INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600
ON THE ROBUSTNESS OF A SYSTEMATIC METHODOLOGY OF FUZZY-LOGIC MODELING

by

WAEL WILLIAM MELEK

A thesis submitted in conformity with the requirements for the degree of Masters of Applied Science
Graduate Department of Mechanical and Industrial Engineering
University of Toronto

© Copyright by W. William Melek 1998
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.
To
My parents, Sameh, and my Aunt in Egypt

To
My family in Canada

To
Dr. M. R. Emami
On the Robustness of A Systematic Methodology of Fuzzy-Logic Modeling
Masters of Applied Science
Wael William Melek
Department of Mechanical and Industrial Engineering
University of Toronto

ABSTRACT

This thesis presents a detailed investigation of the robustness of fuzzy-logic models. The key problem in fuzzy model generation is to find a suitable structure based upon a good model can be found. However, regardless of the modeling approach, three as yet unsolved problems will always arise after developing the model: (i) no guarantee that the developed model has a satisfactory continuous behaviour; (ii) the model may show a "poor" performance if the data used to build the model is contaminated with noise; and (iii) the model may show a "poor" generalization capability if used to predict data that has not been used in the model generation. Each of the above problems is investigated in the context of the fuzzy-logic modeling approach.

To perform this investigation, a definition of the model robustness is introduced and the systematic methodology of fuzzy-logic modeling (Emami et al., 1996a) is reviewed. The goal is to improve this methodology with regard to the above-mentioned problems. First, the continuity behaviour of the fuzzy reasoning mechanism is investigated. As a result, some conditions on the reasoning parameters are introduced to adjust the sensitivity of the inference mechanism to account for any deviation in the inputs. Second, an improved noise rejection clustering algorithm is introduced. The improved algorithm overcomes the problems of the traditional clustering algorithms and introduces a new criterion to cut off the noise in the data. Third, the generalization capability of the fuzzy logic model is investigated and some conditions on the input-output membership functions are introduced to improve the model behaviour when extended to a new set of data that has not been used in the model construction. Also, two different evaluation criteria are studied to validate the developed fuzzy-logic models.

Finally, the results are applied to two real applications. The first application is a well-known gas furnace plant. The second application is a 4 d.o.f robot manipulator developed in the Robotics and Automation Laboratory at the University of Toronto.
ACKNOWLEDGMENTS

I wish to thank my supervisor Professor A. A. Goldenberg for his continuous support throughout the program. His valuable guidance and concern helped my research constantly progress. I owe him for giving me the chance to work under his supervision.

I would also like to extend my thanks to Dr. M. R. Emami for helping me to obtain a better understanding of fuzzy set theory. I am grateful for his continuous support throughout my program. His valuable insights and comments helped me to complete this work at a high level.

I would like to thank Mr. David Pitts, Mr. Lin Sheng, Ms. Liz Montoro and all my colleagues in the Robotics and Automation Laboratory at the University of Toronto for their support and for providing a friendly environment.

Finally, I would like to thank my family, to whom this thesis is dedicated. my parents and my brother for their everlasting patience and support. I would also like to thank my family here in Canada for their continuous support and guidance.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>III</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>IV</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>V</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>VII</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>XI</td>
</tr>
<tr>
<td>NOTATIONS</td>
<td>XII</td>
</tr>
<tr>
<td><strong>CHAPTER 1: INTRODUCTION</strong></td>
<td>1</td>
</tr>
<tr>
<td>1.1: MOTIVATION</td>
<td>1</td>
</tr>
<tr>
<td>1.2: FUZZY-LOGIC MODELING</td>
<td>2</td>
</tr>
<tr>
<td>1.3: A REVIEW OF THE SYSTEMATIC METHODOLOGY OF FUZZY-LOGIC MODELING</td>
<td>3</td>
</tr>
<tr>
<td>1.3.1: Reasoning Mechanism</td>
<td>5</td>
</tr>
<tr>
<td>1.3.2: Fuzzy Structure Identification</td>
<td>6</td>
</tr>
<tr>
<td>1.4: DEFINITION OF THE MODEL ROBUSTNESS</td>
<td>9</td>
</tr>
<tr>
<td>1.5: LITERATURE REVIEW</td>
<td>10</td>
</tr>
<tr>
<td>1.5.1: Robustness of the Fuzzy Inference Mechanism</td>
<td>10</td>
</tr>
<tr>
<td>1.5.2: Robustness of the Fuzzy Clustering Algorithm</td>
<td>10</td>
</tr>
<tr>
<td>1.5.3: Generalization Capability of Fuzzy-Logic Models</td>
<td>13</td>
</tr>
<tr>
<td>1.6: THESIS CONTENTS</td>
<td>15</td>
</tr>
<tr>
<td>1.7: CONTRIBUTIONS</td>
<td>16</td>
</tr>
<tr>
<td><strong>CHAPTER 2: ROBUSTNESS OF THE FUZZY INference MECHANISM</strong></td>
<td>18</td>
</tr>
<tr>
<td>2.1: INTRODUCTION</td>
<td>18</td>
</tr>
<tr>
<td>2.2: A UNIFIED PARAMETERIZED FORMULATION OF THE REASONING PROCESS</td>
<td>19</td>
</tr>
<tr>
<td>2.3: THE MODULUS OF CONTINUITY AND THE DEVIATION OF THE INPUT MEMBERSHIP FUNCTIONS</td>
<td>21</td>
</tr>
<tr>
<td>2.4: MARGINAL CONTINUITY OF THE FUZZY OUTPUT</td>
<td>24</td>
</tr>
<tr>
<td>2.4.1 Single-Input Single-Output Systems</td>
<td>24</td>
</tr>
<tr>
<td>2.4.2 Multi-Input Single-Output Systems</td>
<td>30</td>
</tr>
<tr>
<td>2.5: MARGINAL CONTINUITY OF THE CRISP OUTPUT</td>
<td>34</td>
</tr>
<tr>
<td>2.6: ASSIGNMENT OF PARAMETERS</td>
<td>34</td>
</tr>
<tr>
<td>2.7: CASE STUDY</td>
<td>35</td>
</tr>
<tr>
<td>2.8: CONCLUSIONS</td>
<td>48</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

## CHAPTER 3: A ROBUST FUZZY CLUSTERING ALGORITHM .......................... 49
1. INTRODUCTION ............................................................................. 49
2. THE FUZZY C-MEANS CLUSTERING ALGORITHM .................. 50
3. A NOISE-REJECTION CLUSTERING ALGORITHM ............... 52
4. A COMBINED ALGORITHM WITH A NOISE-REJECTION CAPABILITY .... 54
5. EXAMPLES ................................................................................. 55
6. CONCLUSIONS ............................................................................. 63

## CHAPTER 4: THE GENERALIZATION CAPABILITY OF FUZZY-LOGIC MODELS ........................................................................................................ 64
1. INTRODUCTION ............................................................................. 64
2. CONDITIONS ON THE DATA ...................................................... 65
3. CONSTRUCTION OF THE INPUT-OUTPUT MEMBERSHIP FUNCTIONS .... 66
4. THE TUNING PROCESS OF THE MODEL TRAINING SET .......... 70
5. MODEL VALIDATION CRITERIA ..................................................... 71
   5.1: Description of the Prediction Error .................................. 72
   5.2: The Parsimony Principle .................................................. 72
   5.3: Akaike’s Information Criterion ........................................ 74
   5.4: Occam Razor .................................................................. 75
   5.5: The Relation Between AIC and Occam’s Razor and their Applicability to Fuzzy-Logic Modeling .................. 77
6. EXAMPLE ..................................................................................... 78
7. CONCLUSIONS ............................................................................. 89

## CHAPTER 5: APPLICATIONS .................................................................. 90
1. INTRODUCTION ............................................................................. 90
2. A GAS FURNACE PLANT APPLICATION .................................. 90
3. APPLICATION TO ROBOT MANIPULATORS ......................... 99
4. CONCLUSIONS ............................................................................. 122

## CHAPTER 6: CONCLUSIONS AND FUTURE RESEARCH ............................. 123
1. CONCLUSIONS ............................................................................. 123
2. FUTURE RESEARCH ................................................................ 125

