+t_s) = \mathbf{f}(\hat{\mathbf{x}}(t|t), \mathbf{u}(t), 0, t) + \mathbf{F}(t)(\mathbf{x}(t) - \hat{\mathbf{x}}(t|t)) + \mathbf{G}(t)\omega(t) \quad \text{..... (8.14)}$$

$$\mathbf{y}(t) = \mathbf{h}(\hat{\mathbf{x}}(t|t-t_s), \mathbf{u}(t), 0, t) + \mathbf{H}(t)(\mathbf{x}(t) - \hat{\mathbf{x}}(t|t-t_s)) + \mathbf{J}(t)\upsilon(t) \quad \text{..... (8.15)}$$

where

$$\mathbf{F}(t) = \left.\frac{\partial \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),\boldsymbol{\omega}(t),t)}{\partial \mathbf{x}(t)}\right|_{\mathbf{x}(t)=\hat{\mathbf{x}}(t|t),\,\boldsymbol{\omega}(t)=0}$$

$$\mathbf{G}(t) = \left.\frac{\partial \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),\boldsymbol{\omega}(t),t)}{\partial \boldsymbol{\omega}(t)}\right|_{\mathbf{x}(t)=\hat{\mathbf{x}}(t|t),\,\boldsymbol{\omega}(t)=0}$$

$$\mathbf{H}(t) = \left.\frac{\partial \mathbf{h}(\mathbf{x}(t),\mathbf{u}(t),\upsilon(t),t)}{\partial \mathbf{x}(t)}\right|_{\mathbf{x}(t)=\hat{\mathbf{x}}(t|t-t_s),\,\upsilon(t)=0}$$

$$\mathbf{J}(t) = \left.\frac{\partial \mathbf{h}(\mathbf{x}(t),\mathbf{u}(t),\upsilon(t),t)}{\partial \upsilon(t)}\right|_{\mathbf{x}(t)=\hat{\mathbf{x}}(t|t-t_s),\,\upsilon(t)=0}$$

Applying the Kalman filter algorithm (Watanabe, 1992) to (8.14) and (8.15), the algorithm for EKF is therefore:

1. Given $\hat{\mathbf{x}}(t)$, obtain $\hat{\mathbf{x}}(t+t_s)$ using the initial condition of $\hat{\mathbf{x}}(0|0)=\hat{\mathbf{x}}(0)$,

$$\hat{\mathbf{x}}(t+t_s|t) = \mathbf{f}(\hat{\mathbf{x}}(t|t),\mathbf{u}(t),0,t) \quad \text{..... (8.16)}$$

2. Obtain the filter error covariance matrix $\mathbf{P}(t+t_s|t)$ using the initial condition of $\mathbf{P}(0|0)=\mathbf{P}(0)$,

$$\mathbf{P}(t+t_s|t) = \mathbf{F}(t)\mathbf{P}(t|t)\mathbf{F}^{\mathrm{T}}(t)+\mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^{\mathrm{T}}(t) \quad \text{..... (8.17)}$$

3. Compute the filter gain $\mathbf{K}$ at time $t+t_s$,

$$\mathbf{K}(t+t_s)=\mathbf{P}(t+t_s|t)\mathbf{H}^{\mathrm{T}}(t+t_s)\begin{pmatrix}\mathbf{H}(t+t_s)\mathbf{P}(t+t_s|t)\mathbf{H}^{\mathrm{T}}(t+t_s)+\\ \mathbf{J}(t+t_s)\,\mathbf{R}(t+t_s)\,\mathbf{J}^{\mathrm{T}}(t+t_s)\end{pmatrix}^{-1} \quad \text{.... (8.18)}$$

4. Given the output $\mathbf{y}(t+t_s)$, determine the current state estimate at time $t+t_s$,

$$\hat{\mathbf{x}}(t+t_s|t+t_s)=\hat{\mathbf{x}}(t+t_s|t)+ \mathbf{K}(t+t_s)\left(\mathbf{y}(t+t_s)-\mathbf{h}\left(\hat{\mathbf{x}}(t+t_s|t),\mathbf{u}(t+t_s),0,t+t_s\right)\right) \quad \text{... (8.19)}$$

5. Update the filter error covariance matrix

$$\mathbf{P}(t+t_s|t+t_s) = (\mathbf{I} - \mathbf{K}(t+t_s)\,\mathbf{H}(t+t_s))\,\mathbf{P}(t+t_s|t) \quad \text{..... (8.20)}$$

8.5 SIMULATION RESULTS

The simulation example used is the single stage evaporation unit of the liquor burning process of Chapter 3. MAPLE V.3 was used to implement the simulation for both observer designs (GNF and EKF). The scenario investigated is the open loop response of a 2 m^3/hr increase in the product flow. The sampling time used in each case is 0.05 hour and the time span is 1 hour. All process variables are presented in deviation values. The same two outputs system described in Chapter 3 were considered. They are the inventory level and the density of the liquor discharge of the flash tank. The initial conditions of states were assumed to be zero in both GNF and EKF designs. The tracking performance of each filter is measured by the integral time-weighted absolute error (ITAE) performance index. In this case, the error is defined as stated in (8.8).

8.5.1 Determination of Actual Plant States

The determination of the actual plant states required the solution trajectory of the three state equations (3.1), (3.2) and (3.3). The three ordinary differential equations were solved using the numerical routine in MAPLE V.3 after the analytical and Laplace routines were unable to provide a solution because of the complex algebra involved. Gaussian white noise using the identity matrix **I** as the covariance matrix was added to the computed values of the states to represent the actual plant states. Therefore, the actual model with the noise is

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t)) + \begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.00001 & 0 \\ 0 & 0 & 0.001 \end{bmatrix} \omega(t) \\
\mathbf{y}(t) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0.001 & 0 \\ 0 & 0.00001 \end{bmatrix} \omega(t)
\end{aligned} \quad \text{..... (8.21)}$$

where $\mathbf{f}(\mathbf{x}(t),\mathbf{u}(t))$ represents (3.1), (3.2) and (3.3) in the standard form of (2.11).

8.5.2 Implementation of the Geometric Nonlinear Filter

The first step in implementing the GNF for the liquor burning process is to compute the observability matrix and hence the observability indices. By trial and error, the observability indices obtained were $l_1 = 2$ and $l_2 = 1$. Then the Jacobian matrix of the transformation is determined. Since MAPLE V.3 does not have the ability to solve the partial differential equations here, the alternative GNF design provided in section 8.3.3 was applied. The observer gain matrix used in the simulation was

$$\mathbf{K} = \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{..... (8.22)}$$

which gave the following characteristic polynomial with all three roots equal to -1.

$$P(s) = (1+2s+s^2)(1+s)$$

The final observer design equations are too long and complicated to be included in this thesis. Further details can be found in To and Tadé (1996e). However, it is important to note that these differential equations cannot be solved using the analytical or Laplace routine in Maple V.3 and using simple Euler method caused serious numerical errors in the results. Therefore, the more sophisticated numerical *dsolve* routine in MAPLE V.3 was used instead. The open loop response curves for each state are shown in *Figures 8.1*. The ITAE indices are shown in *Table 8.1*. These results are discussed below.

Table 8.1: ITAE for GNF and EKF

	GNF	EKF
Height	0.000750	0.000209
Density	0.351×10^{-5}	0.156×10^{-5}
Temperature	0.000395	0.000438

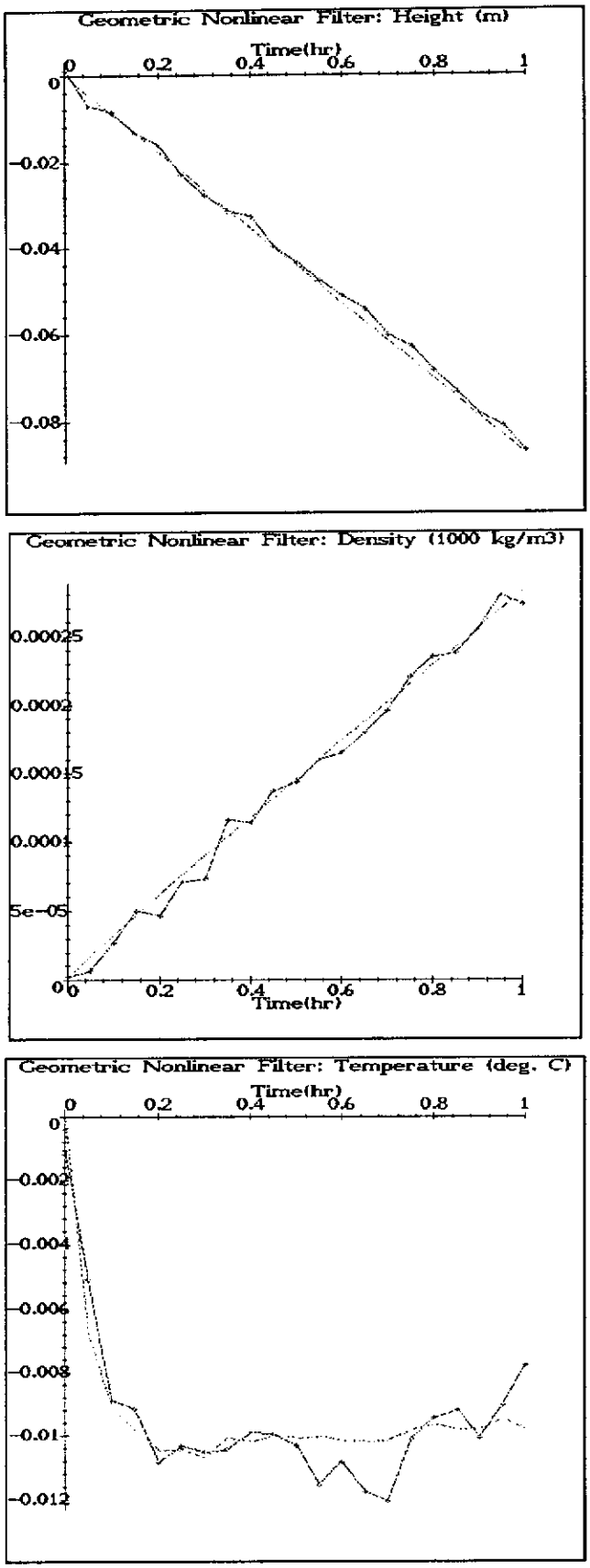


Figure 8.1: State Estimation of LBP using GNF (—+— Actual States - - - - - GNF)

8.5.3 Implementation of the Extended Kalman Filter

To implement the discrete time EKF, it was first necessary to convert the continuous time model equations (3.1) to (3.3) to discrete time form using the finite difference techniques such as the forward difference approximation (Seborg et al, 1989). That is,

$$\mathbf{x}(t + t_s) = \mathbf{x}(t) + \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))\, t_s$$

The reason for using this technique is its simplicity in implementation which is the same reason for choosing the discrete time EKF over the continuous time format. In addition, the discrete time EKF requires only simple matrix algebra to solve the discrete Ricatti differential equation on-line in order to determine the filter error covariance matrix at each sampling time. However, since the finite difference technique is used, the choice of sampling time is very important. This is discussed further in the next section.

The initial condition of the filter error covariance matrix $\mathbf{P}(0)$ was assumed to be $\mathbf{0}$. The simulation results and the ITAE indices are shown in *Figure 8.2* and *Table 8.1*, respectively.

8.6 Discussions

The reason for the use of 0.05 hour (180 seconds) as the sampling time for the simulation was that serious numerical round-off error was experienced in the state estimation of the liquor discharge temperature of the flash tank when the value of 0.1 hour (360 seconds) was employed. In the industry, the sampling time used is usually around 30 seconds so that numerical instability is less likely to occur.

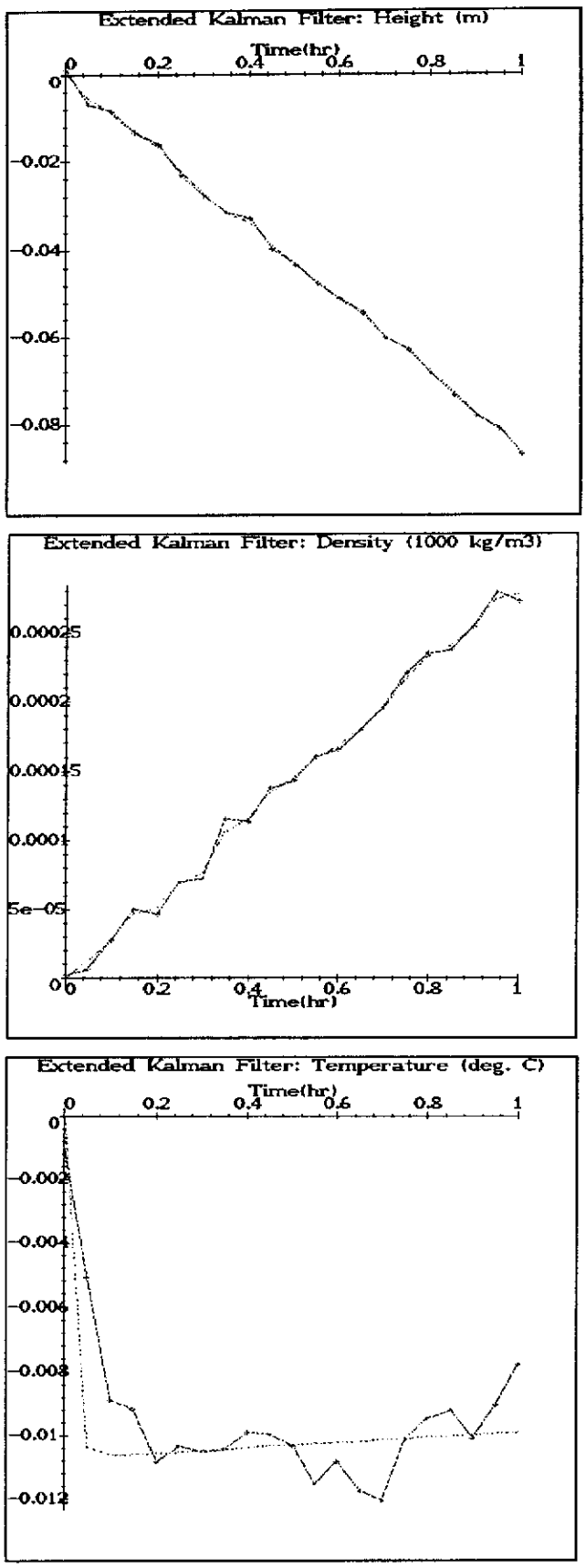


Figure 8.2: State Estimation of LBP using EKF (—+— Actual States - - - - - EKF)

The open loop response curves in *Figures 8.1* and *8.2* and the ITAE indices in *Table 8.1* show that the tracking performance of EKF is better than the GNF for the estimation of height and density. This is because the filter gain of EKF was continuously adjusted based on the information of the noise covariance matrices **Q** and **R** while the filter gain using GNF was constant. However, the tracking performance of the EKF for the temperature variable was poorer than that of GNF, as it is shown in the two figures and the performance indices in *Table 8.1* (0.000438 for EKF and 0.000395 for GNF). The degree of the nonlinearity in this case is far greater than the case for the height and the density. Hence, the results here showed that the GNF is much better than the EKF in coping with systems having high degree of nonlinearity. The performance of EKF increases with decreasing sampling time. This is entirely due to the forward difference numerical technique used to convert the continuous time model into discrete time form. It is recommended that a sampling time of less than 0.05 hour should be used to get closer to the industrial situation of about 30 seconds (0.00833 hour).

The computational time required to determine the states at each sampling time for the GNF is significantly longer than that for the EKF. The results here are opposite to the findings in literature, such as Bishop and Antoulas (1994) and Frezza et al (1988). This is because, in this study, the transformation could not be determined due to the limitation of MAPLE V.3 and the final state observer equations obtained using the alternative observer design approach given in section 8.3.3 were very complicated and needed to be solved numerically at each time step. Therefore, the computational time increased significantly. On the other hand, the discrete time EKF involved only straight-forward matrix algebraic manipulations.

The performances of both filters degraded (*Table 8.2*) when the magnitude of the covariance of the Gaussian white noise was increased by three times. The response curves are similar to *Figures 8.1* and *8.2*; hence, they are omitted to avoid repetitions. The tracking performance of both filters remained good and stable for observing the states, height and density. However, significant deteriorations in performance were

observed in the estimation of the temperature. The deterioration of GNF was considerable and can be explained by the constant filter gain used for the simulation. Performance improved significantly when the magnitude of the filter gain was increased. For example, when $k_{1,0}$ = 50, $k_{1,1}$ = 1200 and $k_{2,0}$ = 5, the corresponding ITAE values for height, density and temperature became 0.000697, 0.974×10^{-5} and 0.001180, respectively. However, instability resulted if the magnitude of the filter gain used was too high. The tracking performance of EKF for height and density was reasonably good because the filter gain was computed based on the new covariance matrix at each sampling time. However, the performance for the temperature remained poorer than that of GNF. This could be due to the high value of the sampling time used in the simulation. Improvement was observed when the sampling time was decreased to 0.01 hour (36 seconds). In fact, the performance of both filters depends significantly on the sampling time since a numerical routine was used for the simulation to solve the differential equations and the discrete time model was used for the EKF implementation.

Table 8.2: ITAE for GNF and EKF with Covariance Matrix = 3 I

	GNF	EKF
Height	0.00143	0.000625
Density	0.0000106	0.468×10^{-5}
Temperature	0.001186	0.00128

8.7 CONCLUSIONS

Two types of nonlinear filters, GNF and EKF, were investigated in this chapter to provide the state observers for nonlinear control strategies studied in the previous chapters. The case study here used the single effect evaporator of the liquor burning process as the target system. The open loop situation of a disturbance of 2 m^3/hr increase in the product flow was investigated. The tracking performances of both filters were good and stable. The GNF was slightly better than EKF in observing highly nonlinear system such as the temperature situation in this case. However, the GNF required significantly longer computational time than EKF because of the more

complicated observer design equations. This was mainly because MAPLE V.3 was unable to obtain the transformation for GNF. Otherwise, the computational time will be significantly shorter and more efficient than EKF (Bishop and Antoulas, 1994 and Frezza et al, 1988). When the covariance of the noise was increased, GNF was re-tuned and EKF required smaller sampling time to maintain the tracking performance. The performances of both filters are a function of the sampling time. In conclusion, both filters are comparable in performance, but the EKF is recommended as the state observer for the nonlinear control strategies studied based on the finding in this case study because of its efficiency and less algebra involved in its implementation. Further work is required to explore the use of GNF for highly nonlinear systems. This may require the use of other sophisticated and efficient software or a simplification of the algorithms for practical implementations.

CHAPTER NINE

PLANT IMPLEMENTATION OF INPUT OUTPUT LINEARIZATION ON THE LIQUOR BURNING PROCESS

9.1 INTRODUCTION

In the preceding chapters, we have concentrated on specific issues in designing a nonlinear control system for the simulated single effect evaporative stage of the liquor burning process in the Alcoa Alumina Refinery, Kwinana. The major emphasis has been on applying the input output linearization and the Su-Hunt-Meyer transformation to the simulation of the nonlinear process. This chapter is devoted to plant trials of implementing the input output linearization on the actual process. The trials were carried out in January 1996 even though it was originally scheduled for in November 1994. Earlier coding and debugging were done in July 1995. Other plant priorities and personnel commitments led to the postponement of the implementation trials till January 1996. We are therefore grateful that these trials took place at all. The I/O was chosen over SHM because the results in the previous chapters showed that the I/O was easier to implement, more robust, and provided better overall control performance in terms of the ITAE indices than SHM. The liquor burning process is a two-input two-output system, as described in Chapter 3. The I/O nonlinear controller was compared to the feedforward ratio linear control strategy (Configuration D, see section 3.1.2.3), which has currently replaced the linear SISO scheme (Configuration A) discussed in this thesis. According to plant personnel, the strategy using Configuration D gave a better performance than the former SISO scheme. It is hoped that this plant trial for input output linearization will stimulate further applications of

the nonlinear control theory for plant situations.

9.2 RESULTS AND DISCUSSIONS

The results obtained during the trial and the discussions on issues involved in implementing the nonlinear controller are given in the following sections. All states were measured on-line at a sampling time of 30 seconds. The control algorithm was coded in Honeywell Control Language (Honeywell, 1991) and the resident device is the customised block of the Application Module in the Local Control Network. The results shown in this chapter are in scaled values so as to preserve the plant proprietary information.

9.2.1 PI Action in the Nonlinear Controller and Industrial PI Controller

The nonlinear controller, as described in section 2.4.2.1, employs the MIMO Global Linearizing Control structure. The structure uses the proportional and integral action to stabilise the transformed linear system. The design parameters $\hat{\beta}_{ik}$ used were the same as those in the simulation. Therefore, substituting $\hat{\beta}_{10} = \hat{\beta}_{20} = 1$ into the PI control law (2.32), gives

$$v_i = y_i^{sp}(t) + K_{c_i}\left(y_i^{sp}(t) - y_i(t)\right) + \frac{K_{c_i}}{\tau_{I_i}}\int_0^t \left(y_i^{sp}(t) - y_i(t)\right) dt \quad \text{..... (9.1)}$$

$$(i = 1, 2)$$

The implication of the above PI control law requires the output of the PI controller as the output variables, height and density, instead of the input variables, flowrate of feed and cooling water. The industrial PI controller is coded in the velocity form and returns only the actual values of the process input variables (*See Figure 9.1*). Therefore, the following solution (*Figure 9.2*) was proposed to form the link between the industrial PI controller and the nonlinear PI controlling action (9.1).

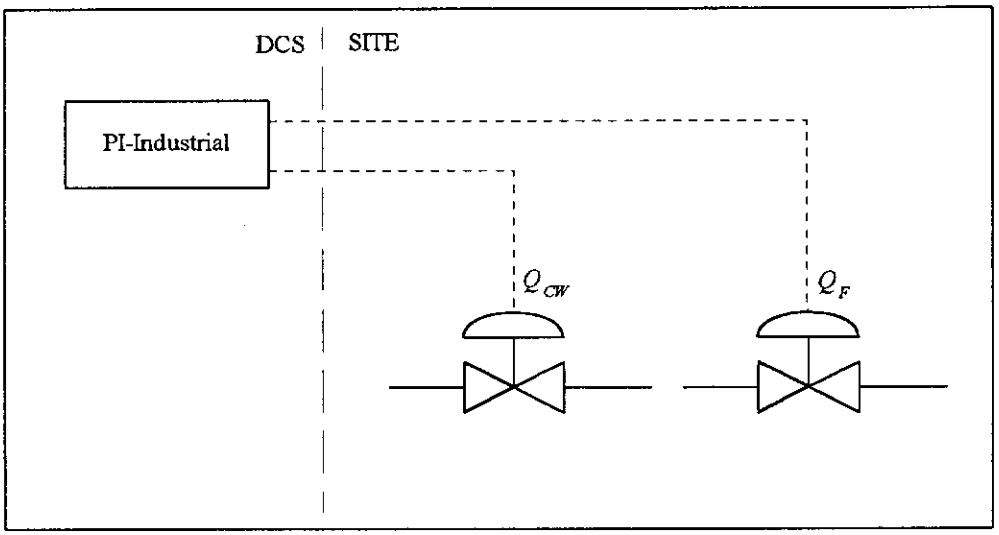


Figure 9.1: PI-Industrial Controller Actions

The flowrate of the feed and the cooling water returned by the industrial PI controller were converted to equivalent level and density values by the following equations.

$$\mathrm{PI}_{\mathrm{GLC}}(h) = 2.21 + \left(\frac{\mathrm{PI}_{\mathrm{Industrial}}(Q_F)}{50}\right) \quad \text{..... (9.2)}$$

$$\mathrm{PI}_{\mathrm{GLC}}(\rho) = 1.20 + 0.7\left(\frac{\mathrm{PI}_{\mathrm{Industrial}}(Q_{CW})}{389.966}\right) \quad \text{..... (9.3)}$$

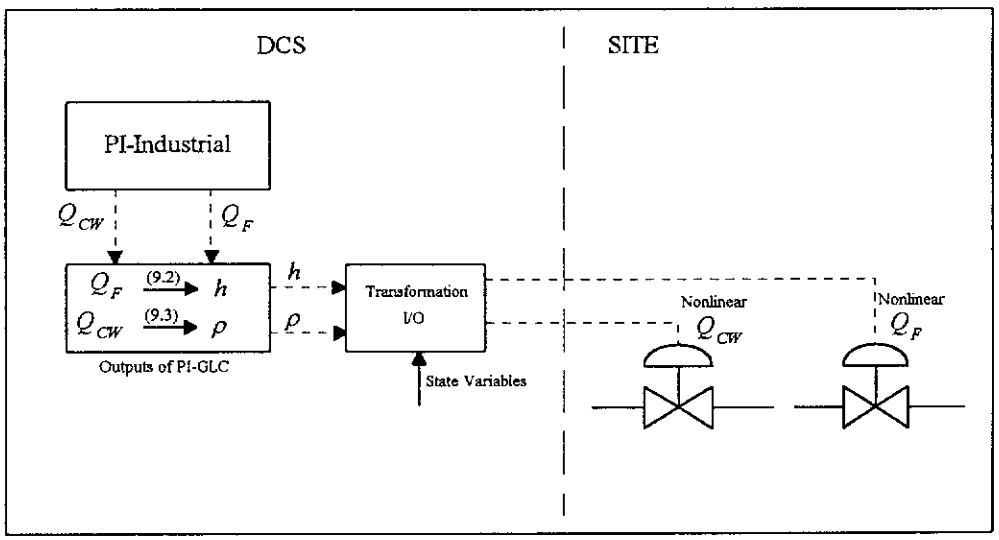


Figure 9.2: Conversion of PI-Industrial to PI-GLC

The above linear relationships simply scale the output of the industrial PI controller to obtain the necessary conversion and are constructed around the following steady state values and ranges of the height and density, respectively.