## REFERENCES .................................................................................... 126

## APPENDIX 1 .................................................................................... 129
# LIST OF FIGURES

1.1 Fuzzy input-output rules of the MISO system ......................................................... 3
1.2 A flow chart of the fuzzy system modeling ............................................................. 4
2.1 The modulus of continuity of the function g ............................................................ 21
2.2 The deviation of the membership functions ............................................................ 23
2.3 The deviation of the fuzzy output and the illustration of the function e .................. 24
2.4 Sugeno-Yasukawa deterministic function ............................................................. 35
2.5 Input-output rules of the original model (Emami, 1997) ........................................ 36
2.6 A comparison between the function and the original model ................................... 37
2.7 Variation of the input deviation with both p and q .................................................. 38
2.8 Selection of the lower bounds of p and q ............................................................... 38
2.9 Variation of the deviation of the crisp input with parameter α .................................. 39
2.10 The tuning process of the robust model training set ............................................. 39
2.11 A comparison between the function and the robust model ................................... 40
2.12 The fuzzy input-output rules of the robust model ................................................. 41
2.13 A comparison between the performance of the original model and the robust model for the testing set 1 .................................................................................. 43
2.14 A comparison between the performance of the original model and the robust model for the testing set 2 .................................................................................. 44
2.15 A comparison between the performance of the original model and the robust model for the testing set 3 .................................................................................. 45
2.16 A comparison between the performance of the original model and the robust model for the testing set 4 .................................................................................. 46
2.17 A comparison between the performance of the original model and the robust model for the testing set 5 .................................................................................. 47
3.1 (a) Krishnapuram example, (b) the fuzzy c-means clustering algorithm for example 3.5.1 ......................................................................................................................... 55
3.2 (b) Selection of the optimum number of clusters, (d) the noise-rejection criterion for Krishnapuram example ............................................................................................. 56
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3</td>
<td>A comparison between the NPCM and the combined algorithm for Krishnapuram example</td>
<td>57</td>
</tr>
<tr>
<td>3.4</td>
<td>(a) Data set 3, (b) trace of the total scatter matrix vs. $m$ for data set 3</td>
<td>58</td>
</tr>
<tr>
<td>3.5</td>
<td>(a) Selection of the optimum number of clusters, (b) The noise-rejection criterion for data set 3</td>
<td>58</td>
</tr>
<tr>
<td>3.6</td>
<td>A comparison between the NPCM and the combined algorithm for data set 3</td>
<td>59</td>
</tr>
<tr>
<td>3.7</td>
<td>(a) Data set A, (b) the choice of the weight exponent for data set A</td>
<td>60</td>
</tr>
<tr>
<td>3.8</td>
<td>(a) The optimum number of clusters, (b) The noise-rejection criterion for data set A</td>
<td>60</td>
</tr>
<tr>
<td>3.9</td>
<td>A comparison between the NPCM and the combined algorithm for data set A</td>
<td>61</td>
</tr>
<tr>
<td>3.10</td>
<td>(a) Sugeno-Yasukawa static function, (b) trace of the total scatter matrix vs. the weighting exponent</td>
<td>62</td>
</tr>
<tr>
<td>3.11</td>
<td>(a) The optimum number of cluster, (b) the noise-rejection criterion for Sugeno-Yasukawa example</td>
<td>62</td>
</tr>
<tr>
<td>4.1</td>
<td>The proposed method for the input membership functions construction</td>
<td>69</td>
</tr>
<tr>
<td>4.2</td>
<td>An algorithm for the constraints on the input membership functions</td>
<td>69</td>
</tr>
<tr>
<td>4.3</td>
<td>Applying the new algorithm while tuning the input-output membership functions</td>
<td>70</td>
</tr>
<tr>
<td>4.4</td>
<td>Occam razor and model comparison</td>
<td>76</td>
</tr>
<tr>
<td>4.5</td>
<td>The tuning process of the model training set</td>
<td>81</td>
</tr>
<tr>
<td>4.6</td>
<td>variation of the reasoning parameters during the tuning process of the model training set</td>
<td>81</td>
</tr>
<tr>
<td>4.7</td>
<td>A comparison between the function and the developed robust model for the training set</td>
<td>82</td>
</tr>
<tr>
<td>4.8</td>
<td>The input-output rules of the developed robust model</td>
<td>82</td>
</tr>
<tr>
<td>4.9</td>
<td>A comparison between the robust model and the original model for the testing set 1 (outer limit characteristics)</td>
<td>84</td>
</tr>
<tr>
<td>4.10</td>
<td>A comparison between the robust model and the original model for the testing set 2 (outer limit characteristics)</td>
<td>85</td>
</tr>
<tr>
<td>4.11</td>
<td>A comparison between the robust model and the original model for the testing set 3</td>
<td>86</td>
</tr>
<tr>
<td>Figure Number</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>4.12</td>
<td>A comparison between the robust model and the original model for the testing set 4</td>
<td></td>
</tr>
<tr>
<td>4.13</td>
<td>A comparison between the robust model and the original model for the testing set 5</td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>Selection of the weight exponent for the gas furnace example</td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>Identification of the optimum number of clusters for the gas furnace example</td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>Selection of the lower bounds for the parameters $p$ and $q$</td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>The range of the parameter $\alpha$ for the gas furnace example</td>
<td></td>
</tr>
<tr>
<td>5.5</td>
<td>The tuning process of the training set of the gas furnace model</td>
<td></td>
</tr>
<tr>
<td>5.6</td>
<td>Variation of the reasoning parameters during the tuning process of the model training set</td>
<td></td>
</tr>
<tr>
<td>5.7</td>
<td>The fuzzy input-output rules of the robust model for the gas furnace example</td>
<td></td>
</tr>
<tr>
<td>5.8</td>
<td>A comparison between the original model and the robust model for the training set</td>
<td></td>
</tr>
<tr>
<td>5.9</td>
<td>A comparison between the original model and the robust model for the testing set 1</td>
<td></td>
</tr>
<tr>
<td>5.10</td>
<td>A comparison between the original model and the robust model for the testing set 2</td>
<td></td>
</tr>
<tr>
<td>5.11</td>
<td>The fuzziness and cluster validity for the IRIS joints</td>
<td></td>
</tr>
<tr>
<td>5.12</td>
<td>The criterion for the selection of the lower bounds for $p$ and $q$ for each joint</td>
<td></td>
</tr>
<tr>
<td>5.13</td>
<td>Selection of the range for parameter $\alpha$</td>
<td></td>
</tr>
<tr>
<td>5.14</td>
<td>The tuning process of the training set for each joint</td>
<td></td>
</tr>
<tr>
<td>5.15</td>
<td>Identification of the tuned reasoning parameters from the tuning process</td>
<td></td>
</tr>
<tr>
<td>5.16</td>
<td>The fuzzy input-output rules of the robust model for joint 1</td>
<td></td>
</tr>
<tr>
<td>5.17</td>
<td>The fuzzy input-output rules of the robust model for joint 2</td>
<td></td>
</tr>
<tr>
<td>5.18</td>
<td>The fuzzy input-output rules of the robust model for joint 3</td>
<td></td>
</tr>
<tr>
<td>5.19</td>
<td>The fuzzy input-output rules of the robust model for joint 4</td>
<td></td>
</tr>
<tr>
<td>5.20</td>
<td>A comparison between the original model and the robust model for the training set of joint 1</td>
<td></td>
</tr>
<tr>
<td>5.21</td>
<td>A comparison between the original model and the robust model for the testing set 1</td>
<td></td>
</tr>
</tbody>
</table>
LIST OF FIGURES

5.22 A comparison between the original model and the robust model for the testing set 2 .......................................................... 115
5.23 A comparison between the original model and the robust model for the training set of joint 2 .......................................................... 116
5.24 A comparison between the original model and the robust model for the testing set 3 .......................................................... 117
5.25 A comparison between the original model and the robust model for the testing set 4 .......................................................... 117
5.26 A comparison between the original model and the robust model for the training set of joint 3 .......................................................... 118
5.27 A comparison between the original model and the robust model for the testing set 5 .......................................................... 119
5.28 A comparison between the original model and the robust model for the testing set 6 .......................................................... 119
5.29 A comparison between the original model and the robust model for the training set of joint 4 .......................................................... 120
5.30 A comparison between the original model and the robust model for the testing set 7 .......................................................... 121
5.31 A comparison between the original model and the robust model for the testing set 8 .......................................................... 121
LIST OF TABLES

2.1 The comparison between the performance of the robust inference mechanism model and the original model for Sugeno-Yasukawa example ................................................. 42

4.1 The comparison between the performance of the robust model and the original model for Sugeno-Yasukawa example ........................................................................... 80

5.1 The comparison between the performance of the robust model and the original model for the gas furnace plant .................................................................................... 92

5.2 The inputs and outputs of the IRIS robot arm ........................................................................ 100

5.3 The optimum model parameters for each joint of the IRIS arm ........................................ 101

5.4 Identification of the parameters of the robust model for the IRIS arm ............................ 101

5.5 Identification of the reasoning parameters for each joint during the tuning process ................................................................................................................................. 102

5.6 The comparison between the performance of the robust model and the original model for the IRIS arm ............................................................................................... 102
a.b.c  Arguments of a parametric operator, membership functions of A.B.C
B_{ij}  Fuzzy set for the j^{th} input in the i^{th} rule
C   Number of clusters
C_k  Trapezoidal membership function vertices
D_{ij}  Fuzzy set for the j^{th} output in the i^{th} rule
\overline{D}  Negation of the fuzzy set D
D   Matrix of dissimilarities
d_{c}\^{2}  Cutoff distance of the noise points
e  Maximum deviation of the fuzzy output
E  The output of the fuzzy set from the unified reasoning
E(.)  Expectation operator
F_{M}  The output fuzzy set from Mamdani's reasoning
F_{L}  The output fuzzy set from Logical reasoning
h  Maximum value of the input deviation
h_{F}  Maximum deviation of the input membership grades
h_{t}  Deviation of the degree of firing value
J_{m}  Fuzzy clusters objective function
m  Weighting exponent
n  Total number of rules, order of the system
N  Number of data points
p  Fuzzy rule implication parameter
P(.)  Plausibility of a proposition
q  Fuzzy antecedents aggregation parameter, system state
\dot{q}  System state velocity
\ddot{q}  System state acceleration
r  Total number of input variables
R_{y_{y}}  Autocorrelation factor
s  Dimension of a data feature
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_{cs}$</td>
<td>Cluster validity index</td>
</tr>
<tr>
<td>$S$</td>
<td>T-conorm operator</td>
</tr>
<tr>
<td>$S_T$</td>
<td>Fuzzy total scatter matrix</td>
</tr>
<tr>
<td>$t$</td>
<td>Time</td>
</tr>
<tr>
<td>$t^{ij}$</td>
<td>Mutual ratio of the data</td>
</tr>
<tr>
<td>$T$</td>
<td>T-norm operator</td>
</tr>
<tr>
<td>$u_{ij}$</td>
<td>Membership grade of datum $j$ in cluster $i$</td>
</tr>
<tr>
<td>$U$</td>
<td>Membership matrix</td>
</tr>
<tr>
<td>$v$</td>
<td>Cluster center</td>
</tr>
<tr>
<td>$\overline{v}$</td>
<td>Fuzzy mean value</td>
</tr>
<tr>
<td>$V_N$</td>
<td>Loss function</td>
</tr>
<tr>
<td>$W_N$</td>
<td>Prediction error variance</td>
</tr>
<tr>
<td>$X$</td>
<td>Input universe of discourse for input variable $x$, system</td>
</tr>
<tr>
<td>$\hat{y}$</td>
<td>Defuzzified output</td>
</tr>
<tr>
<td>$\overline{\hat{y}}$</td>
<td>Deviation of the defuzzified output</td>
</tr>
<tr>
<td>$Y$</td>
<td>Output universe of discourse for output variable $y$, system</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Vector of coefficients, vector of parameters</td>
</tr>
<tr>
<td>$\theta_0$</td>
<td>True parameter vector</td>
</tr>
<tr>
<td>$\hat{\theta}$</td>
<td>Estimated parameter vector</td>
</tr>
<tr>
<td>$\tau_i$</td>
<td>The degree of firing of rule $i$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Defuzzification parameter</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Inference compromisation parameter</td>
</tr>
<tr>
<td>$\Pi_j$</td>
<td>An index for the choice of the significant input parameters</td>
</tr>
<tr>
<td>$\Gamma_j$</td>
<td>Entire range of an input $x_j$</td>
</tr>
<tr>
<td>$\delta$</td>
<td>An estimator for trimming observations with large residuals</td>
</tr>
<tr>
<td>$\gamma_g$</td>
<td>Modulus of continuity of a function $g$</td>
</tr>
<tr>
<td>$\gamma_f$</td>
<td>Modulus of continuity of the fuzzy output</td>
</tr>
<tr>
<td>$\gamma_c$</td>
<td>Modulus of continuity of the crisp output</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Input membership function deviation</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>$\hat{\delta}_i$</td>
<td>Deviation of the T-norm for antecedents aggregation in the $i^{th}$ rule</td>
</tr>
<tr>
<td>$\nu_i$</td>
<td>Resolution parameter that depends on the data partitions</td>
</tr>
<tr>
<td>$\chi_a$</td>
<td>Chi-square distribution value</td>
</tr>
<tr>
<td>$\hat{\lambda}$</td>
<td>Percentage of noise points in the data</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>Sum of columns of the matrix of dissimilarities</td>
</tr>
<tr>
<td>$\hat{\mu}$</td>
<td>Estimated mean value of the output</td>
</tr>
<tr>
<td>$\varepsilon$</td>
<td>Prediction error for the estimated output</td>
</tr>
<tr>
<td>$\Lambda$</td>
<td>Variance</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>Penalizing term that depends on the data</td>
</tr>
<tr>
<td>$\psi_i$</td>
<td>Proposition $i$</td>
</tr>
</tbody>
</table>
CHAPTER  INTRODUCTION

1.1 Motivation

Modeling of nonlinear systems can be approached in a variety of ways. Regardless of the modeling approach adopted, three important issues will always arise. First, there is no guarantee that the developed model will illustrate a continuous, and to a certain extent, smooth behaviour, assuming that the real physical system has a continuous behaviour. In other words, we may experience unwanted "jumps" in the model output even for slight deviations in the input. Second, the model may show a "poor" performance if the training input-output data set is contaminated by noise and outliers. Finally, the developed model might not satisfactorily generalize the behaviour; that is the developed model may show a "poor" performance if extended to a data set that has not been used in the model construction.

The significance of each of these above-mentioned issues varies from one modeling method to another. For example, in the nonlinear "black-box" modeling approach, these problems are very significant due to the fact that there is no physical insight available or used (i.e., model equations are not based on first physical principles). In this research, we investigate these problems for the specific case of the "black box" modeling approach based on fuzzy-logic modeling.

In (Emami, 1996a), a systematic methodology of fuzzy-logic modeling using the system input-output data was developed. This methodology provides a framework for identifying the structure and the parameters of the fuzzy model. This thesis improves the systematic modeling methodology with regard to the above-mentioned problems.

The investigation performed in this thesis has covered the following points:
• The continuity behaviour of the fuzzy-logic model
In the domain in which the actual system is continuous for small variations in the input, the fuzzy-logic model is expected to generate small changes in the output whether or not the values of the deviated input have been used in the training procedure. This property is crucial for the generalization of the fuzzy-logic model to a new data set. In the literature, this is referred to as the robustness property (Sergai, 1996). In this thesis, we simply call the above characteristic the continuity of the model. A definition of the robustness is introduced in Section 1.4.

• The noise-rejection capability of the fuzzy-logic model
The fuzzy model should be able to reproduce the training set accurately (within a pre-specified error). The methodology should identify the noise and the outliers in the training set and eliminate their effects during the model identification.

• The generalization capability of the fuzzy-logic model
This property is the final goal of the thesis. We investigate how the fuzzy-logic model can be extended from the training data set to a new data set that has not been used in the model construction. In order to perform this investigation, some conditions are developed for the fuzzy membership functions to improve the generalization capability of the fuzzy-logic model. Moreover, to evaluate an identified fuzzy model, a comparison between different evaluation criteria introduced in the literature is performed.

1.2 Fuzzy-Logic Modeling
Fuzzy modeling is an approach to form a system model using descriptive language based on fuzzy logic with fuzzy propositions. The linguistic approach of system modeling can be formulated by three separate features (Zadeh, 1973):
• The use of fuzzy variables in place of, or in addition to numerical variables:
• The characterization of simple relations between variables by IF-THEN rules:
• The characterization of complex relations by fuzzy reasoning process.
CHAPTER 1

INTRODUCTION

IF $U_1$ is $B_{11}$ AND ... AND $U_r$ is $B_{1r}$ THEN $V_1$ is $D_{11}$ AND... AND $V_s$ is $D_{1s}$.

ALSO

........

ALSO

IF $U_1$ is $B_{n1}$ AND ... AND $U_r$ is $B_{nr}$ THEN $V_1$ is $D_{n1}$ AND... AND $V_s$ is $D_{ns}$.

Figure 1.1: The fuzzy input-output rules

Fuzzy systems are based on the concept of fuzzy partitioning of information. The decision-making ability of the fuzzy model depends on the existence of a set of rules and a fuzzy reasoning mechanism. In general, the encoded knowledge of a Multi-Input Multi-Output (MIMO) system can be interpreted by fuzzy models consisting of IF-THEN rules with multi-antecedent and multi-consequent variables (with $r$ antecedents, $s$ consequents, and $n$ rules) as shown in Figure 1.1. In this Figure, $U_1$, $U_2$, ..., $U_r$ are input variables, and $V_1$, $V_2$, ..., $V_s$ are output variables. $B_{ij}$ ($i=1,...,n, j=1,...,r$) and $D_{ik}$ ($i=1,...,n, k=1,...,s$) are fuzzy sets of the universes of discourse $X_1$, $X_2$, ..., $X_r$ and $Y_1$, $Y_2$, ..., $Y_s$ of $U_1$, $U_2$, ..., $U_r$ and $V_1$, $V_2$, ..., $V_s$, respectively. The fuzzy sets $B_{ij}$ and $D_{ik}$ are parameters of the fuzzy model. The number of rules of the fuzzy model determines its structure, analogous to the order of the analytical model.

1.3 Background: A Review of the Systematic Methodology of Fuzzy-Logic Modeling

As mentioned in Section 1.1, we adopt the systematic methodology of fuzzy modeling proposed by Emami et al. (Emami. 1996a). The goal of the systematic approach is to improve the objectivity of fuzzy modeling by developing appropriate formulations and criteria to specify those features of the model that are usually assigned heuristically in fuzzy modeling approaches. The systematic approach to fuzzy modeling involves two major steps: (i) a parameterized reasoning process; and (ii) a fuzzy system
identification algorithm. The reasoning mechanism addresses the following: (i) the implementation of the information in each rule; (ii) the aggregation of information among a set of rules; and (iii) fuzzy logic inferred values mapping into crisp equivalent values.

The system identification process can be divided into two parts: structure identification and parameter identification. The structure identification problem consists of the fuzzy rules generation and the construction of the input-output membership functions. The output space is first clustered, then fuzzy partition of the input space is generated by "projection" of the output clustered space onto each of the input variables.

In the parameter identification process, two separate steps are performed. In the first step, the system parameters (i.e., input-output membership functions parameters and the reasoning parameters) are optimized by minimizing a performance index. In the second step, the input-output membership parameters are tuned the same way as ordinary system identification methods. Fig 1.2 shows a flow chart of the fuzzy modeling process. In the next sub-sections, we introduce a detailed mathematical description of the different phases of the fuzzy modeling process.

Figure 1.2: A Flow chart of fuzzy system modeling (Emami, 1996a)
1.3.1 Reasoning Mechanism

The systematic methodology (Emami. 1996a) considers the inference mechanism as an “identifiable” object of fuzzy systems. A unified parameterized formulation of the reasoning process was developed as follows: for the crisp input $x^* = (x_1^*, x_2^*, ..., x_r^*)$, the fuzzy output of the system introduced in fig. 1.1 is obtained as:

$$E(y) = \beta \left( 1 - S_p \left[ T_p \left[ \tau_1 (x^*) , \hat{D}_1 (y) \right] , ..., T_p \left[ \tau_n (x^*) , \hat{D}_n (y) \right] \right] \right)$$

$$+ (1- \beta) S_p \left[ T_p \left[ \tau_1 (x^*) , \hat{D}_1 (y) \right] , ..., T_p \left[ \tau_n (x^*) , \hat{D}_n (y) \right] \right] , \tag{1.1}$$

where $\tau_i$, called the “rule degree of firing”, is computed as:

$$\tau_i (x^*) = T_p (B_{i1} (x_1^*) , B_{i2} (x_2^*) , ..., B_{ir} (x_r^*)) . \tag{1.2}$$

and

$$\overline{D} = 1 - D_i (y) . \tag{1.3}$$

where $r$ is the number of antecedents, $n$ is the number of consequent, and $S_p$ is the $n$-ary $t$-conorm operator computed as:

$$S_p (a_1 , a_2 , ..., a_n) = a_1 ^p + (1-a_1 ^p ) [a_2 ^p + (1-a_2 ^p )[...[a_{n-2} ^p + (1-a_{n-2} ^p )[a_{n-1} ^p + (1-a_{n-1} ^p )a_n ]]]]^{1/r} \quad p > 0 \tag{1.4}$$

The operator $T_p (x = p,q)$ is the $n$-ary $t$-norm operator calculated as:

$$T_p (a_1 , a_2 , ..., a_n) = 1 - S_p (((1-a_1 ),(1-a_2 ) ,...(1-a_n )), \tag{1.5}$$

Equation 1.1 is a linear combination of two extreme reasoning approaches, Mamdani’s and Logical, with adjustable parameters (Emami. 1996a). The crisp output is then obtained using the Basic Defuzzification Distribution method as follows (Yager. 1994):

$$y^* = \frac{\int_{y_0}^{y_1} y [E(y)]^a dy}{\int_{y_0}^{y_1} [E(y)]^a dy} \quad 0 \leq a \leq \infty . \tag{1.6}$$
In equations 1.1 and 1.6, four reasoning parameters $p$, $q$, $a$, and $\beta$ are introduced whose variation will cause a continuous range of variation in the reasoning mechanism. As a result, unlike traditional approaches of selecting the inference mechanism \textit{a priori}, the optimum reasoning mechanism would be identified for the system by adjusting the above parameters based on the input-output data.