Table 9.1: Steady State Values and Ranges of Height and Density

	Steady State	Range
Height	2.71 m at 25 m^3/hour Feed Flow	2.21 to 3.21 m
Density	1.762 tonne/m^3 or 1762 kg/m^3 at 313.087 m^3/hour Cooling Water Flow	1.2 to 1.9 tonne/m^3

The proposed conversion assumed that linear relationships exist between the level and the feed flow, and the density and the cooling water flow. This assumption may actually limit the performance of the nonlinear controller. However, this scaling is essential and is the most direct and practical way of linking the two PI controllers. The assumption was justified by the successful results of the trial given in later sections.

9.2.2 Anti-reset Windup for PI Controller

Reset windup is an inherent disadvantage of the integral control action in the PI controller. As a result of the unusual arrangement of the PI control action in the GLC, constraints on the process inputs cannot be directly imposed on the integral action of this nonlinear PI controller (9.1). To prevent reset windup from occurring in the industrial PI controller, we proposed the following solution.

When the output of the nonlinear controller, feed or cooling water flowrate, reaches the upper limit, any further increase in the output of the corresponding industrial PI controller is set to zero automatically. The same logic is applied to the lower limit of each process input variable. In other words, the output of the industrial PI is implicitly constrained by imposing limits on the output of the nonlinear controller.

Apart from the upper and lower limits constraints on the outputs of the nonlinear controller, there are other constraints imposed on the values to ensure that the flows

do not disturb the process too much. The maximum changes allowed for the feed and cooling water flows are 5 and 50 m^3/hour, respectively.

9.2.3 Initialisation of PI Controller

Since the industrial PI controller is implemented in the velocity form, it is therefore necessary to initialise the controller when the nonlinear controller is switched on. To initialise the PI controller, we require the inverse of the state feedback law. All current states and process inputs were fetched, validated and stored. Using the inverse of the state feedback law, the transformed linear inputs v_1 or $\mathrm{PI}_{\mathrm{GLC}}(h)$ and v_2 or $\mathrm{PI}_{\mathrm{GLC}}(\rho)$ were determined. These values were then corrected to the equivalent flow states, that is, $\mathrm{PI}_{\mathrm{Industrial}}(Q_F)$ and $\mathrm{PI}_{\mathrm{Industrial}}(Q_{CW})$, of the industrial PI controller using the inverse of (9.2) and (9.3). Hence, the current values of the inputs (level and density) and the outputs (equivalent feed and cooling water flow) required to commission the industrial PI controller were determined.

9.2.4 Tuning of PI Controller

The tuning of the PI controller was performed by trial and error. First, the error used in the PI block was defined as Error = Process Value - Set Point. The results are shown in *Figure 9.3*. For the level control, the tuning parameters used were -50 %/% for the proportional gain and 30 minutes for the integral time. These parameters were kept constant during the test. The proportional gain and the integral time for the density control started with the values of -2000 %/% and 20 minutes, respectively. However, as shown in *Figure 9.3*, oscillations were observed in all process variable. This implies that the tuning parameters were too high and the control actions were too severe. Therefore, the proportional gain and the integral time for the density control were changed to -1000 %/% and 50 minutes at 13:30. It is clearly shown in the figure that the oscillations died off immediately and all variables remained stable despite the presence of process noise. These values of tuning parameters were used for all the subsequent tests.

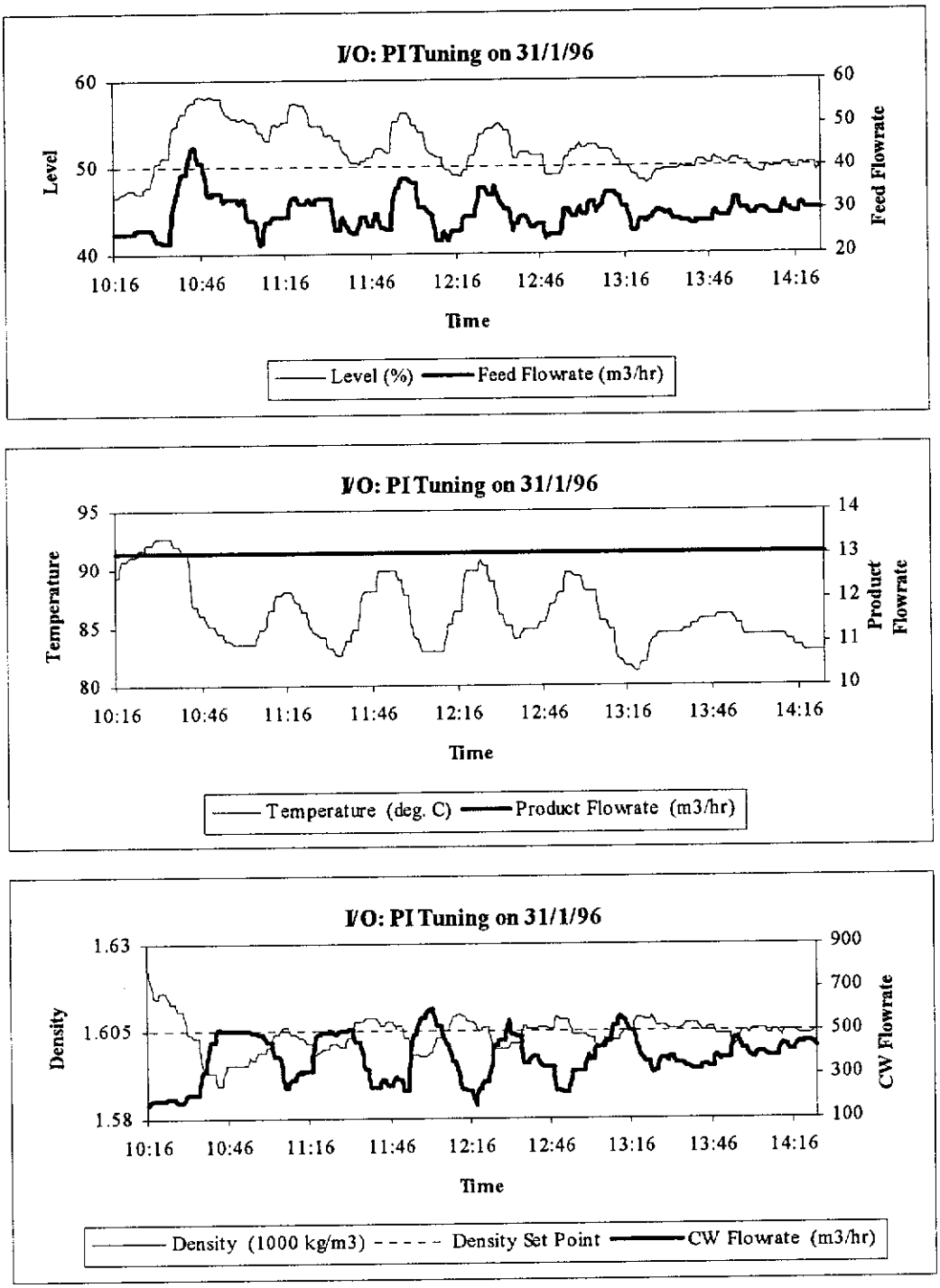


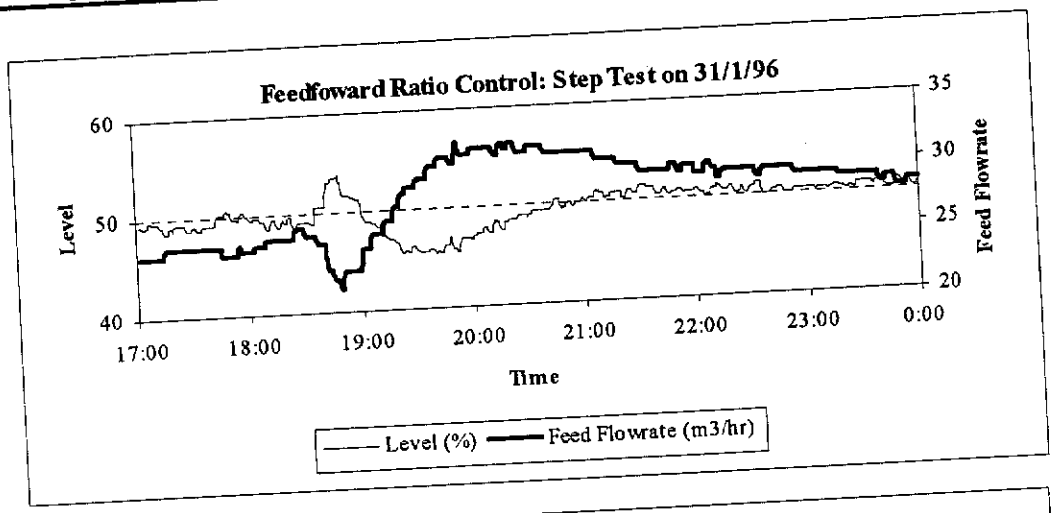
Figure 9.3: PI Controller Tuning for I/O Implementation

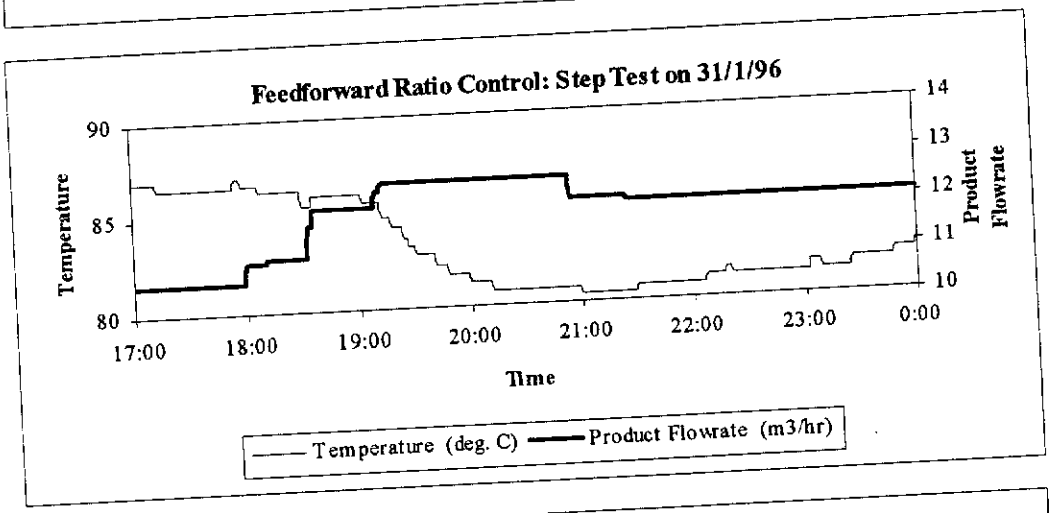
9.2.5 Nonlinear Controller Implementation Tests

Two tests were conducted to examine the performance of the nonlinear controller. They were a flow change in the product stream and a change in the density set point. Similar tests were performed on the existing feedforward ratio controller to provide a bench mark or a basis for comparison. Details and discussions of each test are provided in the following sections.

9.2.5.1 Step Change in Product Flowrate

The step change in the product flowrate for the feed forward ratio and the nonlinear controller is shown in *Figures 9.4* and *9.5*, respectively. For the nonlinear controller (*Figure 9.5*), the approximately 2 m^3/hour decrease in product flowrate had no effect on the density and the level which remained at their set points. However, in the case of the feedforward ratio controller (*Figure 9.4*), the 2 m^3/hour increase in flow led to small disturbances in the density and level response curves which settled to their corresponding set points after a period of approximately 2 hours. In *Figure 9.4*, a decrease of 0.5 m^3/hour in the product flow was also introduced into the process at 21:00 hours. The level controlled by the feed flow was not affected by this disturbance, as in the nonlinear case. Despite the relatively small decrease in flow, the density was greatly disturbed and took more than 3 hours to settle. This observation illustrates that the nonlinear controller performed better than the feedforward ratio controller in the presence of a decrease in the product flow. Since the feedforward ratio controller has no difficulty in coping with any increase in product flow, similar or even better performance can be expected for the nonlinear controller. Therefore, no test was necessary for this scenario.





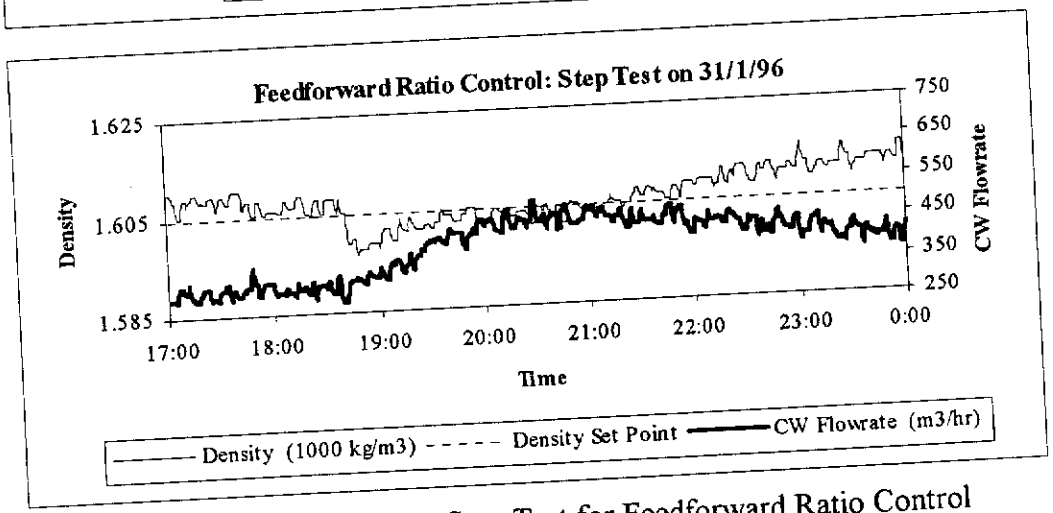
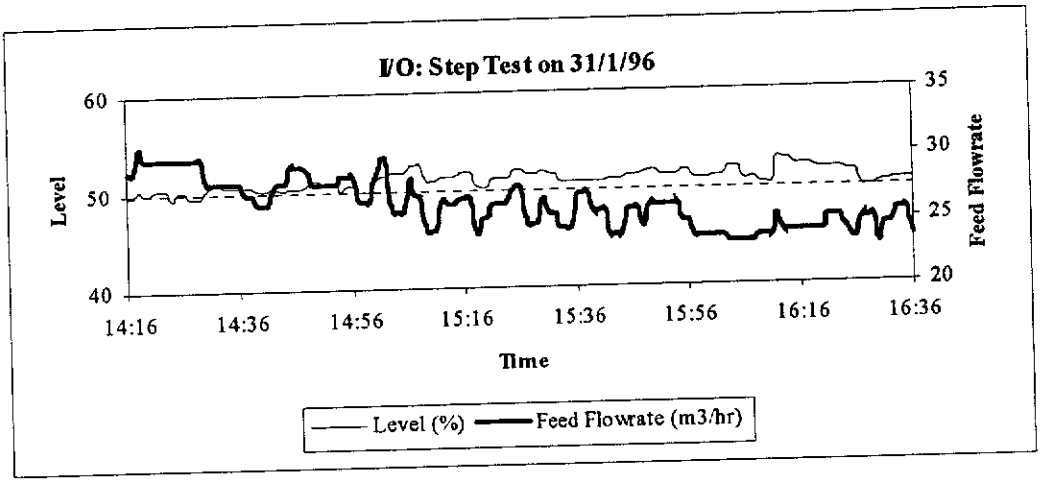
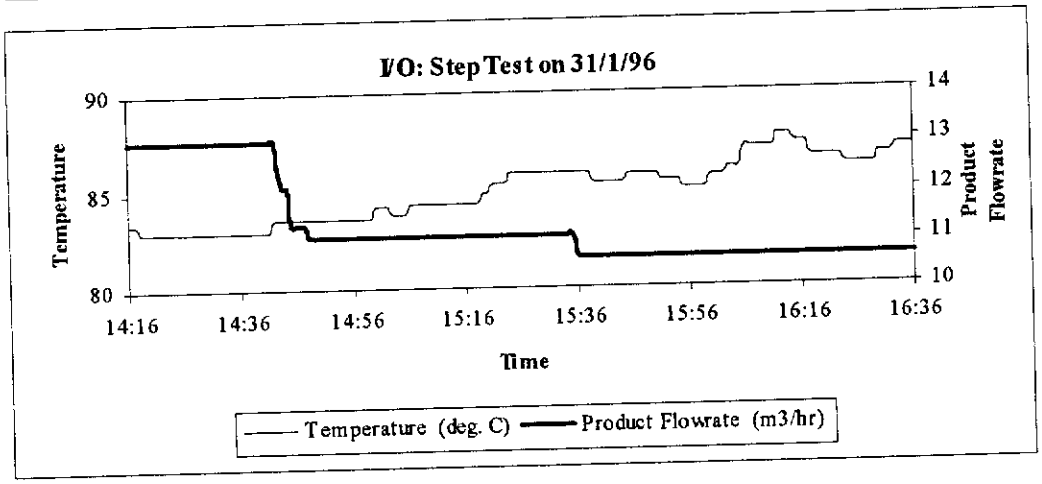


Figure 9.4: Product Flow Step Test for Feedforward Ratio Control





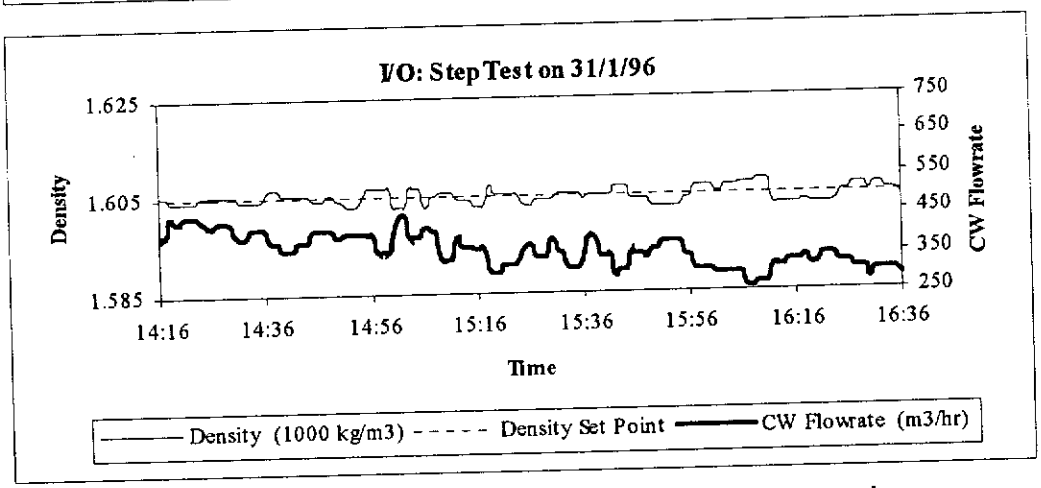


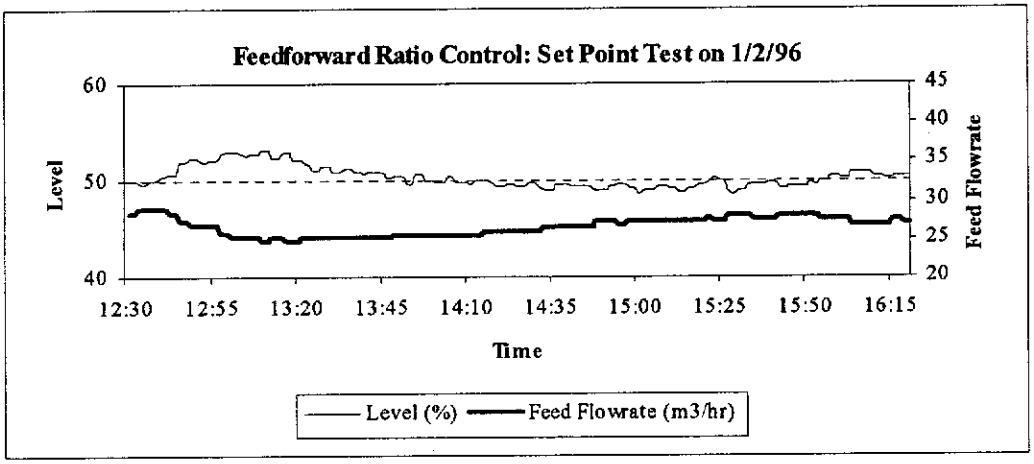
Figure 9.5: Product Flow Step Test for I/O Implementation

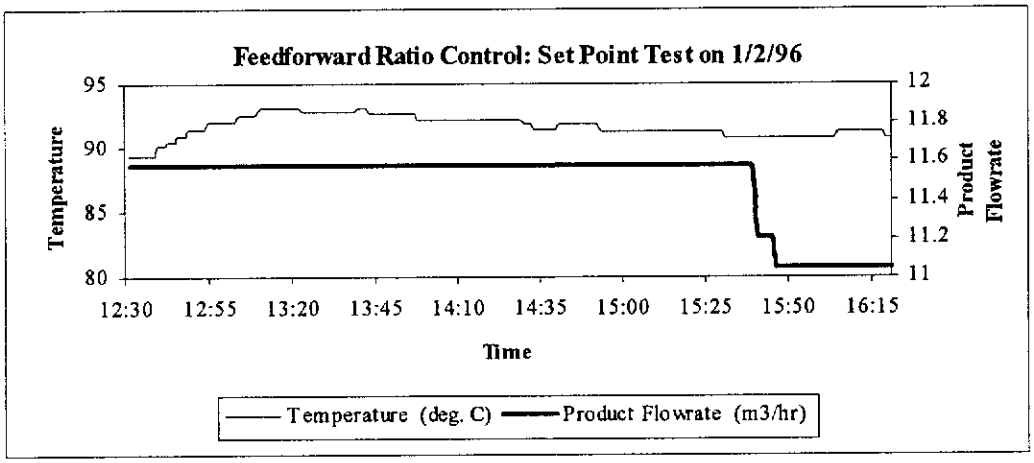
9.2.5.2 Set Point Change in Density Control

A set point test was performed on both controllers. The set point was first dropped from 1.605 to 1.600 tonne/m^3 and then returned to 1.605 tonne/m^3. The results are shown in *Figures 9.6* and *9.7*. For I/O, the test began at 9:00 (*Figure 9.5*). The decrease in set point led to slight disturbances in both level and density controls and the settling time was approximately 1 hour. When the set point was increased back to 1.605 tonne/m^3 at 10:30 hours, larger disruptions were observed and the settling times in this scenario were slightly more than 1 hour for both controls.

For the case of the feedforward ratio controller, similar responses to the nonlinear case were observed when the set point was decreased to 1.600 tonne/m^3 at 12:40 hours (*Figure 9.6*). Only slight disturbances to the level and density control were caused by this change. The settling time for both control was slightly more than 1 hour. However, when the set point was increased back to 1.605 tonne/m^3 at 2:15 hours, the feedforward ratio controller for the density was very sluggish. On the other hand, the level control was unaffected. An unexpected drop of 0.5 m^3/hour in the product flow caused by the downstream operation is observed in *Figure 9.6*. Based on the results in the previous section, this flow cut should lead to a rise in the liquor density; hence, a shorter settling time. In spite of this, the density remained below the set point for at least 2 hours. The final settling time was about 3 hours.

The foregone demonstrates that the nonlinear controller is more aggressive and responds to disturbances faster than the feedforward ratio controller.