\textbf{1.3.2 Fuzzy Structure Identification}

Fuzzy structure identification is the process of assigning the optimum number of rules, significant input variables, input and output membership functions, and the amount of overlap between the membership functions required for the fuzzy model. These are briefly discussed in the following sub-sections.

\textbf{1.3.2.1 Rule Generation and Output Membership Functions}

An intuitive approach to objective rule generation is based upon the clustering of the input-output data. However, in the systematic methodology, we first cluster the output space (Sugeno, 1993), and then derive the input space fuzzy partition by projecting the output space partition onto each input space, separately. Simplicity, particularly for systems with a large number of input variables, is the main advantage of this approach. The output fuzzy clustering is carried out by the Fuzzy C-Means (FCM) algorithm (Bezdek, 1981). The idea of fuzzy clustering is to divide the output data into fuzzy partitions that overlap each other. Therefore, the containment of each datum $y_i$ to each cluster $i$ with a center $v_i$ is defined by a membership grade $u_{ik}$ in $[0.1]$. The membership grades and the cluster centers are obtained through an iterative procedure as follows:

$$u_{ik,j} = \left[ \frac{\sum_{j=1}^{J} \left( \frac{\sqrt{y_k - v_{i,j-1}}}{\sqrt{y_k - v_{j,j-1}}} \right)^2}{\sum_{j=1}^{J} \left( \frac{\sqrt{y_k - v_{i,j-1}}}{\sqrt{y_k - v_{j,j-1}}} \right)^2} \right].$$

where

$$v_{i,j} = \frac{\sum_{k=1}^{N} \left( u_{ik} \right)^n y_k}{\sum_{k=1}^{N} \left( u_{ik} \right)^n}.$$
and $N$ is the number of data, $u_{ikl}$, $v_{ikl}$ are the membership grade of the output $y_k$ in the cluster $i$ and the center of the cluster $i$, respectively, at the $l^{th}$ iteration. Three crucial pieces of a priori information are required for constructing the suitable partitions of the data: (i) an adequate number of rules for expressing the system behaviour that in most cases, equals to the number of clusters $c$; (ii) the order of fuzziness of the system model, which is the overlap of the fuzzy clusters, and is adjusted by the parameter $m$, called "weighting exponent"; and (iii) the suitable initial location of the cluster centers which affect the model formation.

**Specification of the Number of Rules:** The following cluster validity index was developed for assigning the optimum number of output clusters (Emami. 1996b), and hence, the number of rules, in the fuzzy model:

$$s_{cr} = \sum_{k=1}^{N} \left( \sum_{i=1}^{c} (u_{ik})^m \left( (y_k - v_r)^2 - (v_r - \bar{v})^2 \right) \right).$$  \hspace{1cm} (1.9)

where $\bar{v}$ is the weighted mean of data considering their membership to each of the clusters defined as:

$$\bar{v} = \frac{1}{\sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m} \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m y_k.$$  \hspace{1cm} (1.10)

Minimization of $s_{cr}$ will increase the compactness of clusters and the separation between them. Hence, the optimum number of clusters $c$ corresponds to minimum $s_{cr}$. In most cases, $c$ is equal to the number of rules $n$ for the fuzzy model.

**Specification of the Order of Fuzziness:** The weighting exponent $m$ controls the extent of membership sharing between output fuzzy clusters in the data set. In the range of $(1, \infty)$, the larger the $m$, the "fuzzier" the membership assignments to each data point. The following index was developed to select $m$ (Emami. 1996b):

$$S_T = \sum_{k=1}^{N} \left( \sum_{i=1}^{c} (u_{ik})^m \right) (y_k - \bar{v}).$$  \hspace{1cm} (1.11)
An appropriate value for $m$ is chosen to make $S_T$ equal to a constant parameter $\sqrt{2}$. where $z$ is defined as:

$$z = \sum_{k=1}^{N} \left( y_k - \frac{1}{N} \sum_{j=1}^{N} y_j \right)^2 . \tag{1.12}$$

**The Initial Cluster Centers:** The initial cluster centers for the FCM algorithm are assigned through hard clustering techniques (Ward, 1963). This approach provides a more efficient strategy compared to the previous approach of randomly selecting the initial values.

**1.3.2.2 Input Selection and Input Membership Function Assignment**

In order to identify the significant input variables among a finite number of candidates, we first project the output clusters onto the space of each of the input candidates. As a result, for each input candidate $x_j$, the membership functions $\hat{B}(i=1,2...n)$ are formed. Then, we can define the following index:

$$\pi_j = \prod_{i=1}^{n} \frac{\Gamma_{\hat{B}_i}}{\Gamma_j} , \quad j=1,2,...,r \tag{1.13}$$

where $\Gamma_{\hat{B}_i}$ is the range in which the membership function $\hat{B}_i$ is equal to one, $\Gamma_j$ is the entire range of $x_j$, $n$ is the number of rules, and $r$ is the number of input candidates. Less $\pi_j$ indicates a more dominant variable $x_j$, and hence, significant variables are selected among those that produce less $\pi$.

The convex membership functions $B_{ij}$ for significant inputs $x_j (j=1,2...r)$ are then formed by using the range $\Gamma_{\hat{B}_i}$ and performing “fuzzy line clustering” (Emami, 1996b).

**1.3.2.3 Fuzzy Parameter Identification**

The optimum values of the inference parameters ($p$, $q$, $\beta$, and $\alpha$) are identified through a nonlinear constrained optimization problem by minimizing:

$$PI (p,q,\beta,\alpha) = \sum_{k=1}^{N} (y_k - \hat{y}_k)^2 / N . \tag{1.14}$$
subject to the following constraints:

\[
0 < p, q < \infty \quad \& \quad 0 < \alpha < \infty \quad \& \quad 0 < \beta < 1 \quad (1.15)
\]

where \( y_k \) is the actual output and \( \tilde{y}_k \) is the model output.

The input-output membership functions that have already been identified in the structure identification phase are approximated by trapezoidal functions, and then an incremental tuning procedure is used to adjust the membership function parameters based on the training data set and the performance index in equation (1.14) (Emami. 1996a).

1.4 Definition of the Model Robustness

We postulate that the immediate step necessary after developing the fuzzy-logic model is to investigate the following three important characteristics of the model: continuity, noise rejection, and the generalization capability. In this work, we define a "robust" model as one that has the following properties: (i) continuous behaviour: (ii) rejects noise and the outliers in the training data set: and (iii) shows a satisfactory generalization capability when extended to a set of data that has not been used in the model construction. Each of the three properties mentioned above is explained in details as follows:

- **Continuous behaviour:** we have to insure that for the input data points close to the training data set (slightly outside the range of the training set) the model output should also be close to the corresponding outputs. We call this characteristic "marginal continuity". Marginal continuity is equivalent to the definition of the modulus of continuity in the approximation theory (Lorenz, 1966). The marginal continuity characteristic is explained in detail in Chapter 2.

- **Noise rejection capability:** the robust fuzzy model should recognize noise in the training set and eliminate its effect during the structure identification, parameter adjustment, and tuning processes. The hard and the fuzzy C-Means algorithms are the partitioning tools used in many applications. However, they are highly sensitive to the presence of noise and outliers. The FCM algorithm uses the sum of squared errors in its objective function. Therefore, this clustering method may fail completely in the presence of noise (Krishnapuram, 1994). The robust fuzzy clustering algorithms should accurately identify the initial cluster centers in the presence of noise. assign
the appropriate partitions in the data, and reject the noise and the outliers by giving them low or zero membership grades.

- **Generalization capability**: we need to minimize the model complexity (number of rules, number of membership functions, etc) while simultaneously maximizing its accuracy. In this regard, some questions need to be answered: (i) what conditions on the training data result in a reliable model? (ii) what are the suitable criteria to construct the input and the output membership functions? (iii) what is the most suitable training method? and (iv) what are the appropriate criteria to evaluate the identified model? Finding answers to these questions requires new tools to improve the generalization capability of the fuzzy model.

### 1.5 Literature Review

#### 1.5.1 Robustness of the Fuzzy Inference Mechanism

The research published regarding the robustness of the fuzzy inference mechanism is limited. The robustness of AND/OR operators in fuzzy systems is addressed in (Nguyen, 1993). Fuzzy-logic connectives (AND, OR, ALSO, etc.) are characterized by a concept similar to the modulus of continuity in the approximation theory (Lorenz, 1966). It was shown (Nguyen, 1993) that Min-Max operators (Zadeh, 1965) are the most robust operators. Moreover, a theoretical approach related to the robust properties of different fuzzy connectives has been introduced by (Nguyen, 1993, Yager, 1994).

Furthermore, Sergai (Sergai, 1996) introduced a definition of a robust system as follows: consider a system \( X \) with \( n \)-inputs and \( m \)-outputs where the inputs and the outputs are fuzzy sets. Assume that all the fuzzy sets are defined on a finite set \( P=\{ p_1, p_2, \ldots, p_n \} \) and suppose that we (slightly) modify the membership functions of the input. The system \( X \) is robust if the output is also (slightly) changed because of the changes in the input membership functions.

#### 1.5.2 Robustness of the Fuzzy Clustering Algorithm

An important step in objective fuzzy modeling is fuzzy rule generation in which the system data are partitioned into fuzzy clusters. We can define clustering as partitioning of
a group of unlabeled data into a number of clusters such that similar data is assigned to one cluster and data that are less similar is assigned to different clusters (Emami, 1996b).