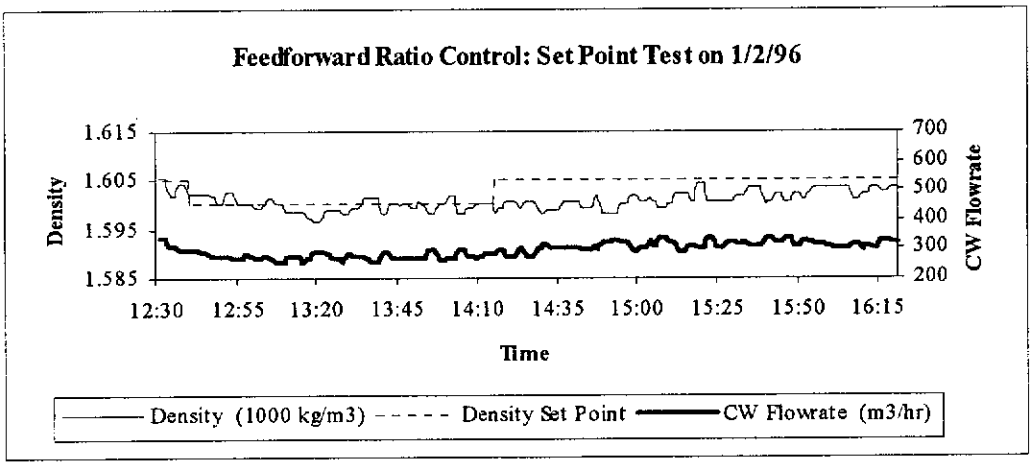


Figure 9.6: Density Set Point Test for Feedforward Ratio Control

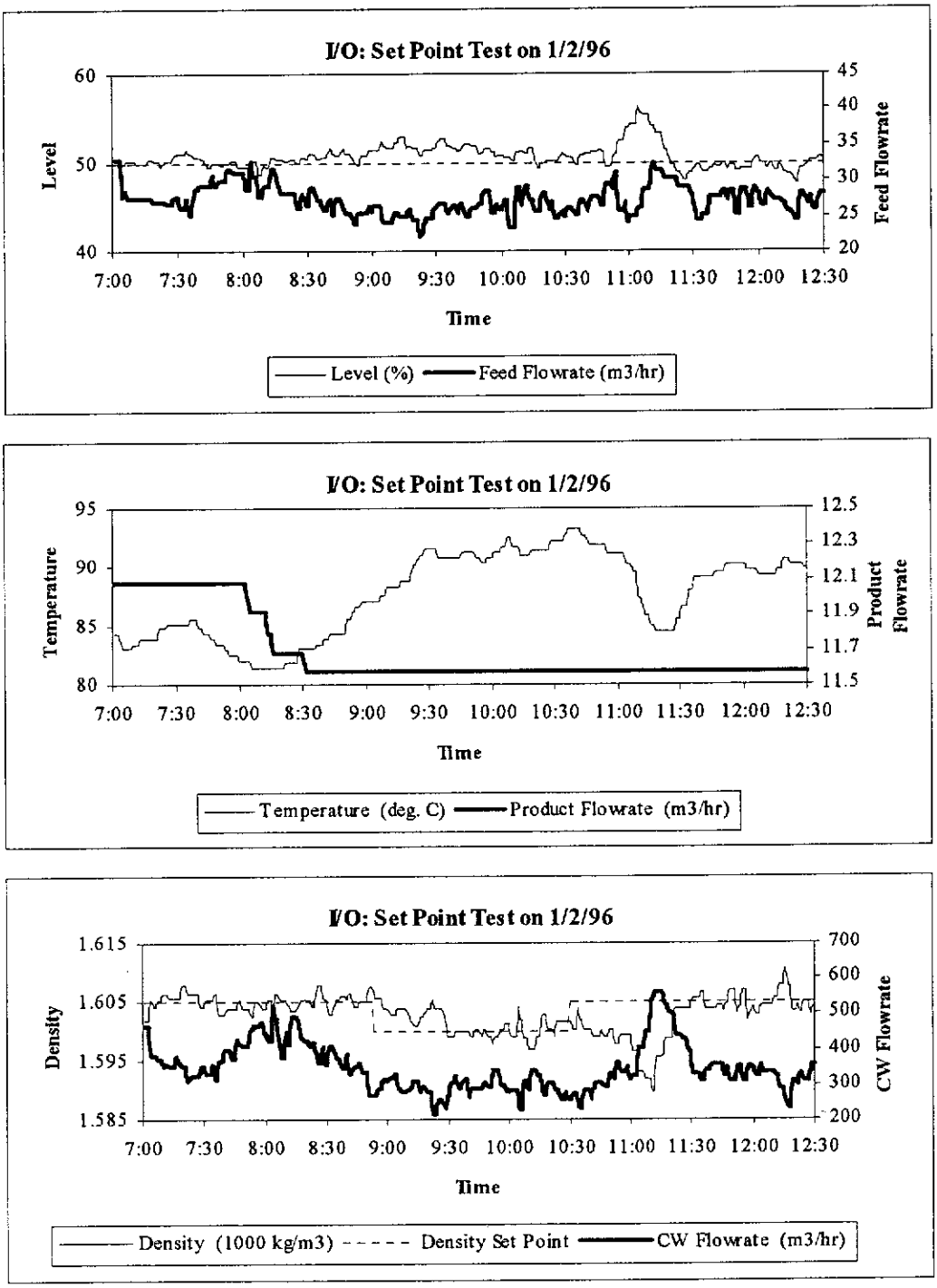


Figure 9.7: Density Set Point Test for I/O Implementation

9.2.5.3 Discussion

The series of tests showed that the product flow is a major disturbance in the process. The nonlinear controller performed well in the presence of a disturbance and a set point change. The comparison here might not be very appropriate because of the changing process conditions, especially the product flow. However, the feedforward ratio controller provided a reasonable reference with which the performance of the nonlinear controller can be gauged. It is possible to quantify the performances obtained during the test using ITAE. However, the results would not be meaningful as a result of the changing process conditions during tests. Therefore only the settling time was used as a quantifying tool here.

Another comment is that the nonlinear controller appeared to be too aggressive. It is obvious that the nonlinear controller used a lot more control energy than the feedforward one. This might increase the operating cost of the process. To curtail the severity of this problem, the tuning parameters can be further reduced. Therefore, relatively smoother control actions can be established with less control energy used. Moreover, the level and density can be kept even closer to their set points. Even though the control action using the nonlinear controller is fairly aggressive as mentioned before, it is important to note that no process variable has ever reached the safety bracket of the plant unit.

In the nonlinear case, a very high level of noise was present in the response curves which could be due to the aggressive behaviour of the nonlinear controller. This situation is similar to those where a very high proportional gain was used in the PI linear control loop. The noise level can be reduced by using filters or by reducing the magnitudes of the PI tuning parameters. Compared with the feedforward ratio control, the increase in CPU loading required to run the nonlinear control codes in the Application Module is negligible because the coding involved only basic algebraic calculations.

The success of the trial here showed that the dynamic model developed in Chapter 3 for this process was valid even though it was not critically checked and validated. In other words, the model provides a suitable representation of the actual process and the assumptions are justified.

9.3 CONCLUSIONS

The MIMO Global Linearizing Control using the input output linearization was successfully implemented on the single effect evaporative stage of the liquor burning process. The major problem encountered in the trial was the difference between the industrial PI controller which has its output as the process input and the nonlinear PI action used in GLC which has its output as the process output. A simple linear relationship was proposed to convert the output of the industrial PI to the equivalent output value of the nonlinear PI. Besides converting the values, the industrial PI controller must be properly initialised using the inverse of the linear relationship and of the state feedback law when switching to the nonlinear controller. Moreover, the output of industrial PI controller was also indirectly constrained to further increase or decrease when the output of the nonlinear controller reaches the upper or lower limit of the process input, respectively. This prevents reset windup of the integral action.

The tuning of the nonlinear controller was done by trial and error. The results showed that the performance of the nonlinear controller is more aggressive and responsive than the existing feedforward ratio controller. At the same time, the assumptions and the dynamic model used to construct the nonlinear controller were indirectly shown to be appropriate.

In conclusion, this chapter covered not only the practicability and superiority of the nonlinear controller using input output linearization, but it has also established a link between academia and industry. It is hoped that further tests can be done in future to implement more of the strategies proposed in this thesis and, in particular, further work can be done to assess the economic benefits of the nonlinear techniques compared to the existing control of this process unit.

CHAPTER TEN

CONCLUSIONS AND RECOMMENDATIONS

10.1 CONCLUSIONS

In this research, two types of nonlinear control theories, namely, input output and input state linearization (Su-Hunt-Meyer transformation), were studied using the liquor burning unit associated with the Bayer process for alumina processing as the target process. Two designs of the liquor burning unit were investigated: a single effect and a triple effects evaporator design. One of the aims of the thesis was to demonstrate the superiority and simplicity of the nonlinear control techniques using computer simulations and plant implementations. The simulation was implemented using MAPLE V.3 as the computing environment and the results were compared with SISO linear control scheme with proportional and integral action.

The algorithm of the input output linearization was straight forward and can be directly applied to the target process. On the other hand, the Su-Hunt-Meyer transformation was too complicated and difficult to be handled by MAPLE V.3 and a simplified solution algorithm was proposed to obtain this nonlinear transformation. The simplified algorithm was successfully implemented in the simulated single effect evaporation unit.

Only the input output linearization was implemented on the simulation of the nonlinear control of the triple effect evaporation unit. Since the dynamic model is input nonlinear, the Su-Hunt-Meyer transformation which is only applicable to input linear systems could not be used. Moreover, the computation of the transformation would be too difficult for MAPLE V.3 to handle. An algorithm was developed to extend the input output linearization to solve a general class of nonlinear MIMO process. This algorithm was successfully implemented on the triple effect evaporator.

Robustness issue was also addressed in this thesis since the nonlinear control strategy

is a model-based technique. A procedure called uncertainty vector adjustment was developed to improve the robustness of each of the nonlinear control theory. The procedure provides an appropriate adjustment to the nonlinear transformation according to the overall effect of modelling errors (structural or parametric) on the time derivative of the actual plant states measured $\dot{\mathbf{x}}_p$. This adjustment or additional control action keeps $\dot{\mathbf{x}}_p$ at zero at all time. Consequently, the corrupted control system with the uncertainty vector adjustment settles and converges faster than the one without the UVA. The effectiveness of the UVA was illustrated on the simulated single effect and the triple effects evaporation units. Using Lyapunov function analysis, the stability of the UVA was theoretically established and the bounds of the uncertainty vector and modelling errors within which the controlled system remains asymptotically stable were also determined.

Geometric nonlinear control requires full state feedback for the computation of the nonlinear transformation. This condition is not always satisfied in an industrial setting. Therefore, a brief discussion on state observers is included in this thesis. The state observers studied were geometric nonlinear and extended Kalman filters. The basic idea of the geometric nonlinear filter is similar to the nonlinear control theory. The nonlinear system is transformed into a state observer canonical form through a nonlinear transformation. The simulation of the open loop dynamics of the single effect evaporator showed that the geometric nonlinear filter can cope with nonlinearity better than the extended Kalman filter. However, in this study, the computational time required by the geometric nonlinear filter is much longer than that for the extended Kalman filter. This is because of the limitation of MAPLE V.3 to determine the transformation. As a result, it was concluded that the extended Kalman filter is more suitable as the state observer for implementing the nonlinear controller because of its comparable performance and less algebraic manipulations.

An integrated MAPLE package was developed for designing and simulating the closed loop dynamics of systems using the nonlinear control theories studied in this thesis. The package automates the computing algorithms for input output linearization and

Su-Hunt-Meyer transformation. The package has the ability to provide graphical representation of the closed loop responses of a nonlinear control system. The usefulness and limitations of MAPLE were also addressed. It was concluded that MAPLE is one of the most suitable tools available for investigating nonlinear control systems. Moreover, there is no doubt that recent developments in this area of symbolic computation is imperative to the design and analysis of control algorithms.

The practical value and feasibility of the nonlinear control theory were investigated by implementing the input output linearization in an industrial environment. The results showed that the nonlinear controller performed better than the feedforward ratio controller, which is currently used on site, in terms of fast settling time, aggressiveness and resistance to disturbances. Besides demonstrating the superiority and the simplicity of implementing the nonlinear controller, this exercise has successfully bridged the gap between academia and industry since substantial interaction was necessary.

10.2 RECOMMENDATIONS

The research in this context has successfully taken up academic and industrial challenge and has also raised many questions which need to be discussed further. Therefore, the following recommendations are stated for future research in the nonlinear control area.

Developing a reliable, interactive and easy-to-implement model is the first step of designing a nonlinear control system. This local description of the process can be identified using the first principles of chemistry and physics (analytical model) or statistical system identification techniques (systemic model). The nonlinear control theory so far has only been designed based on the analytical model. However, it is not always possible to establish an analytical model for a chemical process. Therefore, a systemic model is required. In order to use the nonlinear control theory, the resulting nonlinear polynomial difference equation representation of the process model has to be converted to the form of a nonlinear state space realisation, for example,

NARMAX or Volterra series model. The higher order terms in this realisation represent process uncertainties. An investigation should be performed to develop a conversion algorithm to generate the NARMAX or Volterra series model. Wether the model is analytical or systemic, physical insights and relevant assumptions are essential in developing an adequate, yet not overly simplified model.

A lot of research has been conducted in the area of general nonlinear systems with, for example, non-minimum phase behaviour, high dimensionality and significant time delay. However, they are often very complicated and too mathematically abstract. Therefore, as in this thesis, more efforts should be focused at establishing links between the academic theory and industrial practice, in order to simplify all these abstract theories or at least, isolate special cases so that the theory can be more viable and "user friendly".

The plant trial of the input output linearization was very successful in this case study. However, more tests are necessary in order to actually implement the nonlinear controller in the refinery. Not only that the controller has to be proven performance wise, but it also has to be justified in the economic sense.

The uncertainty vector adjustment developed in this thesis appears to be very practical and easy-to-implement. Further investigations should be performed to explore the ability and limitation of the procedure. The adjustment should be tested with different types of nonlinear control theory and chemical systems.

The robustness of the geometric nonlinear and extended Kalman filters has not been studied in this thesis due to time limitations. It is recommended that the stability and performance of each filter should be investigated in the presence of modelling errors. The possibility of applying UVA to the state observer theory, especially the geometric nonlinear filter, should also be researched. Theoretically, the procedure will provide adjustments to the nonlinear transformation of the nonlinear filter. Hence, improved tracking performance can be achieved in the presence of modelling errors.

REFERENCES

Akhrif, O. and Blankenship, G. L. (1988) Computer Algebra Algorithms for Nonlinear Control. In: *Advanced Computing Concepts and Techniques in Control Engineering* (M. J. Deham and A. J. Laub, Eds), Springer-Verlag, Berlin, 53-80.

Alsop, A. W. (1987) Nonlinear Chemical Process Control by means of Linearizing State and Input Variable Transformations. *PhD Thesis*, University of Texas, Austin, U. S. A. .

Alvarez, J. and González, E. (1989) Global Nonlinear Control of a Continuous Stirred Tank Reactor. *Chemical Engineering Science*, **44**(5), 1147-1160.

Arkun, Y. and Calvet J.-P. (1992) Robust Stabilization of Input/Output Linearizable Systems under Uncertainty and Disturbances. *AIChE Journal*, **38**(8), 1145-1158.

Barolo, M. (1994) On the Equivalence between The GMC and The GLC Controllers. *Computers. Chem. Engng.*, **18**(8), 769-772.

Bequette, B. W. (1991) Nonlinear Control of Chemical Processes: A Review. *Ind. Eng. Chem. Res.* ,**30**(7), 1391-1413.

Bestle, D. and Zeitz, M. (1983) Cononical Form Observer Design for Non-linear Time-variable Systems. *Int. Journal of Control*, **38**(2), 419-431.

Billet, R. (1989) *Evaporation Technology: Principles, Applications, Economics.* VCH, Weinheim, Germany.

Billings, S. A., Gray J. O. and Owens, D. H. (1984) *Nonlinear System Design.* Peter Peregrinus Ltd., London, UK.

Bishop, R. H. and Antoulas, A. C. (1994) Nonlinear Approach to Aircraft Tracking Problem. *Journal of Guidance, Control, and Dynamics*, **17**(5), September-October, 1124-1130.

Boothby, W. M. (1984) Some Comments on Global Linearization of Nonlinear Systems. *Systems & Control Letters*, **4**, 143-147.

Brockett, R W. (1966) The Status of Stability Theory for Deterministic Systems. *IEEE Transactions on Automatic Control*, **AC-11**(3), 596-606.

Brockett, R. W. (1978) Feedback invariants for nonlinear systems. *IFAC Congress Proc. VII*, Helsinki.

Brunovsky, P. (1970) A Classification of Linear Controllable Systems. *Kibernetika*, Praha, **6**, 173-188.

Burdett, J. W. and Holland, C. D. (1971) Dynamics of a Multiple-Effect Evaporator System. *AIChE Journal*, **17**(5), 1080-1089.

Calvet, J.-P. (1989) A Differential Geometric Approach for the Nominal and Robust Control of Nonlinear Chemical Processes. *Ph. D. Thesis*, Georgia Institute of Technology.

Char, B. W., Geddes, K. O., Monagan, M. B. and Watt, S. M. (1988) *MAPLE Reference Manual* (5th ed.). WATCOM, Waterloo, Canada.

Char, B.W., Geddes, K. O., Gonnet, G. H., Leong, B. L., Monagan, M. B. and Watt, S. M. (1991) *MAPLE V Library Reference Manual*. Waterloo MAPLE Publishing, Springer-Verlag, New York.

Char, B.W., Geddes, K. O., Gonnet, G. H., Leong, B. L., Monagan, M. B. and Watt, S. M. (1992) *First Leaves: A Tutorial Introduction to MAPLE V*. Waterloo MAPLE Publishing, Springer-Verlag, New York.

Chen, C. T. (1970) *Introduction to Linear System Theory*. Holt, Rinehart and Winston, New York.

Chen, Y. H. (1988) Design of Robust Controllers for Uncertain Dynamical Systems. *IEEE Transactions on Automatic control*, **33**, 487-491.

Chen, Y. H. and Leitmann, G. (1987) Robustness of Uncertain Systems in the Absence of Matching Assumptions. *Int. J. Control*, **45**(5), 1527-1542.

Clarke, D. W. and Mohtadi, C. and Tuffs, P. S. (1987a) Generalised Predictive Control - Part I. The Basic Algorithm. *Automatica*, **23**(2), 137-148.

Clarke, D. W. and Mohtadi, C. and Tuffs, P. S. (1987b) Generalised Predictive Control - Part II. Extensions and Interpretations. *Automatica*, **23**(2), 149-160.

Comito, V. (1995) *Chemical Engineering Project 491/492: Nonlinear Theory Using MAPLE.* School of Chemical Engineering, Curtin University of Technology, Perth, Australia.

Corless, M. and Leitmann, G. (1981) Continuous State Feedback Guaranteeing Uniform Ultimate Boundedness for Uncertain Dynamics Systems. *IEEE Transactions on Automatic Control*, **AC-26**, 1139-1144.

Daoutidis, P. and Kravaris, C. (1989) Synthesis of Feedforward/State Feedback Controllers for Nonlinear Processes. *AIChE Journal*, **35**(10), 1602-1616.

Daoutidis, P. and Kravaris, C. (1994) Dynamic Output Feedback Control of Mimimum-Phase Multivariable Nonlinear Processes. *AIChE Journal*, **49**(4), 433-447.

Daoutidis, P., Soroush, M. and Kravaris, C. (1990) Feedforward/Feedback Control of Multivariable Nonlinear Processes. *AIChE Journal*, **36**(10), 1471-1484.

Driscoll, R. H., Ng, S. and Chusprasert, S. (1995) Evaporator Process Control using Computer Models. *Food Australia*, 47 (1), 27-31.

Ellis, W. Jr., Johnson, E. W. and Schwalbe, D. (1992) *MAPLE V Flight Manual: Tutorials for Calculus, Linear Algebra, and Differential Equations.* Brook/Cole Publishing Company, Pacific Grove, California.

Essen, H. van (1992) *Symbolic Speak Louder Than Numbers: Analysis and Design of Nonlinear Control Systems with the Symbolic Computation System MAPLE.* Rep. WFW 92.061, Eindhoven Univ. Tech., Faculty of Mechnical Engineering, June.

Essen, H. van and Jager, B. de (1993) Analysis and Design of Nonlinear Control Systems with the Symbolic Computation System MAPLE. In: *Proc. 2nd European Contr. Conf.*, Groningen, the Netherlands, vol. 4, June, 2081-2085.

Figueiredo, Rui J. P., de and Chen, Guanrong (1993) *Nonlinear Feedback Control Systems : An Operator Theory Approach*. Academic Press, Inc., New York.

Frezza, R., Karahan, S., Krener, A. J. and Hubbard, M. (1988) Application of An Efficient Nonlinear Filter. In: *Analysis and Control of Nonlinear Systems*, (C. I. Byrnes, C. F. Martin and R. E. Saeks, Eds.), Elsevier Science Pubblishers B. V., North-Holland, 223-237.

Gerald, C. F. and Wheatley, P. O. (1989) *Applied Numerical Analysis* (4th ed.). Addition-Wesley Publishing Company, U. S. A., Chapter 5.

Gilbert, E.G. and Ha, I. J. (1984) An Approach to Nonlinear Feedback Control with Applications to Robotics. *IEEE Transactions on Syst. Man Cybern.*, **SMC-14**, 879-884.

Glynn, J. (1989) *Exploring Math from Algebra to Calculus with Derive, A Mathematical Assistant*. MathWare.

Graham, D. and McRuer, D. (1961) *Analysis of Nonlinear Control Systems*. Dover Publications, Inc., New York.

Gutman, S. (1979) Uncertain Dynamic Systems - A Lyapunov Min-Max Approach. *IEEE Transactions on Automatic Control* , 24, 437-443.

Ha, I.J. (1989) New Matching Conditions for Output Regulation of a Class of Uncertain Nonlinear Systems. *IEEE Transactions on Automatic Control*, **34**(1), 116-118.

Hahn, W. (1967) *Stability of Motion*. Springer-Verlag, Berlin, 166-168.

Haper, D., Wooff, C. D. and Hodgkinson, D. E. (1991) *A Guide to Computer Algebra Systems*. John Wiley and Sons, England.

Heck, A. (1993) *Introduction to MAPLE*. Springer-Verlag, New York.

Henson, M. A. and Seborg, D. E. (1990) Input-Output Linearization of General Nonlinear Processes. *AIChE Journal*, **36**(11), 1753-1757.

Henson, M. A. (1992) Feedback Linearization Strategies for Nonlinear Process Control. *PhD Thesis*, University of California, Santa Barbara.

Hermann, R. (1984) Perturbation and Linearization of Nonlinear Control Systems. *Workshop on Nonlinear Problems in Control and Fluid Dynamics*, Math Sci. Press, Berkeley-Ames, Series B, Vol. II, 195-237.

Honeywell (1991) Application Module Configuration - Course Note.

Hunt, L. R. , Su, R. and Meyer, G. (1983a) Design for Multi-Input Nonlinear Systems. In: *Differential Geometric Control Theory*, (R. W. Brockett, R. S. Millman, and H. J. Sussman, Eds.), Birkhauser, Bostonne.

Hunt, L. R. , Su, R. and Meyer, G. (1983b) Global Transformations of Nonlinear Systems. *IEEE Transactions on Automatic Control*, **AC-28**(1), 24-31.

Isidori, A. (1995) *Nonlinear Control Systems: An Introduction (3rd ed.)*. Springer-Verlag, Berlin.

Isidori, A. and Krener, A. J. (1982) On Feedback Equivalence of Nonlinear Sytems. *Systems & Control Letters*, **2** (2), 118-121.

Isidori, A. and Ruberti, A. (1984) On the Synthesis of Linear input-output responses for Nonlinear Systems. *Systems & Control Letters*, **4**, 17-22.

Jager, B. de (1992) Nonlinear Control System Analysis and Design with MAPLE. In: *Artificical Intelligence, Expert Systems and Symbolic Computing* (E. N. Houstis and J. R. Rice, Eds), Amsterdam, North-Holland, 155-164.

Jager, B. de (1994) Computer Aide Hybrid Analysis and Design of Nonlinear Control Systems. In: *Proc. 5th Int. Symp. on Application of multivariable System Techniques*, Bradford, UK, March, 247-254.

Jager, B. de (1995) The Use of Symbolic Computation in Nonlinear Control: Is it Viable? *IEEE Transactions on Automatic Control*, **40**(1), 84-89.

Kang, W. (1991) Extended Controller Normal Form, Invariants and Dynamic feedback Linearization of Nonlinear Control Systems. *PhD Thesis*, University of California, Davis, U.S.A..

Kailath, T. (1980) *Linear Systems*. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Kalman, R. E. (1972) Kronecker Invariants and Feedback. In: *Ordinary Differential Equations*, (L. Weiss, Ed.), Academic Press, 459-471.

Kantor, J. C. (1987) An Overview of Nonlinear Geometrical Methods for Process Control. In: *Shell Process Control Workshop*, (D. M. Prett and M. Morari, Eds.), Butterworth, London.

Khorasani, K. (1989) Robust Stabilisation of Nonlinear Systems with Unmodelled Dynamics. *Int. J. Control*, **50**(3), 827-844.

Kim, H. W. and Groves, F. R. (1993) Robustness Analysis of Feedback Linearization for Parametric and Structural Uncertainties. *Chem. Eng. Comm.*, **119**, 71-94.

Kravaris, C. and Arkun, Y. (1989) Geometric Nonlinear Control - An Overview. In: *Chemical Process Control - CPC IV*, (Y. Arkun and W. H. Ray, Eds.), CACHE, Austin, AIChE 1991, New York.

Kravaris, C. and Chung, C.-B. (1987) Nonlinear State Feedback Synthesis by Global Input/Output Linearization. *AIChE Journal*, **33**(4), 529-603.

Kravaris, C. and Kantor, C. J. (1990) Geometric Methods for Nonlinear Process Control: Part 1 and 2. *Ind. Eng. Chem. Res.* , **29**(12), 2295-2323.

Kravaris, C. and Palanki, S. (1988) Robust Nonlinear State Feedback Under Structured Uncertainty. *AIChE Journal*, **34**(7), 1119-1127.

Kravaris, C. and Soroush, M. (1990) Synthesis of Multivariable Nonlinear Controllers by Input/Output Linearization. *AIChE Journal*, **36**(2), 249-264.