Two main approaches for clustering are usually used: hard clustering and fuzzy C-Means clustering. Hard clustering assigns each data point to one and only one cluster, though, the boundaries between clusters are not always fully defined. The fuzzy C-Means algorithm (FCM) was suggested by Bezdek et al. (Bezdek, 1981). The FCM algorithm is the most commonly used algorithm for data partitioning. The FCM has three major problems:

- In order to get optimum partition, the exact initial cluster centers locations should be assigned. The FCM algorithm always converges to a local extreme of $J_w$ (weighted within groups sum of squared errors objective function). Different choices of initial cluster centers may lead to different extremes.
- The scientific basis for the choice of the weight exponent is still not clear.
- The optimum number of partitions in the data should be assigned. There should be a cluster validity criteria to assign the optimum number of clusters.

In the clustering algorithm developed by Emami et al. (Emami, 1996b), a solution is proposed to each of the three above-mentioned problems:

- An approach for choosing the optimum number of clusters is to make the fuzzy clusters compact and far from each other. The idea of scatter matrices is generalized to fuzzy clustering. As a result, a validity index is introduced for the choice of the optimum number of clusters.
- The weight exponent should be chosen far from both of its extremes in order to ensure that the cluster validity index shows the optimum number of fuzzy clusters.
- For the choice of the initial cluster centers, an agglomerative hierarchical clustering algorithm (AHC) was suggested (Ward, 1963). The AHC algorithm puts each of the $N$ data vectors in an individual cluster. Then, by defining a dissimilarity matrix, it merges two or more of these clusters, getting to a more general level of data partition. The process is repeated to form a sequence of nested clustering in which the number of clusters decreases gradually until the required number of clusters is reached.

The drawbacks of the fuzzy C-Means algorithm are not the only concern in this work. As mentioned in Section 1.4, the original fuzzy C-Means clustering algorithms may fail
in the presence of noise. As we expect that the adopted clustering algorithm would show a noise rejection capability, the fuzzy C-Means clustering algorithm by itself may be an inappropriate tool for partitioning the data in the presence of noise and disturbances.

Krishnapuram et al. introduce a family of modified fuzzy C-Means algorithms with noise rejection capability (Krishnapuram, 1994), (Kim, 1995), (Krishnapuram, 1996). Two different noise rejection clustering algorithms introduced by Krishnapuram et al. and Kim et al. are summarized as follows:

1) A robust version of the fuzzy C-Means algorithm is obtained by combining it with the least trimmed square algorithm (Kim, 1995). The least trimmed square algorithm is a general regression algorithm in statistics and is used in many engineering applications. The algorithm depends on an estimator $\hat{\vartheta}$ that achieves robustness by trimming observations with large residuals. The derived version of the FCM algorithm is called the fuzzy trimmed C-Means algorithm (FTCM). Unlike the usual clustering algorithms that estimate cluster prototypes using all the feature vectors including noise, the FTCM algorithm tries to eliminate noisy feature vectors using the minimum or the harmonic mean distance before estimating the cluster prototypes. The FTCM estimates the optimum trimming ratio fairly well and identifies the noise in the data set.

2) An improved fuzzy C-Means algorithm called the Possibilistic clustering algorithm (PCM) is proposed by Krishnapuram et al. (Krishnapuram, 1996) to achieve robustness and eliminate the noise vectors from the data set. The PCM is more robust than the original FCM in presence of noise because its objective function and the center update equations involve unconstrained weights that decrease with the distance from cluster centers. This results in low weights for outliers and decreases their influence. The PCM tries to find the best $c$ clusters independently of each other. The PCM most significant advantage is that it tries to find the valid clusters and gives a robust estimate of the centers after a suitable initialization. On the other hand, the main disadvantage of the PCM algorithms is that they suffer the same drawbacks of the original FCM algorithm.
1.5.3 Generalization Capability of Fuzzy-Logic Models

Research on the generalization capability of the fuzzy-logic models has only generated some general guidelines (Sjoberg, 1995). In order to improve the generalization capability of fuzzy models, the following issues must be taken into consideration: (i) a careful study of the data sampling process should be performed. (ii) over-tuning the model for the training set must be avoided; and (iii) appropriate criteria to evaluate the identified model should be implemented. Summary of the literature on these issues follows:

1) Conditions on the data: a prerequisite step for any modeling and system identification approach is to perform a close and careful observation of the data sampled from the system to be modeled. Samples provide valuable information about the system. Hence, certain precautions must be taken in collecting the data, and certain conditions must be satisfied by the sampled observations. Before undertaking the modeling process, the following questions regarding the data should be answered (Emami, 1997a):

(a) Is the data provided through the experiment meaningful?
(b) Is there any sign of dependence or contingency between the output and the input variables?
(c) Do the input variables vary independently in the data set?
(d) How much we can reduce the number of data such that major information remains intact?

Answer to these questions are presented in Chapter 4.

2) Model estimation: The key problem in system identification is to find a suitable model structure. Assume that we have collected some sets of input-output data from a system and we want to estimate a model based on them. Sjoberg et al. (Sjoberg, 1995) propose studying the following issues to achieve successful results:

- Simple start: "simple" may mean both the size and the computational complexity of models. By searching from simpler to more complex models until a valid one is found, a good trade-off between bias and variance can be achieved.
Training and testing data: the best way to evaluate an identified model is to test it on data not used in the model construction. The use of testing data involves determining the model complexity, the size of the optimization parameters, and the most efficient method for tuning.

The efficient number of parameters: the variance contribution to the model output error is proportional to the number of parameters used in the model structure. These parameters will be optimized through the minimization of the model error. Some of the parameters are not as important for the model fit. If these parameters are estimated inaccurately, even though they do not influence the fit, their large error will be undesirable. The iteration process for tuning the model parameters is stopped where different testing data show a minimum pre-defined error.

3) Model evaluation criteria: A robust fuzzy model should show a satisfactory performance when extended to a data set that has not been used in the model construction. In this regard, we need to define a criterion for the evaluation of the fuzzy-logic models. The parsimony principle can be used as a ground for evaluating the identified models (Box, 1970). This principle has two major indices: Akaike's information criterion (Akaike, 1973) and Occam's Razor (MacKay 1995, Blumer, 1987, Domingo 1997, Natarajan, 1993). Both indices can be used as tools for evaluating an identified model or assigning the most generalized model among a group of competing models in terms of how satisfactorily they perform for the different testing data sets.

4) Tuning the model parameters: Sjoberg et al. (Sjoberg, 1995) propose a tuning method based on evaluating different testing data sets during the tuning process of the model training set. Thus, without over-tuning the training set, the most suitable number of iterations can be chosen such that the different testing data as well as the training data show minimum error.
1.6 Thesis Contents

The contents of this work can be summarized as follows:

1) In Chapter 2, the generalized inference formulation applied in the systematic methodology of fuzzy modeling (Emami, 1996a) is examined. A mathematical formulation of the marginal continuity that relates the deviation of the input membership function with the corresponding deviation of the crisp output is developed. The modulus of continuity is used as a measure of the deviation of the model output. Range constraints are applied to the reasoning parameters $P(p,q,\alpha)$ (Emami, 1996a) to adjust the sensitivity of the inference mechanism to account for the deviation of the input membership function. The developed methodology is tested on the Sugeno nonlinear static model (Sugeno, 1993).

2) In Chapter 3, the robustness of the fuzzy clustering algorithm is investigated. The noisy data can be handled properly by a modified version of the fuzzy C-Means clustering algorithm introduced by Krishnapuram et al. (Krishnapuram, 1996). The modified fuzzy C-Means algorithm is combined with the techniques developed in Emami et al. clustering algorithm (Emami, 1996b) to overcome the drawbacks of the original fuzzy C-Means algorithm (initialization problem, order of fuzziness, and the assignment of the number of cluster). The new algorithm efficiently assigns the partitions in the data subject to the presence of noise. Several examples are presented in this chapter to illustrate the significance of the improved algorithm.

3) Chapter 4 investigates the generalization capability of the fuzzy model. As a result, the following issues are addressed: (i) some conditions on the input-output membership functions are introduced; (ii) some precautions are recommended when collecting the data; and (iii) two model evaluation criteria are studied.

4) In Chapter 5, the proposed robustness techniques are illustrated through two real applications. The final conclusions and the proposed future research are introduced in Chapter 6.
1.7 Contributions

Three basic properties of a robust fuzzy-logic model, i.e., continuity, noise rejection, and generalization are directly affected by the inference mechanism of the model, the classification approach, and the conditions on the data and the model performance indices, respectively. We adopted the systematic methodology of fuzzy-logic modeling (Emami et al., 1996). In this thesis, we improved the systematic methodology of fuzzy modeling in terms of the robustness by studying each of the above mentioned properties of the robust fuzzy model and introducing the following guidelines:

1) The robustness theory developed in Nguyen et al. (Nguyen, 1993), Sergai (Sergai, 1996), and Lorenz (Lorenz, 1966) was implemented to check and improve the continuity of the formulation. A mathematical formulation of the marginal continuity that relates the deviation of the input membership function to the deviation of the fuzzy output was developed. Some conditions on the reasoning parameters $p, q, \alpha$ (Emami, 1996a) were developed to adjust the sensitivity of the inference mechanism to the deviation of the input membership function. A modulus of continuity was used as an index to measure the deviation of the model crisp output.

2) The modified version of the fuzzy C-Means algorithm (Krishnapuram, 1996) was combined with the techniques developed in the clustering algorithm of the systematic methodology of fuzzy modeling (Emami, 1996b). The main reason for using this combination is to avoid the original FCM drawbacks (initialization problem, order of fuzziness, and the number of cluster assignments). The new algorithm is a powerful and systematic approach to partitioning the data in the presence of noise and disturbances.

3) In order to enhance the generalization capability of the fuzzy model, certain conditions on the data collected were recommended. Some conditions on the input-output membership functions were introduced to improve the fuzzy model behaviour when used to evaluate the testing data sets. Different methods of evaluating the generated model such as AIC and Occam's Razor were studied. Finally, a comparison between these two methods was performed based on the applicability of each of them to validation of the fuzzy model.
4) The proposed approach to achieve robustness of fuzzy-logic model was examined experimentally on a 4 degree-of-freedom robot manipulator (IRIS) (Hui et al., 1993). The proposed robustness method was also applied to the data of the gas furnace plant studied in Box et al. (Box, 1970).
CHAPTER 2. ROBUSTNESS OF THE FUZZY INFERENCE MECHANISM

2.1 Introduction

In this chapter, we investigate the continuity behavior of the fuzzy-logic model. The fuzzy model should be constructed in such a way that a small change of the input leads to a small change of the output. In other words, we would like to ensure that there is no unwanted “jump” in the model output. Thus, for input data points close to the training data set, the model output must also be close to the corresponding output of the training data set. This characteristic is the main intuitive pre-assumption of the modeling and system identification, and we call it “marginal continuity”. The marginal continuity characteristic is one of the three basic properties of the robust fuzzy models as explained in Section 1.4.

In Section 2.2, the generalized inference formulation of the systematic methodology of fuzzy modeling (Emami, 1996a) is reviewed. Since in the proposed modeling strategy, the inference mechanism is parameterized as an analytical closed-form formulation, it is feasible to derive conditions for marginal continuity.

In Section 2.3, the tools available for examining the continuity of the reasoning formulation are presented. In Nguyen et al. (Nguyen, 1993), the fuzzy logic connectives are characterized with a concept similar to the modulus of continuity in the approximation theory (Lorenz, 1966). Unlike this approach, our interest is not merely in the robustness of the individual operators, but also in the robustness of the entire fuzzy inference mechanism used for the reasoning process. An index for the deviation of the
input membership functions is introduced. The concept of the *modulus of continuity* is used to express the deviation of the fuzzy output of the model.

In Section 2.4, a mathematical formulation of the marginal continuity that relates the deviation of the input membership functions to the deviation of the fuzzy output is developed. The modulus of continuity is used as a measure of the maximum deviation of the fuzzy output. The relation between the maximum possible deviation of the input membership functions and the reasoning parameters $\mathcal{X}(p,q)$ is studied. Through the developed formulation, the lower bounds of the reasoning parameters can be specified. By adjusting the lower bounds of the reasoning parameters, we can adjust the sensitivity of the inference mechanism to the deviation of the input membership functions.

In Section 2.5, the concept of the marginal continuity is extended to the crisp output of the model, as we are practically more interested in the crisp output. An index is introduced to assign a domain for the defuzzification parameter $\alpha$ such that the crisp output deviation never exceeds a value pre-specified by the modulus of continuity.

In Section 2.6, a technique for assigning the model parameters from the training data set is introduced. In Section 2.7, the results are illustrated through a simple and famous example in the literature (Sugeno, 1993). For this example, a comparison study between the developed "robust" modeling and the original methodology (Emami, 1996a) is performed for different sets of testing data. The conclusions are discussed in Section 2.8.

### 2.2 A Unified Parameterized Formulation of the Reasoning Process

For a Multi-Input Single-Output (MISO) system, the reasoning process is divided into the following steps:

- Fuzzy aggregation of the antecedents in each rule (**AND** connective in Fig 1.1);
- Implication relation for each individual rule (**IF-THEN** connective in Fig 1.1);
- Aggregation of the rules (**ALSO** connective in Fig 1.1);
- Inference from the set of rules and the observed input to obtain the fuzzy output;
- Defuzzification of the output.
Unlike the traditional approach of selecting the inference mechanism \textit{a priori}. The systematic methodology considers the inference mechanism as an “identifiable” object of the fuzzy systems. The unified parameterized formulation of the reasoning process is introduced as a linear combination of the two extremes known as Mamdani’s and the Logical approach, as suggested by Yager \textit{et al.} (Yager, 1994). For the crisp input vector
\[ x^* = (x_1^*, x_2^*, ..., x_r^*) \]
the fuzzy output of the system is obtained as:
\[
E(y) = \beta F_L(y) + (1-\beta) F_M(y).
\] (2.1)
where the fuzzy output of the Logical approach
\[
F_L(y) = 1-S_p[T_p(\tau_i(x^*)), (1-D_i(y))],..., T_p(\tau_n(x^*)), (1-D_n(y))].
\] (2.2)
and the fuzzy output of the Mamdani’s approach
\[
F_M(y) = S_p[T_p(\tau_i(x^*)), D_i(y)],..., T_p(\tau_n(x^*)), D_n(y)]
\] (2.3)
are linearly combined through the parameter \( \beta \) as expressed in equation 2.1.
The parameter \( \tau_i \) is called the “rule degree of firing”, and is computed as:
\[
\tau_i(x^*) = T_q(B_{1i}(x_1^*), B_{2i}(x_2^*),..., B_{ni}(x_n^*)): \; i = 1, 2, ..., r
\] (2.4)
In equation (2.4), \( n \) is the number of rules, \( r \) is the number of input variables, and the operator \( S_p \) in equation 2.3 is the \( n \)-ary \( t \)-conorm operator computed as (Emami, 1996a):
\[
S_p(a_1, a_2, ..., a_n) = \left( \sum_{i=1}^{n} a_i^* - \sum_{j=1}^{n} a_j^* a_i^* + \sum_{j=1}^{n} \sum_{k=j+1}^{n} a_j^* a_k^* a_i^* - + \cdots + \prod_{i=1}^{n} a_i^* \right)
\] (2.5)
The operator \( T_x(\chi = p,q) \) is the \( n \)-ary \( t \)-norm operator that is calculated as (Emami, 1996a):
\[
T_x(a_1, a_2, ..., a_n) = 1-S_x((1-a_1), (1-a_2), ..., (1-a_n))
\] (2.6)
The crisp output is then obtained by using the \textit{Basic Defuzzification Distribution} method as follows (Yager, 1994):
\[
y^* = \frac{\int_{y_1}^{y_n} y[E(y)]^\alpha \, dy}{\int_{y_1}^{y_n} [E(y)]^\alpha \, dy} \quad 0 \leq \alpha \leq \infty.
\] (2.7)
In the above reasoning formulation (i.e., equations 2.1 and 2.7), four reasoning parameters \( p, q, \alpha, \) and \( \beta \) are introduced, and their variation will cause a continuous range
of changes of the reasoning mechanism. By adjusting the above parameters, the optimum reasoning mechanism can be identified based on the input-output data (Emami, 1996a).

2.3 The Modulus of Continuity and the Deviation of the Input Membership Functions

The concept of the “modulus of continuity” is often applied to measuring the variation of a function as a result of the change of its independent variables. For a real function \( g \) that belongs to the family of continuous functions \( c[X] \) defined over the real interval \( X = [a,b] \), the modulus of continuity is defined as follows (Lorenz, 1966):

\[
\gamma_g(h) = \max_{x,\delta} \left| g(x + \delta) - g(x) \right| ; \quad \text{for each } x \in X \text{ and } |\delta| \leq h \tag{2.8}
\]

where \( 0 \leq h \leq (b-a) \).

Figure 2.1 illustrates the above definition. Depending on the distance \( h \), the value of \( \gamma_g \) varies from zero (for \( h = 0 \)) to a maximum value when \( h \) is equal to the entire range, i.e., \( h = b-a \). The value of \( \gamma_g \) indicates how sharply the function \( g \) varies within its range.

The following properties of the real function \( \gamma_g(h) \) can be deduced readily:

- \( \gamma_g(h) \) is continuous
- \( \gamma_g(h) \to 0 \) as \( h \to 0 \)
- \( \gamma_g(h) \) is positive and monotonically increasing:
- \( \gamma_g(h) \) is subadditive, i.e., \( \gamma_g(h_1 + h_2) \leq \gamma_g(h_1) + \gamma_g(h_2) \).

![Figure 2.1: The modulus of continuity of the function g](image-url)
The definition of the modulus of continuity can be extended to the multi-variable real functions \( g(x_1, x_2, \ldots, x_r) \) as follows:

\[
\gamma_k(h_1, h_2, \ldots, h_r) = \max_{x, \delta} | g(x_1 + \delta_1, x_2 + \delta_2, \ldots, x_r + \delta_r) - g(x_1, x_2, \ldots, x_r) | \tag{2.9}
\]

for each \( x = (x_1, x_2, \ldots, x_r) \in X \) and \( |\delta_k| \leq h_k : k = 1, 2, \ldots, r \).

Here, \( X \) is the hyper-parallelepiped defined in the \( r \)-dimensional sub-space \([a_1, b_1], [a_2, b_2], \ldots, [a_r, b_r]\), and \( 0 \leq h_k \leq (b_k - a_k) \), \( k = 1, 2, \ldots, r \).

It is possible to simplify the definition in equation 2.9 to a single-variable real function \( \gamma_k(h) \) if we choose:

\[
h = \max (h_1, h_2, \ldots, h_r) \tag{2.10}
\]

Therefore, we have:

\[
\gamma_k(h) = \max_{x, \delta} | g(x_1 + \delta_1, x_2 + \delta_2, \ldots, x_r + \delta_r) - g(x_1, x_2, \ldots, x_r) | \tag{2.11}
\]

for each \( x \in X \) and \( |\delta_k| \leq h : k = 1, 2, \ldots, r \).

The output of a system is a function of its input, and hence the modulus of continuity would be an appropriate index to indicate the variation of the system output. In this research, we are primarily interested in the fuzzy output that is inferred from a set of rules having some input observations. The existence of a closed-form formulation for the reasoning mechanism makes it feasible to exploit the modulus of continuity index and derive some analytical formulations for representing the output variation. However, some slight modifications of the definition in equation 2.11 are required. In a fuzzy-logic model, the values of the input variables are first mapped to their membership spaces, a procedure called "fuzzification". In other words, for a fuzzy model, it is not the value of the input but its grade of membership to the input clusters that matters. Therefore, mathematically, the output of the fuzzy model is primarily a function of the input "membership grades". This would not cause a significant difference as long as the input membership functions are "smooth enough", i.e., a "slight" deviation of the input variables leads to a "slight" deviation of the input membership grades. This condition is perfectly fulfilled for most of the membership function families such as the trapezoidal functions, which are the ones we use in this research. Figure 2.2 shows how we can translate the deviation of the input variable \( x_j (j = 1, 2, \ldots, r) \) to the deviation of its
corresponding membership grade $B_{ij}$ in the $i^{th}$ rule ($i = 1, 2, \ldots, n$). We use the ~ sign for the deviated function. We recall that $r$ is the number of input variables and $n$ is the number of rules.