Kravaris, C., Daoutidis, P. and Wright, R. A. (1994) Output Feedback Control of Nonminimum-Phase Nonlinear Processes. *Chemical Engineering Science*, **49**(13), 2107-2122.

Krener, A. J. (1984) Approximate Linearization by State Feedback and Coordinate Change. *Systems and Control Letters*, **5**, 181-185.

Krener, A. J. and Isidori, A. (1983) Linearization by Output Injection and Non-linear Observers. *Systems and Control Letters*, **3**, 47-52.

La Salle, J. and Lefschetz, S. (1961) *Stability by Liapunov's Direct Method.* Academic Press.

Lee, P. L. (1989) Direct Use of Nonlinear Models for Process Control. In: *Chemical Process Control - CPC IV*, (Y. Arkun and W. H. Ray, Eds.), CACHE, Austin, AIChE 1991, New York.

Lee, P. L. and Sullivan, G. R. (1988) Generic Model Control (GMC). *Comput. Chem. Engng.* , **12**(6), 573-580.

Lee, P. L., Ed. (1993) *Nonlinear Process Control: Applications of Generic Model Control.* Springer-Verlag, London.

Liao, T.-L., Fu, L.-C. and Hsu, C.-F. (1991) Robust Output Tracking Control of Mismatch Uncertain Nonlinear Systems. *Proceedings of the American Control Conference*, Boston, 528-532.

Loparo, K. A. and Blankenship, G. L. (1978) Estimating the Domain of Attraction of Nonlinear Feedback Systems. *IEEE Transactions on Automatic Control*, **AC-23**(4), 602-608.

Lu, X. Y. and Bell, D. J. (1994) An Intrinsic Property of Relative Order and its Application. *Int. Journal of Control*, **60**(5), 1045-1050.

Luenberger, D. G. (1967) Canonical Forms for Linear Multivariable Systems. *IEEE Transactions on Automatic Control*, June, 290-293.

Lyapunov, A. M. (1992) PhD Thesis: The General Problem of Motion Stability. Reprinted in *Int. Journal of Control*, **55**(3), 529-784.

Marino, R. and Tomei, P. (1993) Robust Stabilization of Feedback Linearizable Time-varying Uncertain Nonlinear Systems. *Automatica*, **29**(1), 181-189.

McLellan, P. J., Harris, T. J. and Bacon, D. W. (1990) Error Trajectory Descriptions of Nonlinear Controller Designs. *Chemical Engineering Science*, **45**(10), 3017-3034.

Michel, A. N., Miller, R. K. and Nam, B. H. (1982) Stability Analysis of Interconnected Systems Using Computer Generated Lyapunov Functions. *IEEE Trans. Circuits Syst.*, **CAS-29**(7), 431-440.

Nakamoto, K. and Watanabe, N. (1991) Multivariable Control Experiments of Nonlinear Chemical Process Using Nonlinear Feedback Transformation. *J. Process Control*, **1**, May, 140-145.

Newell, R. B. and Lee, P. L. (1989) *Applied Process Control: A Case Study*. Prentice-Hall of Australia Pty Ltd., Sydney.

Nijmeijer, H. (1982) Invertibility of Affine Nonlinear Control Systems : A Geometric Approach. *Systems & Control Letters*, **2**(3), 163-168.

Ogunye, B. A. (1994) On-Line Multivariable Control of Continuous Solution Polymerisation Reactors: *PhD Thesis in Chemical Engineering*, University of Waterloo, Waterloo, Ontario, Canada.

Palanki, S., Kravaris, C. and Wang, H. Y. (1994) Optimal Feedback Control of Batch Reactors with a State Inequality Constraint and Free Terminal Time. *Chemical Engineering Science*, **49**(1), 85-97.

Pavelle, R. and Wang, P. S. (1985) MACSYMA from F to G. *Journal of Symbolic Computation*, **1**, 69-100.

Phelps, A. R. and Krener, A. J. (1988) Computation of Observer Normal Form Using Macsyma. In: *Analysis and Control of Nonlinear Systems*, (C. I. Byrnes, C. F. Martin and R. E. Saeks, Eds.), Elsevier Science Pubblishers B. V., North-Holland, 475-482.

Pleau, M. and McLellan, J. (1995) Robust Control of Uncertain Systems with Shrinkable Normalized Uncertainty Envelope. To appear in *IEEE Transactions on Automatic Control*.

Pollard, J. F. and Brosilow, C. B. (1985) Model Selection for Model Based Controller. *Proceedings of the American Control Conference*, **3**, 1286-1292.

Quaak, P. and Gerritsen, J. B. M. (1990) Modelling Dynamic Behaviour of Multiple-Effect Falling-film Evaporators. In: *Computer Applications in Chemical Engineering* (H. Th. Bussemaker and P. D. Ledema, Eds.), Elsevier Science Publishers B. V., Amsterdam, Netherlands.

Qu, Z. and Dorsey, J. (1992) Robust Control by two Lyapunov Functions. *Int. J. Control*, **55**(6), 1335-1350.

Ray, H. W. (1981) *Advanced Process Control.* McGraw-Hill Book Company, New York, Chapter 3 and 5.

Rayna, G. (1987) *REDUCE: Software for Algebraic Computation.* Springer-Verlag, Berlin.

Reboulet, C. and Champetier, C. (1984) A new method for linearizing non-linear systems : the pseudolinearization. *Int. J. Control*, **40**(4), 631-638.

Rothfuß, R., Schaffner, J., and Zeitz, M. (1993) Rechnergestützte Analyse and Synthese nichtlinearer Systeme. In: *Nichtlinearer Regelung: Methoden, Werkzeugen, Anwendungen*, vol. 1206 of VDI Berichte, VDI-Verlag, Düsseldorf, 267-291.

Rouchon, P. (1992) Remarks on Some Applications of Nonlinear Control Techniques to Chemical Processes. In: *IFAC Nonlinear Control Systems Design: Selected Papers from the 2nd IFAC Symposium, Bordeaux, France, 24 - 26 June 1992* (M. Fliess, Ed), Pergamon Press, Oxford.

Rugh, W. J. (1981) Nonlinear System Theory : The Volterra/Wiener Approach. *PhD Thesis*, The John Hopkins University Press, Baltimore & London.

Rugh, W. J. (1984) An Input- Output Characterisation for Linearization by Feedback. *Systems & Control Letters* **4**, 227-229.

Runyon, C. H., Rumsey, T. R. and McCarthy, K. L. (1991) Dynamic Simulation of a Nonlinear Model of a Double Effect Evaporator. *Journal of Food Engineering*, **14** (3), 185-201.

Seborg, D. E., Edgar, T. F. and Mellichamp, D. A. (1989) *Process Dynamics and Control.* John Wiley & Sons, Inc., New York.

Seborg, D. E., Edgar, T. F. and Shah, S. L. (1986) Adaptive Control Strategies for Process Control: A Survey. *AIChE Journal*, **32**(6), 881-913.

Shinsky, F. G. (1988) *Process Control Systems: Application, Design and Tuning (3rd ed.).* McGraw-Hill Inc., Singapore.

Simulink: Dynamic System Simulation Software (1992) *User's Guide.* The MathWorks Inc.

Slotine, J. - J. E. and Li, W. (1991) *Applied Nonlinear Control.* Prentice-Hall, Inc., U.S.A..

Slotine, J. -J. E. and Hendrick, J. K. (1993) Robust Input-Output Linearization. *Int. Journal of Control*, **57**(5), 1133-1139.

Soroush, M. and Kravais, C. (1992) Discrete-Time Nonlinear Controller Synthesis by Input Output Linearization. *AIChE Journal*, **38**(12), 1923-1945.

Soroush, M. and Kravais, C. (1994) Synthesis of Discrete-Time Nonlinear Feedforward/Feedback Controllers. *AIChE Journal*, **40**(3), 473-495.

Soroush, M. and Kravaris, C. (1993) Multivariable Nonlinear Control of a Continuous Polymerization Reactor - An Experimental Study. *AIChE Journal*, **39**(12), 1920-1937.

Soroush, M. and Kravaris, C. (1994) Nonlinear Control of a Polymerization CSTR with Singular Characteristic Matrix. *AIChE Journal*, **40**(6), 980-990.

Su, R., Meyer, G. and Hunt L. R. (1983) Robustness in Nonlinear Control. In: *Differential Geometric Control Theory*, (R.W. Brockett, R. S. Millman, and H.J. Sussman, Eds.), Birkhauser, Bostonne.

Su, R. (1982) On the Linear Equivalents of Nonlinear Systems. *Systems and Control Letters*, **2**(1), 48-52.

Taylor, R. and Atherley, K. (1995) Chemical Engineering with MAPLE. *Chemical Enginering Education, Winter*, 56-61.

Thau, F. E. (1973) Observing the State of Non-linear Dynamic Systems. *Int. Journal of Control*, **17**(3), 471-479.

To, L. C., Tadé, M. O., Kraetzl, M., and Le Page, G. P. (1994) A Differential Geometric Approach to Control an Evaporation Process. *Proc. of Chemeca '94*, Perth, Australia, September, **2**, 735-742.

To, L. C., Tadé, M. O., Kraetzl, M. and Le Page, G. P. (1995a) Nonlinear Control of a Simulated Industrial Evaporation Process. *Journal of Process Control*, **5**(3), 173-182.

To, L. C., Tadé, M. O., and Kraetzl, M. (1995b) An Uncertainty Vector Adjustment for Process Modelling Error Compensation. To appear in *Journal of Process Control*.

To, L. C., Tadé, M. O. and Kraetzl, M. (1995c) *Technical Report 1/96: Nonlinear Control in Alumina Refinery*. School of Chemical Engineering, Curtin University of Technology, Perth, West Australia.

To, L. C., Tadé, M. O. and Kraetzl, M. (1996a) An Integrated MAPLE Package for Nonlinear Control Studies. *Chemical Engineering Issue of the Institution of Engineers, Singapore*, **36**(3), 49-55.

To, L. C., and Tadé, M. O. (1996b) An Uncertainty Vector Description for Process Modelling Error: A Lyapunov Function Analysis. *Proc. of APCChE 1996 Congress*, Taipei, Taiwan, March, **2**, 637-641.

To, L. C., Tadé, M. O., and Kraetzl, M. (1996c) Process Modelling Error Description Using an Uncertainty Vector. *Proc. of 1996 IChemE Research Event / Second European Conference for Young Researchers*, Leeds, UK, April, **2**, 729-731.

To, L. C., Tadé, M. O. and Kraetzl, M. (1996d) *Technical Report 6/96: Listings of Nonlinear Control Procedures*. School of Chemical Engineering, Curtin University of Technology, Perth, West Australia.

To, L. C. and Tadé, M. O. (1996e) *Technical Report 7/96: Nonlinear State Observers.* School of Chemical Engineering, Curtin University of Technology, Perth, West Australia.

To, L. C. and Tadé, M. O. (1996f) A Chemical Engineering Application of Geometric Nonlinear Filter. *Proc. of Chemeca '96,* Sydney, Australia, September.

Tonelli, S. M., Romagnoli, J. A. and Porras, J. A. (1990) Computer Package for Transient Analysis of Industrial Multiple-Effect Evaporators. *Journal of Food Engineering*, **12** (4), 267-281.

Vannelli, A. and Vidyasagar, M. (1985) Maximal Lyapunov Functions and Domains of Attraction for Autonomous Nonlinear Systems. *Automatica*, **21**(1), 69-80.

Vidyasagar, M. (1993) *Nonlinear Systems Analysis.* Prentice-Hall, Englewood Cliffs, N. J..

Walcott, B. L., Corless, M. J. and Żak, S. H. (1987) Comparative Study of Non-linear State-Observation Techniques. *Int. Journal of Control*, **45**(6), 2109-2132.

Walker, J. A. and McClamoroch, H. N. (1967) Finite Regions of Attraction for the Problem of Lur'e. *Int. Jorunal of Control*, **6**(4), 331-336.

Watanabe, K. (1992) *Adaptive Estimation and Control: Partitioning Approach.* Prentice Hall International, U. K., Chapter 2.

Weissenberger, S. (1968) Application of Results from the Absolute Stability Problem to the Computation of Finite Stability Domains. *IEEE Transactions on Automatic Control*, **AC-13**(1), 124-125.

Willems, J. L. (1971) Direct Methods for Transient Stability Studies in Power System Analysis. *IEEE Transactions on Automatic Control*, **AC-16**(4), 332-341.

Woham, W. M. (1979) *Linear Multivariable Control: A Geometric Approach.* Springer-Verlag, New York.

Wolfram, S. (1988) *MATHEMATICA: A System for Doing Mathematics by Computer.* Addison-Wesley, New York.

Wooff, C. D. and Hodgkinson, D. E. (1987) *muMATH: The First Computer Algebra System*. Academic Press.

Wright, R. A. and Kravaris, C. (1993) Dynamically Equivalent Outputs and Their Use in Nonlinear Controller Synthesis. *Chemical Engineering Science*, **48**(18), 3207-3223.

Zhou, W. (1990) Generic Model Control (GMC): Extensions and Applications. *PhD Thesis*, Department of Chemical Engineering, University of Queensland, Australia.

Zhou, W. and Lee, P. L. (1992) Robust Stability Analysis of Generic Model Control. *Chem. Eng. Comm.*, **117**, 41-72.

Zubov, V. I. (1964) *Methods of A. M. Lyapunov and Their Application*. Noordhoff, Gronigen, 90-96.

APPENDIX A

A MAPLE PACKAGE FOR NONLINEAR CONTROL STUDY

A.1 INTRODUCTION

Mathematical programming in MAPLE is straight-forward and conventional and allows repetitions, conditional executions, data structures and visualisation. Combined with MAPLE commands, powerful MAPLE procedures can easily be written and implemented for specific applications. The simulation package in this context includes five MAPLE procedures running on a UNIX platform. The procedures follow the input output linearization (I/O) developed by Kravaris and Soroush (1990) and the special case of SHM transformation proposed in Chapter 2. Apart from computing the associated transformation relationships for each control theory, the procedures are able to simulate the closed loop systems graphically using the nonlinear control schemes studied in this context. As a result, different systems and scenarios can be investigated using these procedures.

A.2 MATHEMATICAL PRELIMINARY

The theories of I/O and the special case of SHM transformation are detailed in Chapter 2. To implement the simulation of the closed loop dynamics of a nonlinear system on computer, it is necessary to determine the state trajectory, that is, the solution of the nonlinear state equations (2.11) in Chapter 2. This can be done by first linearizing (2.11) in Chapter 2 using the first-order Taylor series. That is,

$$\dot{\mathbf{x}} = \Delta \mathbf{x} + \Psi \mathbf{u} + \Gamma \mathbf{d} \qquad \text{(A.1)}$$

The above linear state equations can then be solved analytically as shown in Ray (1981). For an autonomous linear system, the analytical solution takes the following

form

$$\mathbf{x}(t) = e^{\Delta(t-t_o)}\mathbf{x}_o + \int_{t_o}^{t} e^{\Delta(t-r)}\left(\Psi\,\mathbf{u}(r) + \Gamma\,\mathbf{d}(r)\right)dr \quad \text{.....(A.2)}$$

where Δ, Ψ and Γ are Jacobian matrices with respect to the vector of states $\mathbf{x}$, inputs $\mathbf{u}$ and disturbances $\mathbf{d}$, correspondingly. $\mathbf{x}_o$ indicates the initial states at time t_o. (A.2) is implemented in MAPLE in order to obtain the state trajectory at any time t. The use of the simple equation (A.1) and (A.2) is justifiable in the case of the single effect evaporator because the process is rather "linear" as shown in Figures 3.2. However, a more sophisticated numerical technique was required in the simulation of the triple effect evaporator.

A.3 ALGORITHMS FOR MAPLE PROCEDURES

The simulation package in this context includes five MAPLE procedures running on a UNIX platform. The procedures *io* and *shm* are developed to compute static state feedback laws and transformation using I/O and SHM transformation, respectively. The static state feedback law and the transformation are then passed to the procedures *ioloop* and *shmloop* to implement the simulations of the closed loop dynamics against specific disturbances using the corresponding control strategies. The linear control algorithm used has proportional and integral action only. The fifth procedure *L* is developed to compute the Lie derivative required by both control strategies. Data is generated and stored in *sequence* format so that plots can be displayed for each state, output and input variables with the XMAPLE package *plots*. The five procedures also require the MAPLE package *linalg* to perform the necessary matrix manipulations. The arguments required by each procedure are minimal and are fully explained and checked in each procedure. The plots generated can be displayed using the command *display* from the *plots* package. The structures of the implementation of the procedures *io* and *ioloop* are shown in *Figures A.1* and *A.2*. The flowcharts for procedures *shm* and *shmloop* are very similar to *Figures A.1* and *A.2*; hence, they are omitted. It should be noted that the flowcharts presented here do not represent a complete specification of the procedures by themselves, rather they should be used as an aid in understanding of the integral listings of the procedures

which are documentated in Technical Report 6/96 and are excluded from this thesis due to space limitiation.

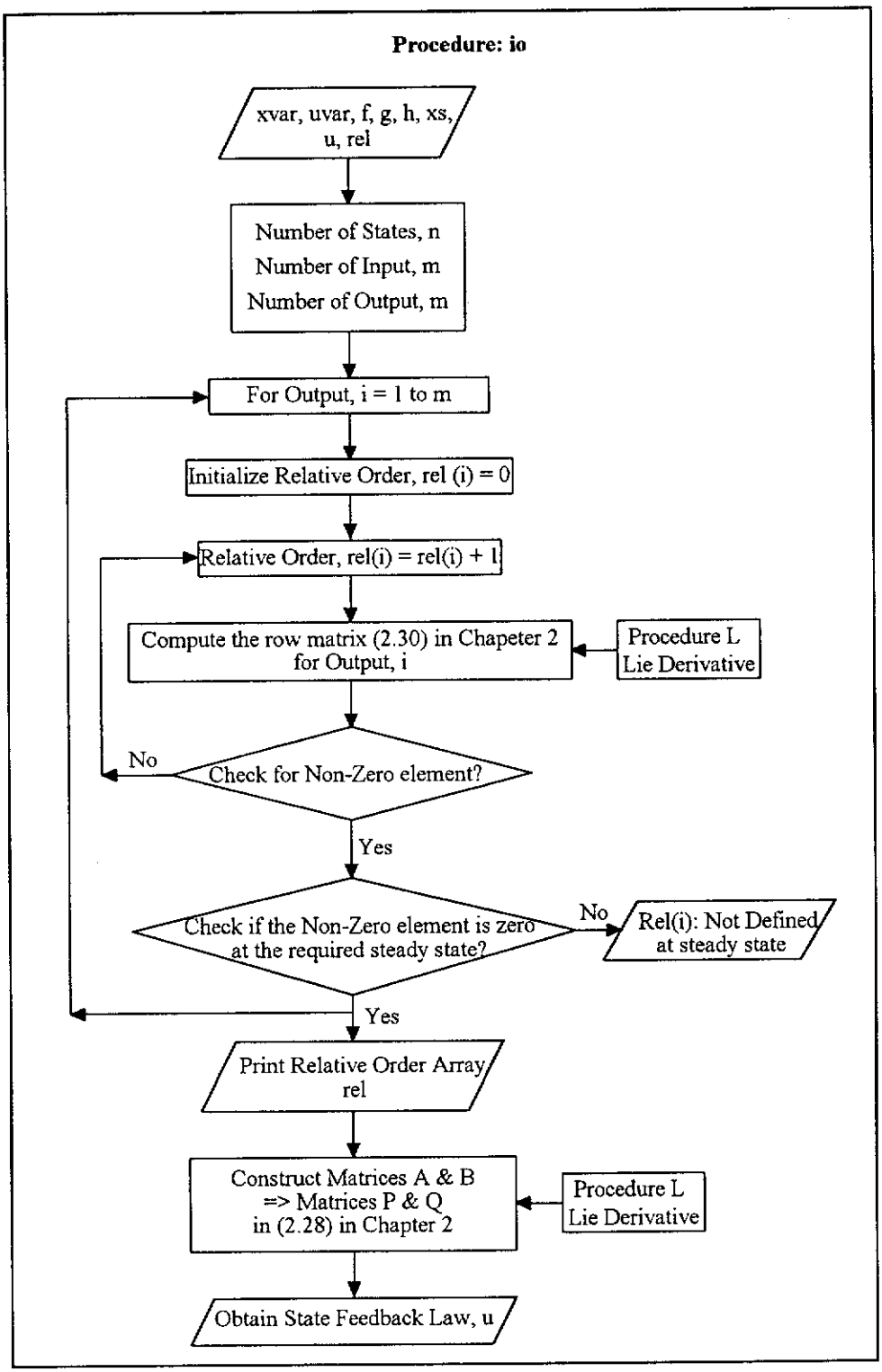


Figure A.1: Flowchart of Procedure *io*

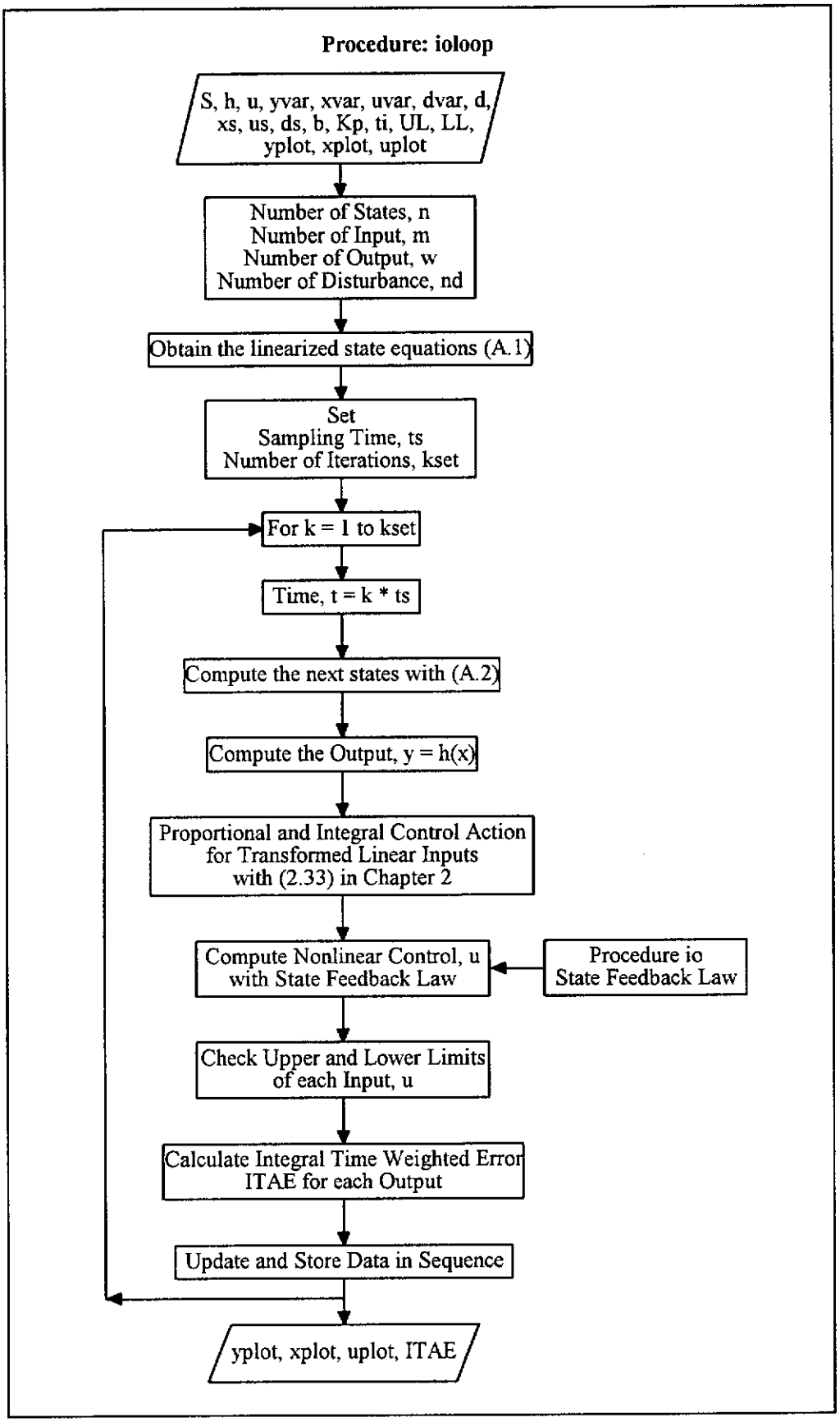


Figure A.2: Flowchart of Procedure *ioloop*

A.4 USER GUIDE FOR MAPLE PROCEDURES

A.4.1 Descriptions of Procedures

This section contains the user's guide for implementing the procedures described above. It is important to note that the user has to use *with ( )* command to include the relevant MAPLE packages in his or her calling sequences in order to run the procedures. In addition, the procedures require XMAPLE V.3 to implement. The following sections provide a description of each procedure in the package.