Further, in the fuzzy-logic model, for a set of crisp input values $x^* = (x_1^*, x_2^*, \ldots, x_r^*)$, the fuzzy output is a “membership function” $E(y)$ defined over the output real axis $\mathbb{R}$. Therefore, the output variation for a specified crisp input vector is a function defined over the output space as illustrated in figure 2.3. Since we are interested in the maximum deviation, then the variation of the fuzzy output can be expressed as follows (refer to Fig 2.3):

$$e(B_{ij}, \delta_j) = \max_{\delta} |E(y; B_{ij} + \delta_j) - E(y; B_{ij})| : y \in \mathbb{R}$$

(2.12)

where $e(B_{ij}, \delta_j)$ is a real function of the input membership grades $B_{ij}(x_i)$ and the deviation $\delta_j$, and $e(B_{ij}, \delta_j)$ varies between 0 and 1. We can now define the modulus of continuity of the function $e$ as:

$$\gamma(h) = \max_{B_{ij}, \delta_j} |e(B_{ij}, \delta_j)|$$

(2.13)

for each $B_{ij} \in [0,1]$ and $|\delta_j| \leq h$, where $h = \max_{i,j} (h_{ij})$ : $0 \leq h \leq 1$ and $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, r$.

In the next section, we attempt to obtain an analytic formulation of equation 2.13 as a function of $h$ and the inference parameters $p, q, \alpha, \beta, \gamma$.

---

**Notes:**
- The figure 2.2 illustrates the deviation of the membership functions.
- The text explains the deviation of the membership grades in the context of fuzzy logic and inference mechanisms.
### 2.4 Marginal Continuity of the Fuzzy Output

#### 2.4.1 Single-Input Single-Output Systems

In this section, we investigate the conditions that guarantee a desirable marginal continuity of the fuzzy output. A mathematical formulation that describes the relation between the maximum deviation of the input membership grades and the maximum deviation of the fuzzy output is developed. The appropriate values of the parameters \( p \) and \( q \) are then specified through the developed formulation. For simplicity, we first consider a single-input single-output system with two rules in the following form:

\[
\text{IF } x \text{ is } B_{1i} \text{ THEN } y \text{ is } D_1 \\
\text{IF } x \text{ is } B_{2i} \text{ THEN } y \text{ is } D_2
\]

(2.14)

If the deviation of the input membership function \( B_{ij} \) is:

\[
|B_{ij}(x_j + \delta_{ij}) - B_{ij}(x_j)| = |\delta_{ij}| \leq h_F, \quad i = 1, 2, \quad j = 1
\]

(2.15)

then the deviation of the fuzzy output is represented as:

\[
e(\delta_{ij}) = \max_y |E(y; B_{ij} + \delta_{ij}) - E(y; B_{ij})|, \quad y \in \mathbb{R}
\]

(2.16a)

and the modulus of continuity of the output is:
\[
\gamma(h_F) = \max_{B_{ij}, \delta_i} \left| e(B_{ij}, \delta_i) \right| \quad i = 1, 2 \quad j = 1
\]

(2.16b)

By using equation 2.1 for the fuzzy output, the function \(e\) can be written as follows:

\[
e(B_{ij}, \delta_i) = \max_\gamma \left\{ \beta \left[ F_L \left( y; B_{ij} + |\delta_i| \right) - F_L \left( y; B_{ij} \right) \right] + (1 - \beta) \left[ F_M \left( y; B_{ij} + |\delta_i| \right) - F_M \left( y; B_{ij} \right) \right] \right\}
\]

(2.17)

An upper bound of the function \(e\) can be readily obtained as:

\[
e(B_{ij}, \delta_i) \leq \beta \left( \max_\gamma \left\{ \left| F_L \left( y; B_{ij} + |\delta_i| \right) - F_L \left( y; B_{ij} \right) \right| \right\} + (1 - \beta) \left( \max_\gamma \left\{ \left| F_M \left( y; B_{ij} + |\delta_i| \right) - F_M \left( y; B_{ij} \right) \right| \right\} \right)
\]

(2.18)

For the SISO system in equation 2.14, and a crisp input \(x_1^*\), the Logical and Mamdani’s formulations in equation 2.18 are as follows:

\[
F_M \left( y; B_{ij} \right) = S_p \left( \tilde{\delta}(y), \phi(y) \right)
\]

(2.19)

\[
F_M \left( y; B_{ij} + \delta_i \right) = S_p \left( \tilde{\delta}(y), \tilde{\phi}(y) \right)
\]

(2.20)

\[
F_L \left( y; B_{ij} \right) = 1 - S_p \left( \psi(y), \eta(y) \right)
\]

(2.21)

\[
F_L \left( y; B_{ij} + \delta_i \right) = 1 - S_p \left( \tilde{\psi}(y), \tilde{\eta}(y) \right)
\]

(2.22)

where

\[
\tilde{\delta}(y) = T_p \left( B_{ij}(x_1^*), D_1(y) \right), \quad \tilde{\phi}(y) = T_p \left( B_{ij}(x_1^*), |\delta_{ij}|, D_1(y) \right)
\]

(2.23)

\[
\phi(y) = T_p \left( B_{ij}(x_1^*), D_1(y) \right), \quad \tilde{\phi}(y) = T_p \left( B_{ij}(x_1^*), |\delta_{ij}|, D_1(y) \right)
\]

\[
\psi(y) = T_p \left( B_{ij}(x_1^*), 1 - D_1(y) \right), \quad \tilde{\psi}(y) = T_p \left( B_{ij}(x_1^*), |\delta_{ij}|, 1 - D_1(y) \right)
\]

\[
\eta(y) = T_p \left( B_{ij}(x_1^*), 1 - D_2(y) \right), \quad \tilde{\eta}(y) = T_p \left( B_{ij}(x_1^*), |\delta_{ij}|, 1 - D_2(y) \right)
\]

Through the following theorems, we attempt to establish a relation between the modulus of continuity of the fuzzy output \(\gamma_c\) and the maximum deviation of the input membership grades \(h_F\). We will notify that this relation is parameterized by the inference parameters \(p\) and \(q\). Hence by adjusting these parameters, we can achieve the desirable marginal continuity of the inference mechanism. This will be elaborated in the sequel. Through Theorems 1 and 2, the upper limit in inequality 2.18 is related to \(h_F\). Then, the modulus of continuity is related to \(h_F\) in Theorem 3. Theorem 4 is an extension to multi-input single-output systems.
Theorem 1:

For the following real continuous functions:

\[ 0 \leq f(y) \leq 1 : 0 \leq g(y) \leq 1 \quad y \in \mathbb{R} \]

defined over the real interval \([0,1]\), the following inequality holds for all \(y \in \mathbb{R}\) having the specific parametric \(t\)-conorm operator selected in this investigation when \(p \geq 0\).

\[
\left| S_p \left( \tilde{f}(y) - f(y), \tilde{g}(y) - g(y) \right) \right| \leq S_p \left( |\tilde{f}(y) - f(y)|, |\tilde{g}(y) - g(y)| \right) \tag{2.24}
\]

where

\[ \tilde{f}(y) = f(y) + \Delta f : \tilde{g}(y) = g(y) + \Delta g \]

and

\[ 0 < \tilde{f}(y), \tilde{g}(y) \leq 1 \]

and \(S_p\) is the parameterized \(t\)-conorm introduced in equation 1.4.

Proof: We demonstrate the inequality (2.24) for the extreme cases of the parameter \(p\).

i.e., \(p \to \infty\) and \(p \to 0\).

(a) \(p \to \infty\): the \(s\)-norm will become a max operator. then we have to show that:

\[
\max_{L(y)} \left( \frac{\left| \max_{x} \left( \tilde{f}(y) - f(x), \tilde{g}(y) - g(x) \right) \right|}{H(y)} \right) \leq \max_{L(y)} \left( \frac{\left| \tilde{f}(y) - f(x), \tilde{g}(y) - g(x) \right|}{H(y)} \right) \tag{2.25}
\]

For each \(y\), the function \(L(y)\) can be either of the following four possibilities:

(i) \( \max_{x} \left| f(y) - \tilde{f}(y) \right| \):

(ii) \( \max_{x} \left| f(y) - \tilde{g}(y) \right| \):

(iii) \( \max_{x} \left| g(y) - \tilde{f}(y) \right| \):

(iv) \( \max_{x} \left| g(y) - \tilde{g}(y) \right| \).