A.4.1.1 L - Lie Derivative

Calling Statement: L (variables, f, h);

Synopsis: The procedure computes the Lie derivative of a scalar function *h* in the direction of vector *f*, given the independent variables, variables. (See Page 2-9)

MAPLE Package: *linalg* for linear algebra calculations

Input Arguments:

variables	list of variables e.g. variables := [x1, x2, x3];
f	vector of functions depending on variables e.g. f := vector ([f1, f2, f3]);
h	a scalar function depending on variables

A.4.1.2 io and ioloop - Input Output Linearization

io	Relative Order and State Feedback Law of Input Output Linearization
Calling Statement:	io (xvar, uvar, f, g, h, xs, u, rel);
Synopsis:	The procedure determines the relative order array and the state feedback law.
MAPLE Package:	*linalg* for linear algebra calculations
MAPLE Procedure:	*L* for Lie derivative calculation

Input Arguments:

xvar	state space variables x, defined as a list of variables e.g. xvar := [x1, x2,...];
uvar	input variables u, defined as a list of variables e.g. uvar := [u1, u2,...];
f	state vector f(x), defined as a vector e.g. f := vector([f1, f2,...]);
g	input matrix g(x), with g1, g2,..., gn as the corresponding input vectors, defined as a matrix e.g. g := augment (g1, g2,...);
h	output vector h(x), defined as a vector e.g. h := vector([h1, h2,...]);
xs	steady state of state variables, defined as a list of real numbers e.g. xs := [xls, x2s,...];

Output Parameters:

u	feedback laws computed, defined as an array
rel	relative order array, defined as an array

ioloop	Closed loop dynamics of input output linearization
Calling Statement:	ioloop (S, h, u, yvar, xvar, uvar, dvar, d, xs, us, ds, b, Kp, ti, UL, LL, yplot, xplot, uplot):
Synopsis:	The procedure generates data for plots to evaluate closed dynamics of a nonlinear process with equal number of outputs and inputs using the input output linearization.
MAPLE Package:	*linalg* for linear algebra calculations *plots* for display plots generated
MAPLE Procedure:	*io* for computing the state feedback law, *u*

Input Arguments:

S	state equations, defined as a vector of equations e.g. S:=vector([S1, S2,...]);

h	output vector, defined as a vector of equations e.g. h:=vector([h1, h2,...]);
u	feedback law computed using procedure *io*, defined as an array
yvar, xvar, uvar, dvar	output, state, input, and disturbance variables, defined as a list of variables e.g. yvar:=[y1, y2,...];
d	disturbance vector, in deviation terms, defined as a vector of real numbers e.g. d:=vector([1, 2,...]);
xs, us, ds	steady state of state, input and disturbance variables, defined as a list of real numbers e.g. xs := [xls, x2s,...];
b	design parameters, defined as an array of real numbers e.g. b := array(1..no of inputs, 0..relative order, [[1,...],[1,..],..]);
Kp	proportional gain for linear controller, (deviation/deviation), defined as a list of real numbers e.g. Kp:=[10, 10,...];
ti	integral time for linear controller (hr), defined as a list of real numbers e.g. ti:=[1, 1,...];
UL, LL	upper and lower limits of input variables in deviations, defined as a list of real numbers e.g. UL:=[10, 10,...];

Output Parameters:

yplot, xplot, uplot	plots for output, state, and input variables in deviation terms use command display (yplot[i]); to view yplot[i]
ITAE	integral time weighted error for each output y, a local print-out from the procedure

A.4.1.3 shm and shmloop - Su-Hunt-Meyer Transformation

shm transformation array for Su-Hunt-Meyer transformation

Calling Statement: shm (xvar, uvar, f, g, T);

Synopsis: The procedure computes the state and input transformation for a nonlinear system with equal number of states and inputs using the simplified solution algorithm proposed in section 4.1.4 of chapter 2.

MAPLE Package: *linalg* for linear algebra calculations

MAPLE Procedure: *L* for Lie derivative calculation

Input Arguments:

xvar state space variables x, defined as a list of variables
e.g. xvar := [x1, x2,...];

uvar input variables u, defined as a list of variables
e.g. uvar := [u1, u2,...];

f state vector f(x), defined as a vector
e.g. f := vector([f1, f2,...]);

g input matrix g(x), with g1, g2,..., gn as the corresponding input vectors, defined as a matrix
e.g. g := augment (g1, g2,...);

Output Parameters:

T transformation functions computed, defined as an array
e.g. T := array (1..n+n);
where
T[1], T[2],..., T[n] = state transformation
T[n+1], T[n+2],..., T[n+n] = input transformation

shmloop closed loop dynamics of Su-Hunt-Meyer transformation

Calling Statement: shmloop (S, h, T, yvar, xvar, uvar, dvar, d, Kp, ti, UL, LL, yplot, xplot, uplot):

Synopsis:	The procedure generates data for plots to evaluate closed dynamics of a nonlinear process with equal number of states and inputs using simplified Su-Hunt-Meyer transformation.
MAPLE Package:	*linalg* for linear algebra calculations *plots* for display plots generated
MAPLE Procedure:	*shm* for computing the transformation array, T
Input Arguments:	
S	state equations, defined as a vector of equations e.g. S:=vector([S1, S2,...]);
h	output vector, defined as a vector of equations e.g. h:=vector([h1, h2,...]);
T	transformation relationships computed using procedure *shm*, defined as an array
yvar, *xvar*, *uvar*, *dvar*	output, state, input, and disturbance variables, defined as a list of variables e.g. yvar:=[y1, y2,...];
d	disturbance vector, in deviation terms, defined as a vector of real numbers e.g. d:=vector([1, 2,...]);
Kp	proportional gain for linear controller, (deviation/deviation), defined as a list of real numbers e.g. Kp:=[10, 10,...];
ti	integral time for linear controller (hr), defined as a list of real numbers e.g. ti:=[1, 1,...];
UL, LL	upper and lower limits of input variables in deviations, defined as a list of real numbers e.g. UL:=[10, 10,...];
Output Parameters:	
yplot, xplot, uplot	plots for output, state, and input variables in deviation terms use command display (yplot[i]); to view yplot[i]

ITAE	integral time weighted error for each output y, a local print-out from the procedure

A.4.2 Other Options and Settings in Procedures ioloop and shmloop

Other pertinent programming options and default settings in procedure *ioloop* and *shmloop* are listed as follow.

Number of iterations, *kset*	default value = 15
Frequency of data output and stored, *out*	default value = 5
Sampling time, *ts*	default value = 0.01
Base time, *t0*	default value = 0

A.4.3 Other Conditions Required by Procedure ioloop and shmloop

Two other implicit conditions are required by procedures *ioloop* and *shmloop* in order to implement the closed loop response of a system.

- Both procedures require that the system starts from its steady state.
- As presented in the Chapter 2, SHM transformation requires all states and inputs to be in deviation term. Therefore, the procedure *shmloop* always sets the steady state to the origin.

A.5 ILLUSTRATIVE EXAMPLE TO USE THE MAPLE PROCEDURES

The illustrative example presented here is the evaporator stage of liquor burning process. Details of the process can be found in Chapter 3. The calling sequences for procedures *io* and *ioloop* are shown in *Appendix B* with the sample output, while the sequences and output for procedures *shm* and *shmloop* are presented in the same *appendix*. The procedures have been tested extensively with examples from Essen (1992), which are detailed in Comito (1995). *Figures A.3* and *A.4* show examples of graphs generated in XMAPLE V.3. The plot generated in XMAPLE is printed as a *gif* file, colour-edited and inserted into Window WinWord 6.0 as a picture. The procedures developed here form the basis of the MAPLE procedure developed for the simulation of a triple effect evaporator in Chapter 7.

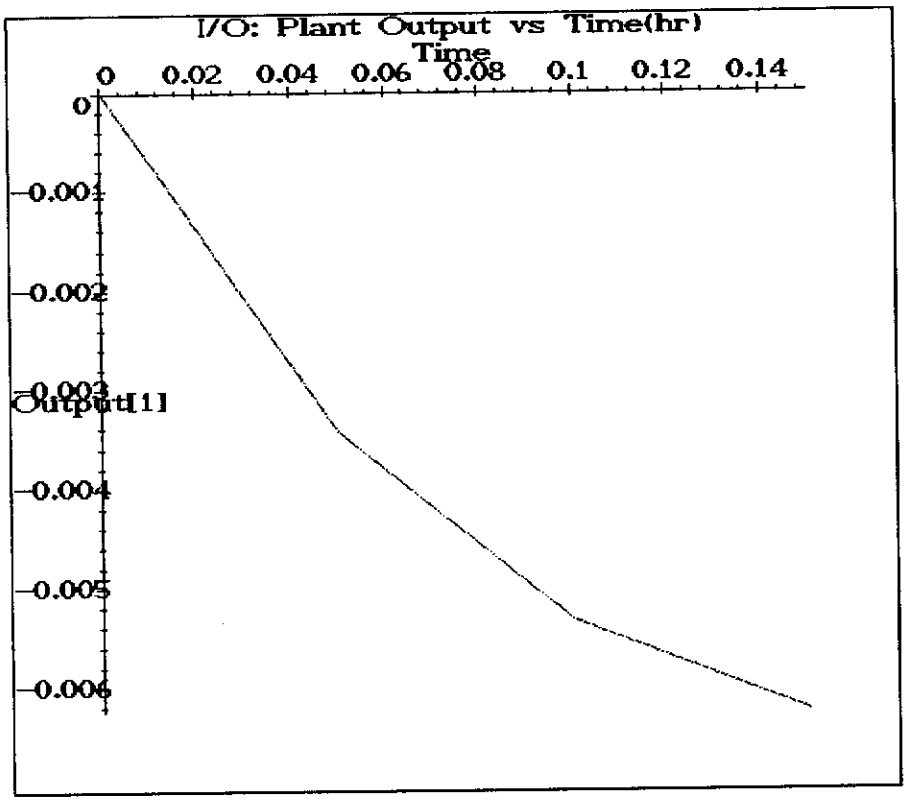


Figure A.3: Sample Plot of Procedure *ioloop*

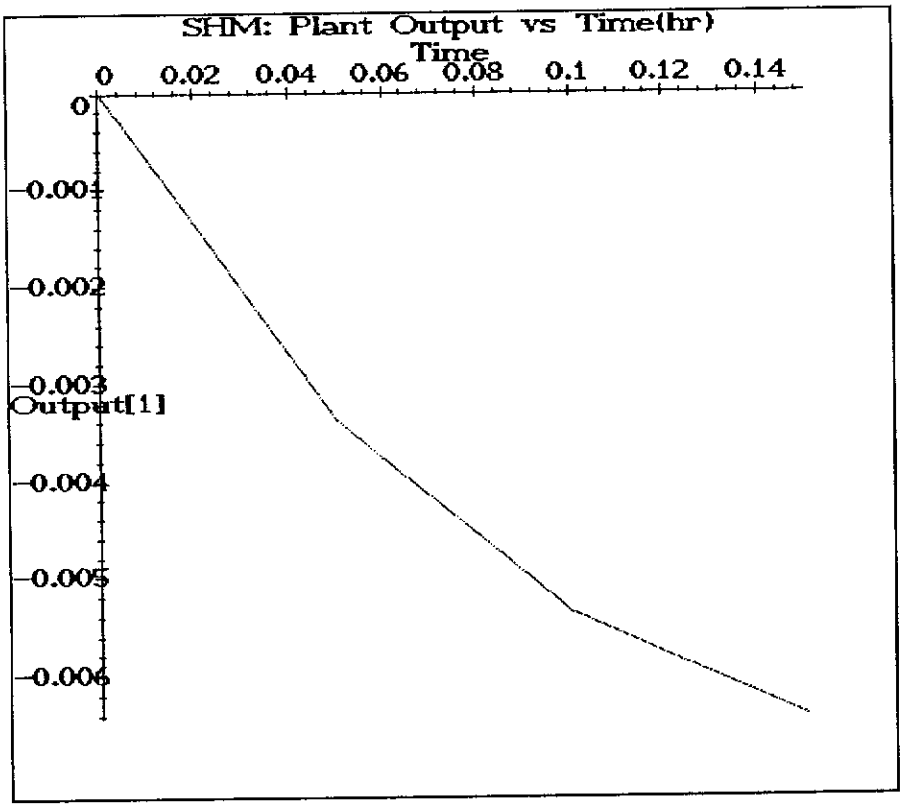


Figure A.4: Sample Plot of Procedure *shmloop*

A.6 DISCUSSION

A.6.1 Single Effect Evaporator

The single effect evaporator in the liquor burning unit involved a three dimensional problem with two inputs and two outputs. MAPLE implemented the algorithms for the I/O and successfully computed the state feedback law and simulated the closed loop dynamics. For the SHM transformation, attempts to solve the original algorithm presented in Hunt et al (1983a) as discussed in section 2.4.1.3 proved abortive. This was because the original algorithm involved computing of the Lie bracket and solving a system of ordinary differential equations which necessitated more sophisticated *solve* and *dsolve* procedures (Jager, 1995). Therefore, the special case with the simplified solution algorithm was proposed to redress the problem. In order to implement the simplified algorithm, the example was modified to a three-input three-output system so that the number of states equals the number of inputs. MAPLE successfully solved and simulated the closed loop system in this case.

A.6.2 Triple Effect Evaporator

Since MAPLE was able to handle the above example smoothly and successfully, a triple effects evaporator model was next considered as a challenging task. The model has nine state equations with six inputs and six outputs. The I/O strategy was tested. The first problem experienced during development was the zero-equivalence problem (Heck, 1993) or rather round-off error in the numerical sense. MAPLE allows the user to employ as many digits as required. However, the verification of the equality of the following equations turned out to be non-trivial in MAPLE.

$$S = -0.04325259516\left(Q_{pl} + Q_f\right) - 179.9307959\left(\frac{Q_{cw}\left(T_1 - 48\right)}{0.2321460 \times 10^7 - 4160\,T_1}\right)$$

$$R = -\frac{100409.1696\left(Q_{pl} + Q_f\right) + 179.9307959\,T_1\left(Q_{pl} - Q_{cw} - Q_f\right) + 8636.678203\,Q_{cw}}{-0.2321460 \times 10^7 + 4160\,T_1}$$

It can be easily shown that $S = R$ by hand calculation. However, the answer obtained

for (S - R) using MAPLE is not an exact zero. For example,

$$S - R = 0.4 \times 10^{-10} \frac{(43T_1 - 49833)(Q_{pl} - Q_f) - 250Q_{cw}}{-116073 + 208T_1}$$

The above answer was obtained using 100 digits. A slightly different answer was obtained when the number of digits used was 10. Despite the fact that the leading term is 10^{-10}, the numerical value of the rest of the expression could be in the region of 10^5 - 10^7. Therefore, (S - R) would fall in the range of 10^{-5} - 10^{-3} and hence a serious round-off error was observed in the results. This problem was observed only when *numeric* calculations involved a division. Similar situations were encountered when solving *linear* equations using *solve*. Therefore, it is recommended to perform all calculation symbolically first and then substitute the numeric values for the relevant symbolic parameters afterwards. However, MAPLE results are not easy to check. Inherent to the use of symbolic computation is the creation of results which contain very long and intricate expressions. This is clearly shown in the state feedback laws and transformations in *Appendix C*. The clarity and evaluation of such results are difficult and almost impossible.

Another problem was with the *linalg* package which failed to return a solution for (A.2) for the linearized model of the triple effects evaporator. MAPLE failed to return a solution for the term exp(Δ(t-r)). This could be due to memory limitations caused by the fact that Δ is a 9 by 9 matrix. Therefore, instead of using the linear analytical solution, the state trajectory was obtained using the simple Euler method. With the above problems noted and corrected, MAPLE successfully implemented the simulation of the triple effects evaporator using the I/O scheme.

In general, the MAPLE computations are smooth, fast and efficient in terms of time and programming statements. Concerning the limits of the software, these can often be corrected by modifying the procedures to accommodate more possibilities. Sometimes the limitation could lie in the control algorithm itself or in the properties of the system under study. A suitable ERROR message is usually returned, either by

MAPLE or the relevant procedure. Obviously there is a lot of room for improvement in the procedures developed here. The procedures form a template for more sophisticated procedures and have achieved the objective of demonstrating the capability and limitations of MAPLE as an integrated simulation package for complicated nonlinear control theories.

A.7 CONCLUSIONS ON DEVELOPED MAPLE PROCEDURES

An integrated simulation package for two nonlinear control schemes was developed using the symbolic computational software, MAPLE V.3. The package contains five procedures which automate the computational algorithms for I/O and SHM transformation. At the same time, graphic representation of the closed loop responses of a system using the nonlinear control strategies is achieved. The capability of MAPLE was also explored. Typical textbook problems can be solved successfully by MAPLE. However, more complicated problems cause some difficulties in computations. It was found that computation cannot be done for complex methods like the original SHM transformation. Modifications in procedures *solve* and *dsolve* are recommended. The zero-equivalence problem which leads to round-off error is very undesirable. Despite of all these problems, MAPLE, in most situations, provided symbolic and numerical results quickly and efficiently with a tremendous gain in time and with a minimal effort. Even though the strict viability of employing MAPLE for investigating nonlinear control systems cannot be established, there is no doubt that MAPLE is the most suitable tool among the few means available and should be used in both academia and industry by instructors to illustrate fundamental concepts in different areas of chemical engineering. The one to one correspondence between symbolic code in MAPLE V.3 and mathematical algorithms being implemented encourages better understanding and appreciation of the aspects of mathematics in process control. With the unique advantages and different possibilities possessed by symbolic computations, its developments have opened a new dimension in the design and analysis of control algorithms.

APPENDIX B

CALLING SEQUENCES FOR MAPLE PROCEDURES

B.1 CALLING SEQUENCES FOR PROCEDURES IO AND IOLOOP

```
#
#     File: iotest
#     Calling sequence for io
#
#     By To Lap Chi
#     on 11-10-95
#
#
#     Specify all data
#
with(linalg):
with(plots):
A:=23.12:
Cpcw:=4160:
Cphw:=4160:
Cpd:=3142:
F:=21:
Tcw:=40:
Thd:=95:
lambdav:=2660000:
rhof:=1.3:
#
#     Steady State values for u1s and u2s
#
#     u1s = Feed Flowrate, tonne/hr
#     u2s = Cooling Water Flowrate, tonne/hr
#
#
u1s:=25.0147:
u2s:=313.087:
us:=[u1s, u2s]:
#
#     Steady State values for Disturbance, d1s and d2s
```

```
#
#     d1s = Product Flowrate, Qp, tonne/hr
#     d2s = Heater Feed Flowrate, Qhf, tonne/hr
#
#
d1s:=10:
d2s:=1000:
ds:=[d1s, d2s]:
#
#     Steady State values
#
#     x1s = height in BFT, m
#     x2s = density of liquor out of BFT, tonne/m3
#     x3s = temperature in BFT, deg.C
#
x1s:=2.71:
x2s:=1.762:
x3s:=88.431:
xs:=[x1s,x2s,x3s]:
#
#     Specify g1 vector
#
g1:=vector([1/A,(rhof-x2)*(1-d1s/d2s)/(A*x1),((Thd-x3)/(A*x1*x2))*(rhof-
x2)*(1-d1s/d2s)]):
#
#     Specify g2 vector
#
m:=Cpcw*(x3-Tcw-F)/(A*(lambdav-Cphw*(x3-F))):
g2:=vector([-m,m*(x2-1)/x1,m*(x3-lambdav/Cpd)/(x1*x2)]):
#
g:=augment(g1,g2):
#
#     Specify D1 vector
#
D1:=vector([-1/A,(x2-rhof)*u1s/(A*x1*d2s),((Thd-x3)/(A*x1*x2))*((x2-
rhof)*u1s/d2s-x2)]):
#
#     Specify  D2 vector
#
D2:=vector([0,-(x2-rhof)*d1s*u1s/(A*x1*d2s^2),(Thd-x3)/(A*x1*x2)*
(x2-(x2-rhof)*(d1s*u1s/d2s^2))]):
#
A1:=scalarmul(D1,d1s):
A2:=scalarmul(D2,d2s):
f:=add(A1,A2):
#
#     Specify h1 & h2
```

```
#
h1:=x1:
h2:=x2:
h:=vector([h1,h2]):
#
#   Set yvar, xvar, uvar, dvar
#
yvar:=[x1, x2]:
xvar:=[x1,x2,x3]:
uvar:=[u1, u2]:
dvar:=[d1, d2]:
#
#   Set up State Equations
#
mv:= (Cpcw*(x3-(Tcw+F))/(lambdav-Cphw*(x3-F)))*u2:
#
S1:=1/A*(u1-d1-mv):
S2:=1/(A*x1)*((d2-d1)*u1*rhof/d2-mv-(d2-d1-mv-(d2-d1)*(d2-u1)/d2)
*x2):
S3:=1/(Cpd*A*x1*x2)*(((d2-d1)*(u1*rhof+(d2-u1)*x2)*Cpd*Thd/d2)-(mv
*lambdav)-((d2-d1)*(u1*rhof+(d2-u1)*x2)/d2-mv)*Cpd*x3):
#
S:=vector([S1, S2, S3]):
#
#   Specify the disturbance
#
d:=vector([2,0]):
#
#   Specify the Proportional Gain and Integral Time
#
Kp:=[10, 10]:
ti:=[0.6666666, 0.5]:
#
#   Specify the upper and lower limits
#
UL:=[25, 350]:
LL:=[-25, -300]:
#
#   Call procedure io
#
io(xvar,f,g,h,xs,u,rel);
#
#   Specify design parameters, b
#   Note: b must be specified after the calling of io
#         because io returns an empty array of b with u
#
b:=array(1..2, 0..1, [[1,1],[0.5,0.5]]):
```

```
#
#   Call procedure ioloop
#
ioloop(S,h,u,yvar,xvar,uvar,dvar,d,
                        xs,us,ds,b,Kp,ti,UL,LL,
                        yplot,xplot,uplot):
```

B.2 CALLING SEQUENCES FOR PROCEDURES SHM AND SHMLOOP

```
#
#   File: shmtest
#   Calling sequence for shm and shmloop
#
#   By TO Lap Chi
#   Date: 11-10-95
#
#
#      Specify all data
#
with (plots):
with(linalg):
A:=23.12:
Cpcw:=4160:
Cphw:=4160:
Cpd:=3142:
F:=21:
Tcw:=40:
lambdav:=2660000:
rhof:=1.3:
#
#      Steady State values for u1s and u2s
#
#      u1s = Feed Flowrate, tonne/hr
#      u2s = Cooling Water Flowrate, tonne/hr
#    u3s = Heater Discharge Temperature, deg.C
#
u1s:=25.0147:
u2s:=313.087:
u3s:=95:
#
#      Steady State values for Disturbance, d1s and d2s
#
#      d1s = Product Flowrate, Qp, tonne/hr
#      d2s = Heater Feed Flowrate, Qhf, tonne/hr
#
d1s:=10:
d2s:=1000:
```

```
#
#     Steady State values
#
#     x1s = height in BFT, m
#     x2s = density of liquor out of BFT, tonne/m3
#     x3s = temperature in BFT, deg.C
#
x1s:=2.71:
x2s:=1.762:
x3s:=88.431:
#
#     Convert to deviation variables
#
x1:=x1s+dx1:
x2:=x2s+dx2:
x3:=x3s+dx3:
#
u1:=u1s+du1:
u2:=u2s+du2:
u3:=u3s+du3:
#
d1:=d1s+dd1:
d2:=d2s+dd2:
#
#   Specify g1 vector
#
g1:=vector([1/A,(rhof-x2)*(1-d1s/d2s)/(A*x1),(1/(A*x1*x2))*(rhof-
x2)*(1-d1s/d2s)*(u3s-x3)]):
#
#   Specify g2 vector
#
m:=Cpcw*(x3-Tcw-F)/(A*(lambdav-Cphw*(x3-F))):
g2:=vector([-m,m*(x2-1)/x1,m*(x3-lambdav/Cpd)/(x1*x2)]):
#
#   Specify g3 vector
#
g3:=vector([0,0,(1/(A*x1*x2))*((rhof-x2)*(1-d1s/d2s)*u1s+
(d2s-d1s)*x2)]):
#
g:=augment (g1, g2, g3):
#
#   Form f vector
#
H:=vector([-d1s/A,0,1/(A*x1)*(u1s*u3s*(1-d1s/d2s)*((x2-rhof)/x2)+
(d1s-d2s)*x3)]):
#
A1:=scalarmul(g1,u1s):
```

```
A2:=scalarmul(g2,u2s):
A3:=scalarmul(g3,u3s):
add(A1,A2):
add(",A3):
f:=add(",H):
mv:= (Cpcw*(x3-(Tcw+F))/(lambdav-Cphw*(x3-F)))*u2:
#
#     Set up State Equations, S
#
S1:=1/A*(u1-d1-mv):
S2:=1/(A*x1)*((d2-d1)*u1*rhof/d2-mv-(d2-d1-mv-(d2-d1)*(d2-u1)/d2)
*x2):
S3:=1/(Cpd*A*x1*x2)*(((d2-d1)*(u1*rhof+(d2-u1)*x2)*Cpd*u3/d2)-(mv
*lambdav)-((d2-d1)*(u1*rhof+(d2-u1)*x2)/d2-mv)*Cpd*x3):
#
S:=vector([S1, S2, S3]):
#
#     Set xvar, uvar, dvar, yvar
#
xvar:=[dx1, dx2, dx3]:
uvar:=[du1, du2, du3]:
dvar:=[dd1, dd2]:
yvar:=[dx1, dx2, dx3]:
d:=vector([2,0]):
#
#     Specify h1, h2 and h3
#
h1:=dx1:
h2:=dx2:
h3:=dx3:
h:=vector([h1, h2, h3]):
#
#     Set the proportional gain and integral time
#
Kp:=[10, 10, 20]:
ti:=[0.6666666, 0.5, 1e10]:
#
#     Set upper and lower limit of inputs
#
UL:=[25, 350, 20]:
LL:=[-25, -300, -20]:
#
#     Call procedure shm and shmloop
#
shm(xvar,uvar,f,g,T);
shmloop(S,h,T,yvar,xvar,uvar,dvar,d,Kp,ti,UL,LL,
                    yplot,xplot,uplot):
```

APPENDIX C

SAMPLE OUTPUTS FOR MAPLE PROCEDURES

C.1 SAMPLE OUTPUT OF PROCEDURES IO AND IOLOOP

Warning: new definition for norm
Warning: new definition for trace

Relative Order Array:
[1, 1]

Characteristic matrix:

$$\left[.04325259516\, b[1,1],\ 1.124567474\, \frac{b[1,1]\,(x3 - 61.)}{-17171. + 26.\, x3}\right]$$

$$\left[-.004282006920\, \frac{b[2,1]\,(-13. + 10.\, x2)}{x1},\ -1.124567474\, \frac{b[2,1]\,(x3 - 61.)\,(x2 - 1.)}{(-17171. + 26.\, x3)\, x1}\right]$$

State feedback control law for input, 1, :

$$\frac{1.124567474\, b[2,1]\, x2\, v[1] - 1.124567474\, b[2,1]\, x2\, b[1,0]\, x1 + .4864046168\, b[2,1]\, x2\, b[1,1] - 1.124567474\, b[2,1]\, v[1] + 1.124567474\, b[2,1]\, b[1,0]\, x1 - .4864046168\, b[2,1]\, b[1,1] + 1.124567474\, x1\, b[1,1]\, v[2] - 1.124567474\, x1\, b[1,1]\, b[2,0]\, x2}{(.0004864046200\, x2 + .01395981250)\, b[1,1]\, b[2,1]}$$

State feedback control law for input, 2, :

```
- ( - 17171. + 26. x3) ( - .05566608996 b[2, 1] v[1]
   + .05566608996 b[2, 1] b[1, 0] x1 - .02407702853 b[2, 1] b[1, 1]
   + .04282006920 b[2, 1] x2 v[1] - .04282006920 b[2, 1] x2 b[1, 0] x1
   + .01852079118 b[2, 1] x2 b[1, 1] + .04325259516 x1 b[1, 1] v[2]
   - .04325259516 x1 b[1, 1] b[2, 0] x2)/
  ((.0004864046200 x2 + .01395981250) b[1, 1] b[2, 1] (x3 - 61.))
```

Matrix A:

$$\begin{bmatrix} 0 & 0 & -.02481022129 \\ & & \\ & -6 & \\ .13*10 & -.15561123 & .006976158169 \\ & & \\ -.00025697 & 57.43366241 & -19.50041664 \end{bmatrix}$$

Matrix B:

$$\begin{bmatrix} .04325259516 & -.002074262889 \\ & \\ -.007299952763 & .0005832429224 \\ & \\ -.02721531770 & -.3293453149 \end{bmatrix}$$

Matrix D:

$$\begin{bmatrix} -.04325259516 & 0 \\ & \\ & -5 \\ .00018445063 & -.184450*10 \\ & \\ -.1041559930 & .1048367754 \end{bmatrix}$$

Closed Loop Dynamics - t, y, x, u:

$$.05, \left[-.003445098, .8211 \cdot 10^{-5} \right],$$

$$\left[-.003445097825, .8210660910 \cdot 10^{-5}, -.1639461205 \right],$$

$$\left[2.25005468, 29.9959135 \right]$$

$$.10, \left[-.005329050, .000016227 \right],$$

$$\left[-.005329049648, .00001622725292, -.5063215959 \right],$$

$$\left[3.61367480, 51.6374497 \right]$$

$$.15, \left[-.006251955, .000036752 \right],$$

$$\left[-.006251955172, .00003675189709, -.8166971469 \right],$$

$$\left[4.35921890, 64.8701697 \right]$$

Integral Time Weighted Absolute Errors:

$$\begin{bmatrix} & -6 \\ .0000618782329 & .2455035 \cdot 10 \end{bmatrix}$$

```
> display (yplot[1]);
```

```
>
```

C.2 SAMPLE OUTPUT OF PROCEDURES SHM AND SHMLOOP

Warning: new definition for norm
Warning: new definition for trace

Transformation, 1, :

23.12000000 dx1

Transformation, 2, :

1306.488206 ln(.7620000000 + dx2) + 355.1148914

Transformation, 3, :

.06370592190 dx3

Transformation, 4, :

$$.2500000000*10^{15}\,\frac{-7\,(-.2974358799*10^{15}+.5200000002*10^{12}\,dx3)\,du1}{-.7435897*10^{7}+13000.\,dx3}$$

$$+.2500000000*10^{14}\,\frac{-7\,(.1426412000*10^{14}+.5199999999*10^{12}\,dx3)\,du2}{-.7435897*10^{7}+13000.\,dx3}$$

$$+.2500000000*10^{8}\,\frac{-7\,.31004000*10^{8}+.1706128840*10^{15}\,dx3}{-.7435897*10^{7}+13000.\,dx3}$$

Transformation, 5, :

$$-.0003266220515\,(-.2942067966*10^{17}+.5143546713*10^{14}\,dx3$$

$$-.6368112481*10^{17}\,dx2+.1113321799*10^{15}\,dx2\,dx3)\,du1$$

$$/\,/\,((-.7435897*10^{7}+13000.\,dx3)\,(271.+100.\,dx1)\,(381.+500.\,dx2))$$

$$/$$

$$- .0003266220515\,(.2350618391\cdot 10^{16} + .3084801038\cdot 10^{16}\,dx2$$

$$+ .8569204150\cdot 10^{14}\,dx3 + .1124567474\cdot 10^{15}\,dx2\,dx3)\,du2$$

$$/\,/\,((-.7435897\cdot 10^{7} + 13000.\,dx3)(271. + 100.\,dx1)(381. + 500.\,dx2))/$$

$$- .0003266220515\,(-.1415731426\cdot 10^{13} + .2811570697\cdot 10^{17}\,dx3$$

$$- .6271531308\cdot 10^{18}\,dx2 + .3799368674\cdot 10^{17}\,dx2\,dx3)$$

$$/\,/\,((-.7435897\cdot 10^{7} + 13000.\,dx3)(271. + 100.\,dx1)(381. + 500.\,dx2))/$$

Transformation, 6, :

$$.5000000000\cdot 10^{-15}\,(-.3550655689\cdot 10^{28} - .1977691755\cdot 10^{28}\,dx2$$

$$+ .6207525999\cdot 10^{25}\,dx3 + .3457550961\cdot 10^{25}\,dx2\,dx3)\,du3/$$

$$(\%1\,(271. + 100.\,dx1)(881. + 500.\,dx2)) + .5000000000\cdot 10^{-15}\,($$

$$- .2051727762\cdot 10^{25}\,dx2\,dx3 - .9478982251\cdot 10^{24}\,dx3$$

$$+ .3546259580\cdot 10^{22}\,dx2\,dx3^{2} + .1332477233\cdot 10^{26}\,dx2 + .1638371926\cdot 10^{22}\,dx3^{2}$$

$$+ .6156044812\cdot 10^{25})\,du1/(\%1\,(271. + 100.\,dx1)(881. + 500.\,dx2))$$

$$+ .5000000000*10^{-15}$$

$$\frac{(.2617542646*10^{25}\,dx3 - .3582080383*10^{22}\,dx3^2 + .7449718368*10^{26})\,du2}{\%1\,(271. + 100.\,dx1)\,(881. + 500.\,dx2)}$$

$$+ .5000000000*10^{-15}\,(-.1576281969*10^{24} - .1299145717*10^{29}\,dx2$$

$$+ .2000404407*10^{28}\,dx2\,dx3 + .4410951501*10^{28}\,dx3$$

$$- .3457550960*10^{25}\,dx2\,dx3^2 - .7329028798*10^{25}\,dx3^2)/$$

$$(\%1\,(271. + 100.\,dx1)\,(881. + 500.\,dx2))$$

$$\%1 := -.7435897*10^{7} + 13000.\,dx3$$

Matrix A

$$\begin{bmatrix} 0 & 0 & -.02481022130 \\ .13*10^{-6} & -.15561122 & .006976158171 \\ -.00025680 & 57.43366257 & -19.50041664 \end{bmatrix}$$

Matrix B

$$\begin{bmatrix} .04325259516 & -.002074262889 & 0 \\ -.007299952763 & .0005832429224 & 0 \\ -.02721531760 & -.3293453149 & 15.69712784 \end{bmatrix}$$

Matrix D

$$\begin{bmatrix} -.04325259516 & 0 \\ .00018445063 & -.184450*10^{-5} \\ -.1041559926 & .1048367754 \end{bmatrix}$$

Closed Loop Dynamics - t, y, x, u:

.05, [-.003385228419, .00001485683866, -.006638188019],

[-.003385228419, .00001485683866, -.006638188019],

[2.024272234, 25.15601428, .5312861880]

.10, [-.005566401246, .00002336434738, -.009168993985],

[-.005566401246, .00002336434738, -.009168993985],

[3.474315104, 43.16659244, .9115411582]

.15, [-.006545540784, .00002767890555, -.01017696582],

[-.006545540784, .00002767890555, -.01017696582],

[4.295926545, 53.35074610, 1.126534763]

Integral Time Weighted Absolute Errors:

$$\begin{bmatrix} .00006432662351 & .2714187624*10^{-6} & .0001059276998 \end{bmatrix}$$

> display (yplot[1]);

>

APPENDIX D

MODEL EQUATIONS FOR TRIPLE EFFECTS EVAPORATOR

D.1 MODEL EQUATONS FOR TRIPLE EFFECTS EVAPORATOR

$$Qd1 := Qp1 + Qhf1 - Qf$$

$$Rho\_hf1 := \frac{Rho\_f\,Qf + Rho\_1\,Qhf1 - Rho\_1\,Qf}{Qhf1}$$

$$dh1dt := \frac{-mv1 - Qp1 + Qf}{A1}$$

$$dRho\_1dt := \frac{Rho\_f\,Qf - Rho\_1\,Qf - mv1 + Rho\_1\,mv1}{A1\,h1}$$

$$dT1dt := ((Rho\_f\,Qf + Rho\_1\,Qhf1 - Rho\_1\,Qf)\,Cphd1\,Thd1 - mv1\,Hv1$$

$$- Cp1\,T1\,(Rho\_f\,Qf + Rho\_1\,Qhf1 - Rho\_1\,Qf - mv1))/(Cp1\,A1\,h1\,Rho\_1)$$

$$Qd2 := Qp2 + Qhf2 - Qp1$$

$$Rho\_hf2 := \frac{Rho\_1\,Qp1 + Rho\_2\,Qhf2 - Rho\_2\,Qp1}{Qhf2}$$

$$dh2dt := \frac{-mv2 - Qp2 + Qp1}{A2}$$

$$dRho\_2dt := \frac{Rho\_1\,Qp1 - Rho\_2\,Qp1 - mv2 + Rho\_2\,mv2}{A2\,h2}$$

$$dT2dt := ((Rho\_1\ Qp1 + Rho\_2\ Qhf2 - Rho\_2\ Qp1)\ Cphd2\ Thd2 - mv2\ Hv2 - Cp2\ T2\ (Rho\_1\ Qp1 + Rho\_2\ Qhf2 - Rho\_2\ Qp1 - mv2))/(Cp2\ A2\ h2\ Rho\_2)$$

$$Qd3 := Qp3 + Qhf3 - Qp2$$

$$Rho\_hf3 := \frac{Rho\_2\ Qp2 + Rho\_3\ Qhf3 - Rho\_3\ Qp2}{Qhf3}$$

$$dh3dt := \frac{-mv3 - Qp3 + Qp2}{A3}$$

$$dRho\_3dt := \frac{Rho\_2\ Qp2 - Rho\_3\ Qp2 - mv3 + Rho\_3\ mv3}{A3\ h3}$$

$$dT3dt := ((Rho\_2\ Qp2 + Rho\_3\ Qhf3 - Rho\_3\ Qp2)\ Cphd3\ Thd3 - mv3\ Hv3 - Cp3\ T3\ (Rho\_2\ Qp2 + Rho\_3\ Qhf3 - Rho\_3\ Qp2 - mv3))/(Cp3\ A3\ h3\ Rho\_3)$$

$$mv1 := \frac{Qcw\ Cpcw\ (T1 - TF - Tcw)}{Hv1 - Cpcw\ (T1 - TF)}$$

$$Thd1 := Thf1 + \frac{ms1\ Hs1 + mv2\ Hv2 + mv3\ Hv3}{(Rho\_f\ Qf + Rho\_1\ Qhf1 - Rho\_1\ Qf)\ Cphf1}$$

$$mv2 := \frac{Qp1\ Rho\_1\ Cp1\ T1 + ms2\ Hs2 - Qp2\ Rho\_2\ Cp2\ T2}{Hv2}$$

$$Thd2 := Thf2 + \frac{ms2\ Hs2\ (Rho\_1\ Qp1 + Rho\_2\ Qhf2 - Rho\_2\ Qp1)\ Cphf2}{Qhf2^2}$$

$$mv3 := \frac{Qp2\ Rho\_2\ Cp2\ T2 + ms3\ Hs3 - Qp3\ Rho\_3\ Cp3\ T3}{Hv3}$$

$$Thd3 := Thf3 + \frac{ms3\ Hs3\ (Rho\_2\ Qp2 + Rho\_3\ Qhf3 - Rho\_3\ Qp2)\ Cphf3}{Qhf3^2}$$

$$Thf1 := \frac{Rho\_f\ Qf\ Cpf\ Tf + Rho\_1\ Cp1\ T1\ Qhf1 - Rho\_1\ Cp1\ T1\ Qf}{(Rho\_f\ Qf + Rho\_1\ Qhf1 - Rho\_1\ Qf)\ Cphf1}$$

$$Thf2 := \frac{Qp1\ Rho\_1\ Cp1\ T1 + Rho\_2\ Cp2\ T2\ Qhf2 - Rho\_2\ Cp2\ T2\ Qp1}{(Rho\_1\ Qp1 + Rho\_2\ Qhf2 - Rho\_2\ Qp1)\ Cphf2}$$

$$Thf3 := \frac{Qp2\ Rho\_2\ Cp2\ T2 + Rho\_3\ Cp3\ T3\ Qhf3 - Rho\_3\ Cp3\ T3\ Qp2}{(Rho\_2\ Qp2 + Rho\_3\ Qhf3 - Rho\_3\ Qp2)\ Cphf3}$$

> dh1dt;

$$-\frac{Qp1}{A1} + \frac{Qf}{A1} - \frac{Qcw\ Cpcw\ (T1 - TF - Tcw)}{A1\ (Hv1 - Cpcw\ (T1 - TF))}$$

> dh2dt;

$$\frac{\left(\frac{Rho\_1\ Cp1\ T1}{Hv2} + 1\right) Qp1}{A2} + \frac{\left(\frac{Rho\_2\ Cp2\ T2}{Hv2} - 1\right) Qp2}{A2} - \frac{ms2\ Hs2}{A2\ Hv2}$$

> dh3dt;

$$\frac{\left(\frac{Rho\_2\ Cp2\ T2}{Hv3} + 1\right) Qp2}{A3} + \frac{\left(\frac{Rho\_3\ Cp3\ T3}{Hv3} - 1\right) Qp3}{A3} - \frac{ms3\ Hs3}{A3\ Hv3}$$

> dRho_1dt;

$$\frac{(Rho\_f - Rho\_1)\ Qf}{A1\ h1} + \frac{\left(\frac{Cpcw\ (T1 - TF - Tcw)}{Hv1 - Cpcw\ (T1 - TF)} + \frac{Rho\_1\ Cpcw\ (T1 - TF - Tcw)}{Hv1 - Cpcw\ (T1 - TF)}\right) Qcw}{A1\ h1}$$

> dRho_2dt;

$$\frac{\left(-\rho_2 - \frac{\rho_1 C_{p1} T_1}{H_{v2}} + \rho_1 + \frac{\rho_2 \rho_1 C_{p1} T_1}{H_{v2}}\right) Q_{p1}}{A_2 h_2}$$

$$+ \frac{\left(\frac{\rho_2^2 C_{p2} T_2}{H_{v2}} + \frac{\rho_2 C_{p2} T_2}{H_{v2}}\right) Q_{p2}}{A_2 h_2} + \frac{\left(\frac{\rho_2 H_{s2}}{H_{v2}} - \frac{H_{s2}}{H_{v2}}\right) m_{s2}}{A_2 h_2}$$

> dRho_3dt;

$$\frac{\left(-\rho_3 - \frac{\rho_2 C_{p2} T_2}{H_{v3}} + \rho_2 + \frac{\rho_3 \rho_2 C_{p2} T_2}{H_{v3}}\right) Q_{p2}}{A_3 h_3}$$

$$+ \frac{\left(\frac{\rho_3^2 C_{p3} T_3}{H_{v3}} + \frac{\rho_3 C_{p3} T_3}{H_{v3}}\right) Q_{p3}}{A_3 h_3} + \frac{\left(\frac{\rho_3 H_{s3}}{H_{v3}} - \frac{H_{s3}}{H_{v3}}\right) m_{s3}}{A_3 h_3}$$

> dT1dt;

$$\frac{\left(\frac{C_{phd1}(\rho_f C_{pf} T_f - \rho_1 C_{p1} T_1)}{C_{phf1}} - C_{p1} T_1 (\rho_f - \rho_1)\right) Q_f}{C_{p1} A_1 h_1 \rho_1} + \frac{C_{phd1} T_1 Q_{p1}}{A_1 h_1 C_{phf1}}$$

$$+ \left(- \frac{C_{phd1} \rho_3 C_{p3} T_3 Q_{p3}}{C_{phf1}}\right.$$

$$\left. + \frac{\left(\frac{C_{pcw}(T_1 - T_F - T_{cw}) H_{v1}}{H_{v1} - C_{pcw}(T_1 - T_F)} + \frac{C_{p1} T_1 C_{pcw}(T_1 - T_F - T_{cw})}{H_{v1} - C_{pcw}(T_1 - T_F)}\right) Q_{cw}}{}\right.$$

$$+\frac{Cphd1\ Hs1\ ms1}{Cphf1}+\frac{Cphd1\ Hs2\ ms2}{Cphf1}+\frac{Cphd1\ Hs3\ ms3}{Cphf1}+\frac{Cphd1\ Rho\_1\ Cp1\ T1\ Qhf1}{Cphf1}$$

$$\frac{-Rho\_1\ Cp1\ T1\ Qhf1)/(Cp1\ A1\ h1\ Rho\_1)}{}$$

> dT2dt;

$$\left(\left|Rho\_2\ Cp2\ T2-\frac{Cp2^2\ T2^2\ Rho\_2}{Hv2}\right|Qp2+\right.$$

$$\left|(Rho\_1\ Qp1+Rho\_2\ Qhf2-Rho\_2\ Qp1)\frac{Cphd2\ Hs2\ Cphf2^2}{Qhf2^2}-Hs2+\frac{Cp2\ T2\ Hs2}{Hv2}\right|ms2$$

$$+\frac{Cphd2\ (Qp1\ Rho\_1\ Cp1\ T1+Rho\_2\ Cp2\ T2\ Qhf2-Rho\_2\ Cp2\ T2\ Qp1)}{Cphf2}$$

$$-Qp1\ Rho\_1\ Cp1\ T1$$

$$-Cp2\ T2\left|Rho\_1\ Qp1+Rho\_2\ Qhf2-Rho\_2\ Qp1-\frac{Qp1\ Rho\_1\ Cp1\ T1}{Hv2}\right|)/$$

$$\frac{(Cp2\ A2\ h2\ Rho\_2)}{}$$

> dT3dt;

$$\left(\left|Rho\_3\ Cp3\ T3-\frac{Cp3^2\ T3^2\ Rho\_3}{Hv3}\right|Qp3+\right.$$

$$\left|(Rho\_2\ Qp2+Rho\_3\ Qhf3-Rho\_3\ Qp2)\frac{Cphd3\ Hs3\ Cphf3^2}{Qhf3^2}-Hs3+\frac{Cp3\ T3\ Hs3}{Hv3}\right|ms3$$

$$+\frac{Cphd3\ (Qp2\ Rho\_2\ Cp2\ T2+Rho\_3\ Cp3\ T3\ Qhf3-Rho\_3\ Cp3\ T3\ Qp2)}{Cphf3}$$

$$- Qp2\ Rho\_2\ Cp2\ T2$$

$$- Cp3\ T3 \left| Rho\_2\ Qp2 + Rho\_3\ Qhf3 - Rho\_3\ Qp2 - \frac{Qp2\ Rho\_2\ Cp2\ T2}{Hv3} \right| \Big)/$$

$$\frac{}{(Cp3\ A3\ h3\ Rho\_3)}$$

>

APPENDIX E

SIMULATION PROGRAM TRIIOUV AND SAMPLE OUTPUTS

E.1 PROGRAM *triiouv* FOR SIMULATION OF TRIPLE EFFECTS EVAPORATOR

```
#
#     File: triiouv
#     Equations Summary and Checking
#     for Triple Effect Evaporator: Effect 1
#     With Data (Revised Model)
#     With Uncertainty Vector
#     With Thd3 and Thf3 equations
#
#     By To Lap Chi
#     on 14 Sept 1995
#     on 21 Sept 1995
#
#
#     Lie derivative L_f h(x)
#     variables = list of variables
#     f = list of functions depending on variables
#     h = one function depending on variables
#
#     By Miro Kraetzl
#
L := proc(variables,f,h)

local n, i, g, j, k, var;


      n:=nops(variables);

      var:=seq(variables[i], i=1..n);

      g:=unapply(h,var);

S:=array(1..n):
```

```
for j from 1 to n do
S[j] := f[j]*D[j](g)(var);
od;

      sum(S[k], k=1..n);
      simplify(");

end:

#
#     Equations Summary and Checking
#     for Triple Effect Evaporator: Effect 1
#     With Data
#
with (linalg):
readlib (unassign):
with (plots):
setoptions(titlefont=[TIMES, ROMAN, 15]):
#
#     Data
#
#     Define the output variables
#
y[1]:=h1: y[2]:=h2: y[3]:=h3: y[4]:=T1: y[5]:=Rho_2: y[6]:=Rho_3:
#
#     Feed Stream
#
#Qf:=29.7:
#Rho_f:=1.38:
#Tf:=66:
Cpf:=3290:
#
#     HF1 Stream
#
Qhf1:=2296:
Cphf1:=Cp1:
#
#     HD1 Stream
#
Cphd1:=Cp1:
#
#     V1 Stream
#
Hv1:=2234100:
#
#     D1 Stream
#
```

```
BPE1:=16:
Cp1:=3250:
#
#     CW Stream
#
#Qcw:=77.4681:
Cpcw:=4160:
#Tapp:=5:
TF:=BPE1+Tapp:
#Tcw:=27:
#
#     Flash Tank1
#
A1:=23.12:
#
#     P1 Stream
#
#Qp1:=23.7:
#
#     S1 Stream
#
ms1:=0:
Hs1:=2110500:
#
#
Qd1:=Qp1+Qhf1-Qf:
Rho_hf1:=simplify ((Rho_f*Qf+Rho_1*Qd1-Rho_1*Qp1)/Qhf1):
dh1dt:=1/A1*(Qhf1-mv1-Qd1):
dRho_1dt:=simplify(1/(A1*h1)*(Qhf1*Rho_hf1-mv1+(mv1-Qhf1)*Rho_1)):
dT1dt:=1/(Cp1*A1*h1*Rho_1)*(Qhf1*Rho_hf1*Cphd1*Thd1-mv1*Hv1-Cp1*T1*
(Qhf1*Rho_hf1-mv1)):
#
#
#     Equations Summary and Checking
#     for Triple Effect Evaporator: Effect 2
#
#
#     Data
#
#     HF2 Stream
#
Qhf2:=2305:
Cphf2:=Cp2:
#
#     HD2 Stream
#
Cphd2:=Cp2:
```

```
#
#     V2 Stream
#
Hv2:=2190700:
#
#     D2 Stream
#
BPE2:=24:
Cp2:=3320:
#
#     Flash Tank2
#
A2:=23.12:
#
#     P2 Stream
#
#Qp2:=19.6:
#
#     S2 Stream
#
#ms2:=6.276004:
Hs2:=Hs1:
#
Qd2:=Qp2+Qhf2-Qp1:
Rho_hf2:=simplify ((Rho_1*Qp1+Rho_2*Qd2-Rho_2*Qp2)/Qhf2):
dh2dt:=1/A2*(Qhf2-mv2-Qd2):
dRho_2dt:=simplify(1/(A2*h2)*(Qhf2*Rho_hf2-mv2+(mv2-Qhf2)*Rho_2)):
dT2dt:=1/(Cp2*A2*h2*Rho_2)*(Qhf2*Rho_hf2*Cphd2*Thd2-mv2*Hv2-Cp2*T2*
(Qhf2*Rho_hf2-mv2)):
#
#
#     Equations Summary and Checking
#     for Triple Effect Evaporator: Effect 3
#
#     Data
#
#     HF3 Stream
#
Qhf3:=2302.5:
Cphf3:=Cp3:
#
#     HD3 Stream
#
Cphd3:=Cp3:
#
#     V3 Stream
#
```

```
Hv3:=2190700:
#
#     D3 Stream
#
BPE3:=30:
#Cp3:=3410:
#
#     Flash Tank2
#
A3:=23.12:
#
#     P3 Stream
#
#Qp3:=17.1:
#
#     S3 Stream
#
#ms3:=2.69646215:
Hs3:=Hs1:
Ts:=144:
#
Qd3:=Qp3+Qhf3-Qp2:
Rho_hf3:=simplify ((Rho_2*Qp2+Rho_3*Qd3-Rho_3*Qp3)/Qhf3):
#
dh3dt:=1/A3*(Qhf3-mv3-Qd3):
dRho_3dt:=simplify(1/(A3*h3)*(Qhf3*Rho_hf3-mv3+(mv3-Qhf3)*Rho_3)):
dT3dt:=1/(Cp3*A3*h3*Rho_3)*(Qhf3*Rho_hf3*Cphd3*Thd3-mv3*Hv3-Cp3*T3*
(Qhf3*Rho_hf3-mv3)):
#
mv1:=(Qcw*Cpcw*(T1-TF-Tcw))/(Hv1-Cpcw*(T1-TF)):
#
#     Find mv2 and Thd1
#
Tv2:=T2-BPE2:
Thd1:=Thf1+(ms1*Hs1+mv2*Hv2+mv3*Hv3)/(Qhf1*Rho_hf1*Cphf1):
mv2:=1/Hv2*(Qp1*Rho_1*Cp1*T1+ms2*Hs2-Qp2*Rho_2*Cp2*T2):
#
#     Find mv3 and Thd2
#
Tv3:=T3-BPE3:
Thd2:=Thf2+(ms2*Hs2/(Qhf2*Rho_hf2*Cphf2)):
mv3:=1/Hv3*(Qp2*Rho_2*Cp2*T2+ms3*Hs3-Qp3*Rho_3*Cp3*T3):
#
#     Thd3 equation
#
Thd3:=Thf3+(ms3*Hs3)/(Qhf3*Rho_hf3*Cphf3):
#
```

```
#     Thf equations
#
Thf1:=simplify ((Rho_f*Qf*Cpf*Tf+Rho_1*Qd1*Cp1*T1-Rho_1*Qp1*Cp1*T1)
/(Rho_hf1*Qhf1*Cphf1)):
Thf2:=simplify ((Rho_1*Qp1*Cp1*T1+Rho_2*Qd2*Cp2*T2-Rho_2*Qp2*Cp2*T2)
/(Rho_hf2*Qhf2*Cphf2)):
Thf3:=simplify ((Rho_2*Qp2*Cp2*T2+Rho_3*Qd3*Cp3*T3-Rho_3*Qp3*Cp3*T3)
/(Rho_hf3*Qhf3*Cphf3)):
#
#     Define variables
#
xvar:=[h1,h2,h3,Rho_1,Rho_2,Rho_3,T1,T2,T3];
uvar:=[Qf,Qp1,Qp2,Qcw,ms2,ms3];
dvar:=[Qp3];
uncer:=[Tapp,Rho_f,Tf,Tcw,Cp3];
#
#     Determining no of states
#
ns:=nops(xvar):
#
#     Determining no of inputs
#
nu:=nops(uvar):
#
#     Determining no of disturbance
#
nd:=nops(dvar):
#
#     Determining no of uncertainty
#
nuncer:=nops(uncer):
#
#     Final Equations
#
uvarseq:=seq(uvar[i], i=1..nu):
dh1dt:=collect(dh1dt, {uvarseq})+uv[1]:
dh2dt:=collect(dh2dt, {uvarseq})+uv[2]:
dh3dt:=collect(dh3dt, {uvarseq})+uv[3]:
dRho_1dt:=collect(dRho_1dt, {uvarseq})+uv[4]:
dRho_2dt:=collect(dRho_2dt, {uvarseq})+uv[5]:
dRho_3dt:=collect(dRho_3dt, {uvarseq})+uv[6]:
collect(simplify(dT1dt), {uvarseq}):
dT1dt:=collect('', Qf)+uv[7]:
dT2dt:=collect(simplify(dT2dt), {uvarseq})+uv[8]:
dT3dt:=collect(dT3dt, {uvarseq})+uv[9]:
#
#     Feedback Law by Lie Derivative
```

```
#
fvector:=vector([dh1dt, dh2dt, dh3dt, dRho_1dt, dRho_2dt, dRho_3dt,
            dT1dt, dT2dt, dT3dt]):
assign (Qp3=17.1, Tapp=5, Rho_f=1.38, Tf=66, Tcw=27, Cp3=3410):
#
#      Set the design parameters
#
b:=matrix(nu,2,[1,0.5,
         1,0.5,
         1,0.5,
         1,-0.5,
         1,0.5,
         1,0.5]);
#
#      Solve for the inverse
#
for i to nu do
    Ev[i]:=v[i]-b[i,2]*L(xvar, fvector, y[i])-b[i,1]*y[i]:
od:
solve({seq(Ev[i], i=1..nu)}, {seq(uvar[j], j=1..nu)}):
assign(");
for i to nu do
u[i]:=uvar[i]:
od:
#
#   Print Feedback law
#
for i to nu do
    print (`State feedback control law for input `,i);
     lprint(u[i]);
od;
unassign ('Qf', 'Qp1', 'Qp2', 'ms2', 'ms3', 'Qcw'):
unassign ('Qp3', 'Tapp', 'Rho_f', 'Tf', 'Tcw', 'Cp3'):
#
#      Set the steady state value
#
xs:=[2.71,2.71,2.71,1.44,1.49,1.54,90.6,129,135]:
xss:=seq(xvar[i]=xs[i], i=1..ns);
#
us:=[29.7,23.7,19.6,77.4681,6.276004,2.69646215]:
uss:=seq(uvar[i]=us[i], i=1..nu);
#
ds:=[17.1]:
dss:=seq(dvar[i]=ds[i], i=1..nd);
#
uncers:=[5, 1.38, 66, 27, 3410]:
#
```

```
#     For nominal study
#
#uncerp:=uncers:
#
#     For robustness study
#
uncerp:=[14, 1.30, 70, 30, 3415]:
#
uncerss:=seq(uncer[i]=uncers[i], i=1..nuncer);
uncerps:=seq(uncer[i]=uncerp[i], i=1..nuncer);
#
ys:=[2.71,2.71,2.71,90.6,1.49,1.54];
#
#     Set the disturance
#
d:=[Qp3=20.0];
#
#     Initializing the summation for integral action
#
ysum:=matrix(1,nu,0):
#
#     Initializing sampling time
#
t0:=0:
ts:=0.02;
#
#     Initializing the u and x vectors
#
u0:=vector(nu,us):
x0:=vector(ns,xs):
#
#     Set the UL and LL
#
UL:=[30, 30, 30, 500, 50, 50]:
LL:=[seq(-us[vv], vv=1..nu)]:
#
#     Initializing the ITAEx
#
ITAEx:=matrix(1,nu,0):
#
#     Specifying the number of iterations
#
kset:=500:
#
#     Set the printout rate, out
#
out:=2:
```

```
#
#     Ensure kset is multiple of 'out'
#
kset:=out*iquo(kset,out);
#
#     Initializing the arrays
#
x1:=vector(ns,0):
u1:=vector(nu,0):
for i to ns do
xdset[0,i]:=0e-30;
od:
for i to nu do
udset[0,i]:=0;
od:
#
#     PI controller parameters
#
Kp:=[10, 10, 10, -500, 10, 10];
ti:=[1, 1, 1, 1, 1, 1, 1];
#
#     Uncertainty Vector Parameters
#
uvp:=[10,0,0,0,0,0,10,0,0];
uv0:=seq(uv[i]=0, i=1..ns):
#
#     Start control action
#
print (`Closed Loop Dynamics - t, x, u:`);
#
for k to kset do
#
t:=k*ts;
#
#     Compute the next plant states with uncertainty
#
x0s:=seq(xvar[i]=x0[i], i=1..ns);
u0s:=seq(uvar[i]=u0[i], i=1..nu);
#
x1[1]:=subs(x0s,u0s,d,uncerps,uv0, dh1dt)*ts+x0[1];
x1[2]:=subs(x0s,u0s,d,uncerps,uv0, dh2dt)*ts+x0[2];
x1[3]:=subs(x0s,u0s,d,uncerps,uv0, dh3dt)*ts+x0[3];
x1[4]:=subs(x0s,u0s,d,uncerps,uv0, dRho_1dt)*ts+x0[4];
x1[5]:=subs(x0s,u0s,d,uncerps,uv0, dRho_2dt)*ts+x0[5];
x1[6]:=subs(x0s,u0s,d,uncerps,uv0, dRho_3dt)*ts+x0[6];
x1[7]:=subs(x0s,u0s,d,uncerps,uv0, dT1dt)/1e2*ts+x0[7];
x1[8]:=subs(x0s,u0s,d,uncerps,uv0, dT2dt)/1e1*ts+x0[8];
```

```
x1[9]:=subs(x0s,u0s,d,uncerps,uv0, dT3dt)/1e1*ts+x0[9];
#
#      Convert Actual Values to Deviation
#
scalarmul(xvar,-1);
add(x1,");
xd:=subs(xss,");
#
#      Computer the output
#
x1s:=seq(xvar[i]=x1[i], i=1..ns);
#
for i to nu do
#
y1[i]:=subs(x1s,y[i]);
yd[i]:=y1[i]-ys[i];
#
#      Integral action
#
ysum[1,i]:=ysum[1,i]+yd[i];
od:
#
#      Compute next input
#
for i to nu do
v[i]:=ys[i]-Kp[i]*yd[i]-Kp[i]*ysum[1,i]*ts/ti[i];
od:
#
#
#      Obtain the input, w, without uv
#
for i to nu do
w[i]:=subs(x1s,uv0, u[i]);
od;
#
ws:=seq(uvar[i]=w[i], i=1..nu);
#
#      Obtain model dx/dt for uv
#
uv[1]:=uvp[1]*(-subs(x1s,ws,d,uncerss,uv0, dh1dt));
uv[2]:=uvp[2]*(-subs(x1s,ws,d,uncerss,uv0, dh2dt));
uv[3]:=uvp[3]*(-subs(x1s,ws,d,uncerss,uv0, dh3dt));
uv[4]:=uvp[4]*(-subs(x1s,ws,d,uncerss,uv0, dRho_1dt));
uv[5]:=uvp[5]*(-subs(x1s,ws,d,uncerss,uv0, dRho_2dt));
uv[6]:=uvp[6]*(-subs(x1s,ws,d,uncerss,uv0, dRho_3dt));
uv[7]:=uvp[7]*(-subs(x1s,ws,d,uncerss,uv0, dT1dt)/1e2);
uv[8]:=uvp[8]*(-subs(x1s,ws,d,uncerss,uv0, dT2dt)/1e1);
```

```
uv[9]:=uvp[9]*(-subs(x1s,ws,d,uncerss,uv0, dT3dt)/1e1);
#
#     Obtain the final input, u1, with uv
#
for i to nu do
u1[i]:=subs(x1s, u[i]);
od;
#
#     Ensure u1 within limits
#
for i to nu do
if u1[i] > (us[i]+UL[i]) then
u1[i]:=us[i]+UL[i];
elif u1[i] < (us[i]+LL[i]) then
u1[i]:=us[i]+LL[i];
fi;
od:
#
#     Convert actual value to deviation
#
scalarmul(uvar,-1);
add(u1,");
ud:=subs(uss,");
#
#     Compute next ITAE
#
for i to nu do
ITAEx[1,i]:=ITAEx[1,i] + ts*abs(yd[i])*t;
od:
#
#     Print Out results and Store Coordinates for plots
#
for j from 0 to 0 do
#
if (k/out) = iquo(k,out) then
#
print(t,xd,ud);
#
for i to nu do
udseq[k/out,i]:=t, ud[i];
udset[k/out,i]:=ud[i];
od:
#
for i to ns do
xdseq[k/out,i]:=t, xd[i];
xdset[k/out,i]:=xd[i];
od:
```

```
#
fi;
#
od;
#
#     Reset Values
#
t0:=t:
x0:=x1:
u0:=u1:
#
od:
#
#     Print ITAE
#
print (` `);
print (`Integral Time Weighted Absolute Errors:`);
print (ITAEx);
#
#     Data Sequences for plots
#
for i to ns do
xdg[i]:=[0,0,seq(xdseq[j,i], j=1..kset/out)];
xdmm[i]:=seq(xdset[j,i], j=0..kset/out);
xdmax[i]:=max(xdmm[i]);
xdmin[i]:=min(xdmm[i]);
od:
#
for i to nu do
udg[i]:=[0,0,seq(udseq[j,i], j=1..kset/out)];
udmm[i]:=seq(udset[j,i], j=0..kset/out);
udmax[i]:=max(udmm[i]);
udmin[i]:=min(udmm[i]);
od:
#
#     Generation of Plots
#
tg:=t/1:
uplot[1]:=plot({seq(udg[i], i=1..1)}, Time=0..tg,
title=`I/O: Liquor Flow (m3/hr) vs Time (hr)`,
color=red):
uplot[2]:=plot({seq(udg[i], i=2..2)}, Time=0..tg,
title=`I/O: Liquor Flow (m3/hr) vs Time (hr)`, style=point, symbol=cross,
color=white):
uplot[3]:=plot({seq(udg[i], i=3..3)}, Time=0..tg,
title=`I/O: Liquor Flow (m3/hr) vs Time (hr)`, style=point, symbol=box,
color=blue):
```

```
uplot[4]:=plot({seq(udg[i], i=4..4)}, Time=0..tg,
title=`I/O: CW Flow (1000 kg/hr)  vs Time (hr)`, color=red):
uplot[5]:=plot({seq(udg[i], i=5..5)}, Time=0..tg,
title=`I/O: Steam Flow (1000 kg/hr) vs Time (hr)`, style=point,symbol=cross,
color=white):
uplot[6]:=plot({seq(udg[i], i=6..6)}, Time=0..tg,
title=`I/O: Steam Flow (1000 kg/hr) vs Time (hr)`, style=point, symbol=box,
color=blue):
#display({seq(uplot[i], i=1..3)});
#display({seq(uplot[j], j=5..6)});
#display(uplot[4]);
#
xplot[1]:=plot({seq(xdg[i], i=1..1)}, Time=0..tg,
title=`I/O: Height (m) vs Time (hr)`,
color=red):
xplot[2]:=plot({seq(xdg[i], i=2..2)}, Time=0..tg,
title=`I/O: Height (m) vs Time (hr)`, style=point, symbol=cross,
color=white):
xplot[3]:=plot({seq(xdg[i], i=3..3)}, Time=0..tg,
title=`I/O: Height (m) vs Time (hr)`, style=point, symbol=box,
color=blue):
#display({seq(xplot[i], i=1..3)});
#display(xplot[3]);
#
xplot[4]:=plot({seq(xdg[i], i=4..4)}, Time=0..t,
title=`I/O: Density (1000 kg/m3) vs Time (hr)`,
color=red):
xplot[5]:=plot({seq(xdg[i], i=5..5)}, Time=0..tg,
title=`I/O: Density (1000 kg/m3) vs Time (hr)`, style=point, symbol=cross,
color=white):
xplot[6]:=plot({seq(xdg[i], i=6..6)}, Time=0..tg,
title=`I/O: Density (1000 kg/m3) vs Time (hr)`, style=point, symbol=box,
color=blue):
#display({seq(xplot[i], i=5..6)});
#display(xplot[4]);
#
xplot[7]:=plot({seq(xdg[i], i=7..7)}, Time=0..tg,
title=`I/O: Temperature (deg. C) vs Time (hr)`,
color=red):
xplot[8]:=plot({seq(xdg[i], i=8..8)}, Time=0..tg,
title=`I/O: Temperature (deg. C) vs Time (hr)`, style=point, symbol=cross,
color=white):
xplot[9]:=plot({seq(xdg[i], i=9..9)}, Time=0..tg,
title=`I/O: Temperature (deg. C) vs Time (hr)`, style=point, symbol=box,
color=blue):
#display({seq(xplot[i], i=7..9)});
#
```

```
#   Save data for excel files
#
#   for k to kset/out do lprint (xdseq[k,1], xdset[k,7]); od;
#
```

E.2 SAMPLE OUTPUTS OF PROGRAM *triiouv*

```
        xvar := [h1, h2, h3, Rho_1, Rho_2, Rho_3, T1, T2, T3]

             uvar := [Qf, Qp1, Qp2, Qcw, ms2, ms3]

                  dvar := [Qp3]

             uncer := [Tapp, Rho_f, Tf, Tcw, Cp3]

                    [ 1   .5 ]
                    [        ]
                    [ 1   .5 ]
                    [        ]
                    [ 1   .5 ]
                b := [        ]
                    [ 1  -.5 ]
                    [        ]
                    [ 1   .5 ]
                    [        ]
                    [ 1   .5 ]
```

State feedback control law for input , 1

```
-.4438775683e-16*(.6816135352e40*T1^2*v[1]+.3588329369e44
+.9703178362e44*v[3]-.6816135352e40*T1^2*h1+.5269697128e37
*T1^3*h1-.3408067676e40*T1^2*uv[1]+.2634848564e37*T1^3*uv[1]
+.5269697128e37*T1^3*v[5]*h2-.5269697128e37*T1^3*v[1]
+.3843572231e45*Rho_2*h2-.1921786115e45*v[5]*h2+.2557734178e28
*Rho_1*T1^3*h3-.9458748800e27*Rho_1*T1^3-.8640709164e42
*T1-.9703178362e44*h3-.1596832238e40*uv[7]*Rho_1*h1*T1
+.7057725629e41*uv[7]*Rho_1*h1-.1948785054e37*T1^3-.2634848564e37
*Rho_1*T1^3*uv[1]-.2557734178e28*Rho_1*T1^3*v[3]+.9608930577e44
*uv[5]*h2+.9608930577e44*Rho_2*uv[2]-.1921786115e45*Rho_2*v[2]
+.9608930577e44*uv[6]*h3-.2336528606e43*v[1]*T1+.2336528606e43*h1
*T1+.1411545126e42*v[4]*Rho_1*h1-.3193664475e40*v[4]*Rho_1*h1*T1
-.6816135352e40*Rho_1*T1^2*v[1]+.3843572231e45*Rho_3*h3-.1921786115e45
*v[6]*h3+.2557734178e28*Rho_1*T1^3*h2+.5269697128e37
*T1^3*v[6]*h3+.1325042662e32*Rho_3*T3+.9514682791e44*Rho_1*v[3]
-.1053939426e38*T1^3*Rho_2*h2+.9608930577e44*Rho_3*uv[3]
-.1921786115e45*Rho_3*v[3]+.2557734178e28*Rho_1*T1^3
*v[5]*h2+.2557734178e28*Rho_1*T1^3*Rho_3*v[3]-.1278867089e28*Rho_1
*T1^3*uv[5]*h2-.5115468357e28*Rho_1*T1^3*Rho_2*h2
```

-.1739052336e30*Rho_1*T1^2*v[6]*h3-.1739052336e30*Rho_1
*T1^2*Rho_2*v[2]+.8695261680e29*Rho_1*T1^2*Rho_2*uv[2]
-.1739052336e30*Rho_1*T1^2*Rho_3*v[3]+.8695261680e29*Rho_1*T1^2*uv[5]
*h2+.3478104672e30*Rho_1*T1^2*Rho_2*h2+.2336528606e43*T1*h3
+.1168264303e43*T1*uv[3]-.2336528606e43*T1*v[3]
-.1739052336e30*Rho_1*T1^2*v[5]*h2+.8695261680e29*Rho_1
*T1^2*uv[6]*h3+.3478104672e30*Rho_1*T1^2*Rho_3*h3
+.8695261680e29*Rho_1*T1^2*Rho_3*uv[3]-.9847447157e27*Rho_1
*Rho_3*T3*T1^2+.3233183011e30*Rho_1*Rho_3*T3*T1
-.2634848564e37*uv[7]*Rho_1^2*h1*T1^2-.5269697128e37*v[4]*Rho_1^2
*h1*T1^2+.5269697128e37*T1^3*h3-.5269697128e37*T1^3*v[2]
+.9514682791e44*Rho_1*v[2]+.1411545126e42*Rho_1^2*h1*T1
-.3193664475e40*Rho_1^2*h1*T1^2+.3193664475e40*v[4]
*Rho_1^2*h1*T1-.4757341396e44*Rho_1*uv[3]-.9703178359e44*Rho_1
*v[1]-.4757341396e44*Rho_1*uv[2]+.9703178359e44*Rho_1*h1
-.9514682791e44*Rho_1*h2+.4851589179e44*Rho_1*uv[1]+.1596832238e40
*uv[7]*Rho_1^2*h1*T1-.7960990759e42*Rho_1*T1-.9514682791e44*Rho_1*h3
+.5269697128e37*T1^3*Rho_2*v[2]+.1660169992e43*T1*Rho_3
-.3834273945e40*T1^2*Rho_3-.1227215834e31*Rho_1*T1*Rho_3*uv[3]
-.1227215834e31*Rho_1*T1*Rho_2*uv[2]+.2454431668e31*Rho_1*T1
*v[6]*h3-.1227215834e31*Rho_1*T1*uv[6]*h3+.2634848564e37
*T1^3*uv[2]+.5269697128e37*T1^3*h2+.2454431668e31*Rho_1*T1
*v[5]*h2+.2454431668e31*Rho_1*T1*Rho_3*v[3]-.1227215834e31
*Rho_1*T1*uv[5]*h2-.4908863335e31*Rho_1*T1*Rho_2*h2
+.2634848564e37*T1^3*uv[3]+.2454431668e31*Rho_1
*T1*Rho_2*v[2]-.4908863335e31*Rho_1*T1*Rho_3*h3
+.1948785054e37*T1^3*Rho_3-.5269697128e37*T1^3*v[3]-.5115468357e28
*Rho_1*T1^3*Rho_3*h3-.1278867089e28*Rho_1
*T1^3*Rho_3*uv[3]+.2557734178e28*Rho_1*T1^3*v[6]*h3
-.1278867089e28*Rho_1*T1^3*uv[6]*h3+.2557734178e28*Rho_1*T1^3*Rho_2
*v[2]-.1278867089e28*Rho_1*T1^3*Rho_2*uv[2]
+.1278867089e28*Rho_1*T1^3*uv[3]+.9703178359e44*v[1]
-.4851589181e44*uv[2]-.9703178359e44*h1-.9703178362e44*h2
+.3408067676e40*Rho_1*T1^2*uv[1]-.4851589179e44*uv[1]
+.5269697128e37*T1^3*Rho_3*v[3]-.2634848564e37*T1^3*Rho_2
*uv[2]-.4851589181e44*uv[3]+.1000979983e41*Rho_1*h1*T1^2
-.2634848564e37*T1^3*uv[5]*h2-.2152726390e43*Rho_1*T1*v[2]
+.2152726390e43*Rho_1*T1*h2+.1076363195e43
*Rho_1*T1*uv[2]-.1168264303e43*Rho_1*uv[1]*T1
-.1776050078e40*Rho_1*T1^2*uv[3]+.3552100155e40*Rho_1*T1^2*v[3]
-.3552100155e40*Rho_1*T1^2*h2+.9703178362e44*v[2]
-.1776050078e40*Rho_1*T1^2*uv[2]
-.2152726390e43*Rho_1*T1*v[3]+.1076363195e43
*Rho_1*T1*uv[3]+.2152726390e43*Rho_1*T1*h3
-.1325042662e32*Rho_1*Rho_3*T3+.2336528606e43*Rho_1*v[1]*T1
+.5269697128e37*Rho_1*T1^3*v[1]+.5269697128e37*Rho_1^2
*h1*T1^3-.1411545126e42*v[4]*Rho_1^2*h1-.2477683118e43*Rho_1*h1*T1

```
-.3552100155e40*Rho_1*T1^2*h3-.6431184028e29*Rho_1*T1^2*Rho_3
+.5184117754e40*T1^2*Rho_3*uv[3]+.2073647102e41*T1^2*Rho_3*h3
+.5184117754e40*T1^2*uv[6]*h3-.1036823551e41*T1^2*v[6]*h3
-.3408067677e40*T1^2*uv[2]-.6816135353e40*T1^2*h2
+.6816135353e40*T1^2*v[2]+.5184117754e40*T1^2*Rho_2*uv[2]
-.1036823551e41*T1^2*Rho_2*v[2]+.2073647102e41*T1^2*Rho_2*h2
+.5184117754e40*T1^2*uv[5]*h2-.1036823551e41*T1^2*Rho_3*v[3]
+.9847447157e27*Rho_3*T3*T1^2-.1053939426e38*Rho_1
*h1*T1^3+.2520672892e40*T1^2-.1036823551e41*T1^2*v[5]*h2
-.6816135353e40*T1^2*h3+.6816135353e40*T1^2*v[3]
-.3408067677e40*T1^2*uv[3]-.2634848564e37*T1^3*Rho_3
*uv[3]-.1053939426e38*T1^3*Rho_3*h3-.7057725629e41*uv[7]*Rho_1^2*h1
-.2634848564e37*T1^3*uv[6]*h3+.5269697128e37*v[4]*Rho_1*h1*T1^2
+.1168264303e43*uv[1]*T1+.2634848564e37*uv[7]*Rho_1*h1*T1^2
+.1313601052e40*Rho_1*T1^2-.3233183011e30
*Rho_3*T3*T1-.2244627498e43*T1*Rho_3*uv[3]
-.8978509993e43*T1*Rho_3*h3+.3552100155e40*Rho_1*T1^2*v[2]
-.2244627498e43*T1*uv[6]*h3+.4489254996e43*T1*v[6]
*h3-.2244627498e43*T1*Rho_2*uv[2]+.4489254996e43*T1*Rho_2*v[2]
-.8978509993e43*T1*Rho_2*h2-.2244627498e43*T1*uv[5]*h2
+.4489254996e43*T1*Rho_3*v[3]+.4489254996e43
*T1*v[5]*h2+.9076726107e30*Rho_1*T1*Rho_3+.9458748800e27
*Rho_1*T1^3*Rho_3+.1278867089e28*Rho_1*T1^3*uv[2]
-.2557734178e28*Rho_1*T1^3*v[2]+.3518621880e44*Rho_1-.7106951249e44
*Rho_3+.1168264303e43*T1*uv[2]+.2336528606e43*T1*h2
-.2336528606e43*T1*v[2])/(-.8065170805e26-.1845976905e22*T1^2
+.8065170805e26*Rho_1-.1773279721e25*Rho_1*T1
+.1845976905e22*Rho_1*T1^2+.1922271048e19*Rho_1*T1^3
+.1773279721e25*T1-.1922271048e19*T1^3)
```

State feedback control law for input , 2

```
.9247999999e-8*(5000000000.*v[5]*h2+5000000000.*Rho_3*v[3]
-2500000000.*uv[5]*h2-.1000000000e11*Rho_2*h2+5000000000.*h3
-5000000000.*v[2]+2500000000.*uv[2]+5000000000.*h2+5000000000.*Rho_2
*v[2]-2500000000.*Rho_2*uv[2]+5000000000.*v[6]
*h3+1849048443.*Rho_3-2500000000.*uv[6]*h3-.1000000000e11*Rho_3*h3
-5000000000.*v[3]-1849048443.-2500000000.*Rho_3*uv[3]
+2500000000.*uv[3])/(Rho_1-1.)
```

State feedback control law for input , 3

```
.9247999999e-8*(5000000000.*v[6]*h3+1849048443.*Rho_3
-2500000000.*uv[6]*h3-.1000000000e11*Rho_3*h3+5000000000.*Rho_3*v[3]
-1849048443.-2500000000.*Rho_3*uv[3]+5000000000.*h3-5000000000.*v[3]
+2500000000.*uv[3])/(-1.+Rho_2)
```

State feedback control law for input , 4

```
.1775510273e-15*(.2150563935e40*T1^2*v[1]+.3782657080e44*v[3]
-.2150563935e40*T1^2*h1+.1818045509e37*T1^3*h1
-.1075281967e40*T1^2*uv[1]+.9090227545e36*T1^3*uv[1]+.1818045509e37
*T1^3*v[5]*h2-.1818045509e37*T1^3*v[1]+.6287369638e45*Rho_2*h2
-.3143684819e45*v[5]*h2+.6394335443e27*Rho_1*T1^3*h3-
.2364687199e27*Rho_1*T1^3-.2595066676e42*T1-.3782657080e44*h3-
.7351797532e39*uv[7]*Rho_1*h1*T1+.2051310565e42
*uv[7]*Rho_1*h1-.6723308434e36*T1^3-.9090227545e36*Rho_1*T1^3
*uv[1]-.6394335443e27*Rho_1*T1^3*v[3]+.1571842410e45*uv[5]*h2
+.1571842410e45*Rho_2*uv[2]-.3143684819e45*Rho_2*v[2]
+.1571842410e45*uv[6]*h3-.7017303102e42*v[1]*T1+
.7017303102e42*h1*T1+.4102621131e42*v[4]*Rho_1*h1
-.1470359506e40*v[4]*Rho_1*h1*T1-.2150563935e40*Rho_1*T1^2*v[1]
+.6287369638e45*Rho_3*h3-.3143684819e45*v[6]*h3
+.6394335443e27*Rho_1*T1^3*h2+.1818045509e37*T1^3*v[6]*h3
+.1398863237e44+.3851203852e32*Rho_3*T3+.2765419111e45*Rho_1*v[3]
-.3636091017e37*T1^3*Rho_2*h2+.1571842410e45*Rho_3*uv[3]
-.3143684819e45*Rho_3*v[3]+.6394335443e27*Rho_1*T1^3
*v[5]*h2+.6394335443e27*Rho_1*T1^3*Rho_3*v[3]-.3197167722e27
*Rho_1*T1^3*uv[5]*h2-.1278867089e28*Rho_1*T1^3*Rho_2*h2
-.3696150840e30*Rho_1*T1^2*v[6]*h3
-.3696150840e30*Rho_1*T1^2*Rho_2*v[2]
+.1848075420e30*Rho_1*T1^2*Rho_2*uv[2]-.3696150840e30*Rho_1
*T1^2*Rho_3*v[3]+.1848075420e30*Rho_1*T1^2*uv[5]*h2
+.7392301680e30*Rho_1*T1^2*Rho_2*h2+.7017303104e42*T1*h3
+.3508651552e42*T1*uv[3]-.7017303104e42*T1*v[3]-
.3696150840e30*Rho_1*T1^2*v[5]*h2+.1848075420e30*Rho_1
*T1^2*uv[6]*h3+.7392301680e30*Rho_1*T1^2*Rho_3*h3+.1848075420e30
*Rho_1*T1^2*Rho_3*uv[3]-.2461861788e27*Rho_1*Rho_3*T3*T1^2
+.2063951788e30*Rho_1*Rho_3*T3*T1-.6587121408e36*uv[7]
*Rho_1^2*h1*T1^2-.1317424282e37*v[4]*Rho_1^2*h1*T1^2
+.1818045509e37*T1^3*h3-.1818045509e37*T1^3*v[2]
+.2765419111e45*Rho_1*v[2]+.4102621131e42*Rho_1^2*h1*T1
-.1470359506e40*Rho_1^2*h1*T1^2+.1470359506e40*v[4]
*Rho_1^2*h1*T1-.1382709556e45*Rho_1*uv[3]-.3782657079e44
*Rho_1*v[1]-.1382709556e45*Rho_1*uv[2]+.3782657079e44*Rho_1*h1
-.2765419111e45*Rho_1*h2+.1891328539e44*Rho_1*uv[1]
+.7351797532e39*uv[7]*Rho_1^2*h1*T1-.3665231127e42*Rho_1*T1
-.2765419111e45*Rho_1*h3+.1818045509e37*T1^3*Rho_2*v[2]
+.6260297803e42*T1*Rho_3-.1123699642e40*T1^2*Rho_3-.3566872585e31
*Rho_1*T1*Rho_3*uv[3]-.3566872585e31*Rho_1*T1*Rho_2*uv[2]
+.7133745170e31*Rho_1*T1*v[6]*h3-.3566872585e31*Rho_1*T1*uv[6]
*h3+.9090227543e36*T1^3*uv[2]+.1818045509e37*T1^3*h2
+.7133745170e31*Rho_1*T1*v[5]*h2+.7133745170e31*Rho_1*T1*Rho_3*v[3]
-.3566872585e31*Rho_1*T1*uv[5]*h2-.1426749034e32*Rho_1*T1*Rho_2*h2
+.9090227543e36*T1^3*uv[3]+.7133745170e31*Rho_1*T1*Rho_2*v[2]
```

```
-.1426749034e32*Rho_1*T1*Rho_3*h3+.6723308434e36*T1^3*Rho_3
-.1818045509e37*T1^3*v[3]-.1278867089e28*Rho_1*T1^3*Rho_3*h3
-.3197167722e27*Rho_1*T1^3*Rho_3*uv[3]+.6394335443e27*Rho_1
*T1^3*v[6]*h3-.3197167722e27*Rho_1*T1^3*uv[6]*h3
+.6394335443e27*Rho_1*T1^3*Rho_2*v[2]-.3197167722e27*Rho_1*T1^3
*Rho_2*uv[2]+.3197167722e27*Rho_1*T1^3*uv[3]+.3782657079e44*v[1]
-.1891328540e44*uv[2]-.3782657079e44*h1-.3782657080e44*h2
+.1075281967e40*Rho_1*T1^2*uv[1]-.1891328539e44*uv[1]
+.1818045509e37*T1^3*Rho_3*v[3]-.9090227543e36*T1^3*Rho_2
*uv[2]-.1891328540e44*uv[3]+.3620923441e40*Rho_1*h1*T1^2
-.9090227543e36*T1^3*uv[5]*h2-.9911127912e42*Rho_1*T1*v[2]
+.9911127912e42*Rho_1*T1*h2+.4955563956e42*Rho_1*T1*uv[2]
-.3508651551e42*Rho_1*uv[1]*T1-.4440125194e39*Rho_1*T1^2*uv[3]
+.8880250388e39*Rho_1*T1^2*v[3]-.8880250388e39*Rho_1*T1^2*h2
+.3782657080e44*v[2]-.4440125194e39*Rho_1*T1^2*uv[2]
-.9911127912e42*Rho_1*T1*v[3]+.4955563956e42*Rho_1*T1*uv[3]
+.9911127912e42*Rho_1*T1*h3-.3851203852e32*Rho_1*Rho_3*T3
+.7017303102e42*Rho_1*v[1]*T1+.1818045509e37*Rho_1*T1^3*v[1]
+.1317424282e37*Rho_1^2*h1*T1^3-.4102621131e42*v[4]*Rho_1^2*h1
-.1111992423e43*Rho_1*h1*T1-.8880250388e39*Rho_1*T1^2*h3
-.1366872391e30*Rho_1*T1^2*Rho_3+.1519294487e40*T1^2
*Rho_3*uv[3]+.6077177947e40*T1^2*Rho_3*h3+.1519294487e40*T1^2
*uv[6]*h3-.3038588973e40*T1^2*v[6]*h3-.1075281968e40*T1^2*uv[2]
-.2150563935e40*T1^2*h2+.2150563935e40*T1^2*v[2]+.1519294487e40
*T1^2*Rho_2*uv[2]-.3038588973e40*T1^2*Rho_2*v[2]
+.6077177947e40*T1^2*Rho_2*h2+.1519294487e40*T1^2*uv[5]*h2
-.3038588973e40*T1^2*Rho_3*v[3]+.2461861788e27*Rho_3*T3*T1^2
-.3135469790e37*Rho_1*h1*T1^3+.7952993791e39*T1^2
-.3038588973e40*T1^2*v[5]*h2-.2150563935e40*T1^2*h3
+.2150563935e40*T1^2*v[3]-.1075281968e40*T1^2*uv[3]-
.9090227543e36*T1^3*Rho_3*uv[3]-.3636091017e37*T1^3*Rho_3*h3
-.2051310565e42*uv[7]*Rho_1^2*h1-.9090227543e36*T1^3*uv[6]*h3
+.1317424282e37*v[4]*Rho_1*h1*T1^2+.3508651551e42*uv[1]*T1
+.6587121408e36*uv[7]*Rho_1*h1*T1^2+.3284002630e39*Rho_1*T1^2
-.2063951788e30*Rho_3*T3*T1-.8464215508e42*T1*Rho_3*uv[3]
-.3385686203e43*T1*Rho_3*h3+.8880250388e39*Rho_1*T1^2*v[2]
-.8464215508e42*T1*uv[6]*h3+.1692843102e43*T1*v[6]*h3
-.8464215508e42*T1*Rho_2*uv[2]+.1692843102e43*T1*Rho_2*v[2]
-.3385686203e43*T1*Rho_2*h2-.8464215508e42*T1*uv[5]*h2
+.1692843102e43*T1*Rho_3*v[3]+.1692843102e43*T1*v[5]*h2
+.2638128080e31*Rho_1*T1*Rho_3+.2364687199e27*Rho_1*T1^3*Rho_3+.
3197167722e27*Rho_1*T1^3*uv[2]-.6394335443e27*Rho_1*T1^3*v[2]
+.1022678780e45*Rho_1-.1162565104e45*Rho_3+.3508651552e42*T1*uv[2]
+.7017303104e42*T1*h2-.7017303104e42*T1*v[2])/(-.8065170805e26
-.1845976905e22*T1^2+.8065170805e26*Rho_1-.1773279721e25*Rho_1*T1
+.1845976905e22*Rho_1*T1^2+.1922271048e19*Rho_1*T1^3
+.1773279721e25*T1-.1922271048e19*T1^3)
```

State feedback control law for input , 5

```
-.3551020546e-19*(-.2703287198e22*Rho_2*h2+.1351643599e22*v[5]*h2
-.1351643599e22*h3*Rho_2+.6758217994e21*Rho_2^2*uv[2]
-.1351643599e22*Rho_2^2*v[2]-.1351643599e22*v[5]*h2*Rho_2
+.4998508984e21*Rho_2-.6758217994e21*uv[5]*h2
-.6758217994e21*Rho_2*uv[2]+.1351643599e22*Rho_2*v[2]
-.1351643599e22*Rho_1*v[3]-.1351643599e22*Rho_1*v[2]
+.6758217994e21*Rho_1*uv[3]+.6758217994e21*Rho_1*uv[2]
+.1351643599e22*Rho_1*h2+.7415508377e18*Rho_1*T1
+.1351643599e22*Rho_1*h3.7575227018e18*Rho_2*T2
+.1002611425e19*Rho_1*T1*Rho_3*uv[3]+.2005222850e19
*Rho_1*T1*Rho_2*uv[2]-.2005222850e19*Rho_1*T1*v[6]*h3
+.1002611425e19*Rho_1*T1*uv[6]*h3-.2005222850e19*Rho_1
*T1*v[5]*h2-.2005222850e19*Rho_1*T1*Rho_3*v[3]
+.1002611425e19*Rho_1*T1*uv[5]*h2+.6015668550e19*Rho_1
*T1*Rho_2*h2-.4010445700e19*Rho_1*T1*Rho_2*v[2]+.4010445700e19*Rho_1
*T1*Rho_3*h3+.4998508984e21*Rho_1*Rho_3-.1024206133e19*Rho_2
*T2*uv[6]*h3-.4096824530e19*Rho_2*T2*Rho_3*h3
+.2048412265e19*Rho_2*T2*Rho_3*v[3]+.1002611425e19*Rho_1*T1
*uv[3]*Rho_2+.2005222850e19*Rho_1*T1*h3*Rho_2+.7415508377e18
*Rho_1*T1*Rho_3*Rho_2-.2005222850e19*Rho_1*T1*v[3]*Rho_2
+.2005222850e19*Rho_1*T1*v[6]*h3*Rho_2-.1002611425e19*Rho_1
*T1*uv[6]*h3*Rho_2+.2005222850e19*Rho_1*T1*v[5]*h2*Rho_2
+.2005222850e19*Rho_1*T1*Rho_3*v[3]*Rho_2-.1002611425e19
*Rho_1*T1*Rho_3*uv[3]*Rho_2-.1002611425e19*Rho_1*T1
*uv[5]*h2*Rho_2-.4010445700e19*Rho_1*T1*Rho_3*h3*Rho_2
+.2005222850e19*Rho_1*T1*Rho_2^2*v[2]-.4998508984e21*Rho_3
*Rho_2-.4010445700e19*Rho_1*T1*Rho_2^2*h2-.1351643599e22
*v[6]*h3*Rho_2+.6758217994e21*Rho_3*uv[3]*Rho_2-.1351643599e22
*Rho_3*v[3]*Rho_2-.1002611425e19*Rho_1*T1*Rho_2^2*uv[2]
-.6758217994e21*uv[3]*Rho_2+.7575227018e18*Rho_2*T2*Rho_1
+.6758217994e21*uv[5]*h2*Rho_2+.6758217994e21*uv[6]*h3*Rho_2
+.2703287198e22*Rho_3*h3*Rho_2+.1351643599e22*v[3]*Rho_2
-.7415508377e18*Rho_2*Rho_1*T1-.2048412265e19*Rho_2*T2*h3*Rho_1
+.2048412265e19*Rho_2*T2*v[3]*Rho_1.1024206133e19*Rho_2*T2*uv[3]
*Rho_1+.1024206133e19*Rho_2*T2*Rho_3*uv[3]*Rho_1-.2048412265e19
*Rho_2*T2*v[6]*h3*Rho_1-.7575227018e18*Rho_3*Rho_2*T2*Rho_1
+.1024206133e19*Rho_2*T2*uv[6]*h3*Rho_1+.4096824530e19
*Rho_2*T2*Rho_3*h3*Rho_1-.2048412265e19*Rho_2*T2*Rho_3*v[3]*Rho_1
-.1024206133e19*Rho_2*T2*Rho_3*uv[3]+.2048412265e19*Rho_2*T2*h3
-.2048412265e19*Rho_2*T2*v[3]+.1024206133e19*Rho_2*T2*uv[3]
+.2048412265e19*Rho_2*T2*v[6]*h3+.2005222850e19*Rho_1*T1*v[2]
-.2005222850e19*Rho_1*T1*h2-.1002611425e19*Rho_1*T1*uv[2]
+.2005222850e19*Rho_1*T1*v[3]-.1002611425e19*Rho_1*T1*uv[3]
-.2005222850e19*Rho_1*T1*h3+.7575227018e18*Rho_3*Rho_2*T2
-.7415508377e18*Rho_1*T1*Rho_3-.2703287198e22*Rho_1*Rho_3*h3
-.4998508984e21*Rho_1+.1351643599e22*Rho_1*Rho_2*v[2]
```

```
+.2703287198e22*Rho_2^2*h2-.1351643599e22*Rho_1*Rho_2*h2
+.1351643599e22*Rho_1*Rho_3*v[3]-.6758217994e21*Rho_1*uv[6]*h3
+.1351643599e22*Rho_1*v[6]*h3-.6758217994e21*Rho_1*Rho_2*uv[2]
-.6758217994e21*Rho_1*Rho_3*uv[3])/(Rho_1-1.)/(-1.+Rho_2)
```

State feedback control law for input , 6

```
-.1775510273e-19*(-.2703287198e22*h3*Rho_2+.9997017967e21*Rho_2
+.1351643599e22*uv[6]*h3+.5406574395e22*Rho_3*h3
-.2703287198e22*v[6]*h3+.1556115912e19*Rho_3*T3
+.1351643599e22*Rho_3*uv[3]-.2703287198e22*Rho_3*v[3]
-.1515045404e19*Rho_2*T2-.2048412265e19*Rho_2*T2*uv[6]*h3
-.8193649060e19*Rho_2*T2*Rho_3*h3+.4096824530e19*Rho_2*T2*Rho_3*v[3]
-.1351643599e22*uv[3]*Rho_2+.2703287198e22*v[3]*Rho_2
-.2048412265e19*Rho_2*T2*Rho_3*uv[3]+.4096824530e19*Rho_2*T2*h3
-.4096824530e19*Rho_2*T2*v[3]+.2048412265e19*Rho_2*T2*uv[3]
+.4096824530e19*Rho_2*T2*v[6]*h3+.1515045404e19*Rho_3*Rho_2*T2
-.1556115912e19*Rho_3*T3*Rho_2-.9997017967e21*Rho_3)/(-1.+Rho_2)
```

xss := h1 = 2.71, h2 = 2.71, h3 = 2.71, Rho_1 = 1.44, Rho_2 = 1.49, Rho_3 = 1.54, T1 = 90.6, T2 = 129, T3 = 135

uss := Qf = 29.7, Qp1 = 23.7, Qp2 = 19.6, Qcw = 77.4681, ms2 = 6.276004, ms3 = 2.69646215

dss := Qp3 = 17.1

uncerss := Tapp = 5, Rho_f = 1.38, Tf = 66, Tcw = 27, Cp3 = 3410

uncerps := Tapp = 14, Rho_f = 1.30, Tf = 70, Tcw = 30, Cp3 = 3415

ys := [2.71, 2.71, 2.71, 90.6, 1.49, 1.54]

d := [Qp3 = 20.0]

ts := .02

kset := 10

Kp := [10, 10, 10, -500, 10, 10]

ti := [1, 1, 1, 1, 1, 1, 1]

uvp := [10, 0, 0, 0, 0, 0, 10, 0, 0]

Closed Loop Dynamics - t, x, u:

.04,

[.001172703, -.000393468, -.002751176, -.001210085, .000216334, -.000162529,

.00137495, $-.8*10^{-6}$, .0151332],

[-9.74084220, -.8472670, 1.2842917, -34.92561549, -2.722981476, -.837237290]

.08,

[.002090553, -.000107895, -.003413127, -.002080445, .000059321, -.000302023,

.00191479, $-.12*10^{-5}$, .0196578],

[-12.47158598, .4773668, 2.1438895, -18.92249694, -2.096309183, -.617451720]

.12,

[.002803509, -.000015651, -.003535793, -.002423963, $.8605*10^{-5}$, -.000338235,

.00301224, $-.12*10^{-5}$, .0205873],

[-15.78713206, .9098480, 2.4208834, -10.37428527, -1.888943183, -.545953539]

.16,

[.003217390, .000013664, -.003486248, -.002359573, $-.7514*10^{-5}$, -.000341162,

.00405574, $-.12*10^{-5}$, .0203658],

[-18.16577190, 1.0376677, 2.5082545, -5.48279577, -1.830890834, -.522746342]

.20,

[.003418406, .000022518, -.003383920, -.002029679, -.000012381, -.000333579,

.00487843, $-.12*10^{-5}$, .0197888],

[-19.68829053, 1.0587317, 2.5339766, -2.53836633, -1.828305242, -.515261778]

Integral Time Weighted Absolute Errors:

$$[.0000628065560, .13745212 \times 10^{-5}, .0000746989564, .000077460660,$$

$$.7557420 \times 10^{-6}, .69836688 \times 10^{-5}]$$

>

APPENDIX F

LINEARIZED MODEL FOR TRIPLE EFFECTS EVAPORATOR

F.1 LINEARIZED MODEL FOR TRIPLE EFFECTS EVAPORATOR

The linearized model of the triple effect evaporator was computed using MAPLE V.3 by linearizing the nonlinear state equations.

$$\dot{\mathbf{x}} = \Delta \mathbf{x} + \Psi \mathbf{u} + \Gamma \mathbf{d} \quad \text{(H.1)}$$

where

$$\Delta = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -0.00782 & 0 & 0 \\ 0 & 0 & 0 & -0.138 & 0.166 & 0 & -0.00191 & 0.00191 & 0 \\ 0 & 0 & 0 & 0 & -0.166 & 0.155 & 0 & -0.00191 & 0.00177 \\ -0.00780 & 0 & 0 & -0.361 & 0 & 0 & 0.00127 & 0 & 0 \\ 0 & -0.00723 & 0 & 0.403 & -0.329 & 0 & 0.000396 & -0.000346 & 0 \\ 0 & 0 & -0.00305 & 0 & 0.346 & -0.299 & 0 & 0.000381 & -0.000353 \\ -0.190 \times 10^{-5} & 0 & 0 & 23.8 & 0 & -26.8 & -1.19 & 0 & -0.306 \\ 0 & -0.212 \times 10^{-6} & 0 & -28.3 & 21.8 & 0 & 0.0699 & -0.0611 & 0 \\ 0 & 0 & 0.0499 & 0 & -22.1 & 18.9 & 0 & 0.0619 & -0.0583 \end{bmatrix}$$

$$\Psi = \begin{bmatrix} 0.0433 & -0.0433 & 0 & -0.00394 & 0 & 0 \\ 0 & 0.0349 & -0.0306 & 0 & -0.0417 & 0 \\ 0 & 0 & 0.0306 & 0 & 0 & -0.0417 \\ -0.000958 & 0 & 0 & 0.000640 & 0 & 0 \\ 0 & 0.000716 & -0.00228 & 0 & 0.00753 & 0 \\ 0 & 0 & 0.00171 & 0 & 0 & 0.00830 \\ -0.364 & 1.446 & 0 & -0.603 & 7.19 & 7.19 \\ 0 & -1.72 & 1.66 & 0 & 1.33 & 0 \\ 0 & 0 & -1.68 & 0 & 0 & 1.35 \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} 0 \\ 0 \\ -0.0293 \\ 0 \\ 0 \\ -0.00279 \\ -2.42 \\ 0 \\ 1.70 \end{bmatrix}$$

The eigenvalues of matrix Δ is $\begin{bmatrix} -1.23 \\ -0.330 + 0.0350I \\ -0.0330 - 0.0350I \\ -0.243 \\ -0.0749 + 0.0603I \\ -0.0749 - 0.0603I \\ 0.824 \times 10^{-8} \\ -0.0165 \\ -0.00496 \end{bmatrix}$.

APPENDIX G

LINEAR CONTROL STRATEGIES FOR LIQUOR BURNING PROCESS

This appendix includes the part of the memo written by Mr. Graham Le Page containing the simulation of the liquor burning process using different linear control strategies. The details of the analysis are omitted due to confidentiality but the main results are included for comparison with those developed in this thesis. The motivations of providing this report are to allow the readers to understand the investigation performed on the liquor burning process by Alcoa in the past and also to compare the results obtained in this thesis with the one obtained in the previous study.

Note: For copyright reasons Appendix G (ppG2-G14 of this thesis) has not been reproduced.

(Co-ordinator, ADT Project (Retrospective), Curtin University of Technology, 13.1.03)