For the possibilities (i) and (iv), we have:

\[
L(y) \leq \max_{x} \left( \left| f(y) - \tilde{f}(y) \right|, \left| g(y) - \tilde{g}(y) \right| \right) = H(y) \tag{2.26}
\]

For the possibility (ii), we can deduce:

\[
f(y) \geq g(y) \]

\[
\tilde{f}(y) \leq \tilde{g}(y) \tag{2.27}
\]
then
\[ L(y) = |f(y) - \tilde{g}(y)| \leq |f(y) - \tilde{f}(y)| \leq \max \left( |f(y) - \tilde{f}(y)|, |g(y) - \tilde{g}(y)| \right) = H(y) \quad (2.29) \]

For the possibility (iii), we can deduce:

\[
\begin{align*}
  f(y) &\leq g(y) \\
  \tilde{f}(y) &\geq \tilde{g}(y)
\end{align*}
\]

then
\[ L(y) = |g(y) - \tilde{f}(y)| \leq |g(y) - \tilde{g}(y)| \leq \max \left( |f(y) - \tilde{f}(y)|, |g(y) - \tilde{g}(y)| \right) = H(y) \quad (2.30) \]

from equations 2.26, 2.28, and 2.30, we conclude

\[ L(y) \leq H(y) \quad \text{when} \quad p \to \infty \quad \text{for all} \quad y \in \mathbb{R} \]

(b) \( p \to 0 \)

For this specific case, each of the \( t \)-conorms in inequality 2.24 will become a drastic \( t \)-conorm in the following form (Emami, 1996a):

\[ S_w(f, g) = \begin{cases} 
  a \lor b & \text{if } a \land b = 0 \\
  1 & \text{otherwise}
\end{cases} = \begin{cases} 
  a & \text{if } b = 0 \\
  b & \text{if } a = 0 \\
  1 & \text{otherwise}
\end{cases} \]

For the inequality 2.24, we have one of the following possibilities:

1. \( f = 0, g = 0, \tilde{f} = 0, \text{and} \tilde{g} = 0 \)  
2. \( f \neq 0, g \neq 0, \tilde{f} \neq 0, \text{and} \tilde{g} \neq 0 \)
3. \( f = 0 \)  
4. \( \tilde{f} = 0 \)  
5. \( g = 0 \)  
6. \( \tilde{g} = 0 \)
7. \( f = 0, \text{and} \ g = 0 \)  
8. \( f = 0, \text{and} \ \tilde{f} = 0 \)  
9. \( f = 0, \text{and} \ \tilde{g} = 0 \)  
10. \( g = 0, \text{and} \ \tilde{f} = 0 \)
11. \( g = 0, \text{and} \ \tilde{g} = 0 \)  
12. \( f = 0, \text{and} \ \tilde{f} = 0, \text{and} \ \tilde{g} = 0 \)  
13. \( g = 0, \text{and} \ \tilde{g} = 0, \text{and} f = 0 \)
14. \( g = 0, \text{and} \ \tilde{g} = 0, \text{and} \ \tilde{f} = 0 \)  
15. \( f = 0, g = 0, \text{and} \ \tilde{g} = 0 \)

- for case 1: L.H.S = 0 and R.H.S = 0
- for case 2: L.H.S = |l - l| = 0 and R.H.S = 1
- for case 3: L.H.S = |l - g| and R.H.S = 1

where cases from 4-6 are also equivalent to case 3.

- for case 7: L.H.S = 1 and R.H.S = 1
- for case 8: L.H.S = |\tilde{g} - g| and R.H.S = |\tilde{g} - g|

where cases 9-11 are equivalent to case 8.
• for case 12: L.H.S = |g| and R.H.S = |g|

where cases 13-15 are equivalent to case 12.

Therefore, we can conclude that for inequality 2.24:

\[ L(y) \leq H(y) \text{ when } p \to 0 \text{ for all } y \in \mathbb{R} \]

By inspection, we realize that the curves of the L.H.S and R.H.S of inequality 2.24 never intersect for any value of \( p \) regardless of the functions \( f, \tilde{f}, \text{ and } \tilde{g} \). Therefore, from the extreme cases of \( p \) and the above fact we conclude:

\[ |S_p(\tilde{f}(y), \tilde{g}(y)) - S_p(f(y), g(y))| \leq S_p(|\tilde{f}(y) - f(y)|, |\tilde{g}(y) - g(y)|) \]

for all \( y \in \mathbb{R} \) and for all values of \( p \).

**Theorem 2:**

Referring to equations 2.23, the following inequalities hold:

\[
\begin{align*}
|\tilde{\vartheta}(y) - \vartheta(y)| &\leq h_{\vartheta}, \\
|\tilde{\phi}(y) - \phi(y)| &\leq h_{\phi}.
\end{align*}
\]

\( y \in \mathbb{R} \)

**Proof:** Considering the duality property of the \( t \)-norm and \( t \)-conorm (Smets, 1982), the inequality 2.26 of Theorem 1 can be restated as follows:

\[ T(1 - \tilde{f}(y).1 - \tilde{g}(y)) - T(1 - f(y).1 - g(y)) \leq S\left(|\tilde{f}(y) - f(y)|, |\tilde{g}(y) - g(y)|\right) \quad (2.31) \]

We use the following substitutions:

\[
\begin{align*}
1 - f(y) &= B_{11}, \quad 1 - \tilde{f}(y) = B_{11} + |\delta_{11}| \\
1 - g(y) &= D_{1}(y), \quad 1 - \tilde{g}(y) = D_{1}(y)
\end{align*}
\]

Then inequality 2.31 can be rewritten as:

\[ |T(B_{11} + \delta_{11}, D_{1}(y)) - T(B_{11}, D_{1}(y))| \leq S(\left|\delta_{11}\right|, 0) = |\delta_{11}| \quad (2.32) \]

Using the notations in equation 2.25, we have:

\[ |\tilde{\vartheta}(y) - \vartheta(y)| \leq |\delta_{11}| \leq h_{\vartheta} \quad \forall \ y \in \mathbb{R} \quad (2.33) \]

If we use the following substitution in inequality 2.31:

\[
\begin{align*}
1 - f(y) &= B_{11}, \quad 1 - \tilde{f}(y) = B_{11} + |\delta_{11}|
\end{align*}
\]
$1 - g(y) = D_i(y)$, $1 - \bar{g}(y) = D_i(y)$

in the same way, we will have:

$$\left| \bar{\phi}(y) - \phi(y) \right| \leq h_F \quad \forall y \in \mathbb{R}$$  \hspace{1cm} (2.34)

Theorems 1 and 2 were proved for Mamdani's part of the inference formulation, i.e., $\vartheta(y)$ and $\phi(y)$. The same results are obtained for the functions $\psi(y)$ and $\eta(y)$ due to their similarity to the Mamdani's function. Hence:

$$\left| \bar{\psi}(y) - \psi(y) \right| \leq h_F \quad \forall y \in \mathbb{R}$$  \hspace{1cm} (2.35)

Using Theorems 1 and 2, we can now in Theorem 3 introduce a relation between the maximum deviation of the input membership grades and the deviation of the fuzzy output:

**Theorem 3:**

$$r_e \leq S_p(h_F, h_F)$$  \hspace{1cm} (2.36)

**Proof:** Using Theorem 2 and considering the monotonicity property of the $t$-conorms, we have:

$$S_p \left( \left| \bar{\vartheta}(y) - \vartheta(y) \right| \left| \bar{\phi}(y) - \phi(y) \right| \right) \leq S_p \left( h_F, h_F \right)$$  \hspace{1cm} (2.37)

and

$$S_p \left( \left| \bar{\psi}(y) - \psi(y) \right| \left| \bar{\eta}(y) - \eta(y) \right| \right) \leq S_p \left( h_F, h_F \right)$$  \hspace{1cm} (2.38)

Hence from Theorem 1:

$$S_p \left( \bar{\vartheta}(y), \bar{\phi}(y) \right) - S_p \left( \vartheta(y), \phi(y) \right) \leq S_p \left( h_F, h_F \right)$$  \hspace{1cm} (2.39)

and

$$S_p \left( \bar{\psi}(y), \bar{\eta}(y) \right) - S_p \left( \psi(y), \eta(y) \right) \leq S_p \left( h_F, h_F \right)$$  \hspace{1cm} (2.40)

Substituting equations 2.21-2.24 into inequalities 2.39 and 2.40 results in:

$$\left| F_M(y; B_\delta + \delta \eta) - F_M(y; B_\eta) \right| \leq S_p \left( h_F, h_F \right)$$  \hspace{1cm} (2.41)

and

$$\left| F_L(y; B_\delta + \delta \eta) - F_L(y; B_\eta) \right| \leq S_p \left( h_F, h_F \right)$$  \hspace{1cm} (2.42)

By multiplying inequalities 2.41 and 2.42 by $(1 - \beta)$ and $\beta$, respectively and summing up the results, we have:
CHAPTER 2
ROBUSTNESS OF THE FUZSY INFERENCE MECHANISM

\[ \beta \left( F_L(y; B_{ij} + \delta_{ij}) - F_L(y; B_{ij}) \right) + \\
(1 - \beta) \left( F_M(y; B_{ij} + \delta_{ij}) - F_M(y; B_{ij}) \right) \leq S_p(h_F, h_F) \]  

(2.43)

From inequalities 2.17 and 2.43, we obtain:

\[ e(B_{ij}, \delta_{ij}) \leq S_p(h_F, h_F) \]  

(2.44)

and hence:

\[ \gamma_{e} = \max_{B_{ij}, \delta_{ij}} |e(B_{ij}, \delta_{ij})| \leq S_p(h_F, h_F) \]  

(2.45)

Theorem 3 states that for a maximum deviation of \( h_F \) in the input membership grades, the modulus of continuity of the fuzzy output is always less than or equal to the \( t \)-conorm of \( h_F \).

2.4.2 Multi-Input Single-Output Systems

Consider the multi-input single-output system in the following form:

<table>
<thead>
<tr>
<th>IF ( x_1 ) is ( B_{11} ) AND ( x_2 ) is ( B_{12} ) AND ... ( x_r ) is ( B_{1r} ) THEN ( y ) is ( D_1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALSO</td>
</tr>
<tr>
<td>......</td>
</tr>
<tr>
<td>ALSO</td>
</tr>
<tr>
<td>IF ( x_1 ) is ( B_{n1} ) AND ( x_2 ) is ( B_{n2} ) AND ... ( x_r ) is ( B_{nr} ) THEN ( y ) is ( D_n )</td>
</tr>
</tbody>
</table>

(2.46)

where \( x = [x_1, x_2, ..., x_r] \) is the vector of the significant input parameters, \( r \) is the number of antecedents, \( n \) is the number of rules, \( y \) is the model output, \( B_{ij} \) \((i = 1, ..., n; j = 1, ..., r)\) and \( D_i \) \((i = 1, ..., n)\) are fuzzy sets of the universes of discourse \( X_1, X_2, ..., X_r \) and \( Y \), respectively. For the multi-input single-output system in 2.46 and a crisp input vector \( x_c = [x_1^c, x_2^c, ..., x_r^c] \), the fuzzy output can be defined as:

\[ E(y) = \beta F_L(y) + (1 - \beta) F_M(y) \]  

(2.47)

where

\[ F_M(y; B_{ij}) = S_p(\delta_1(y), \delta_2(y), ..., \delta_n(y)) \]  

(2.48)

\[ F_L(y; B_{ij}) = 1 - S_p(\psi_1(y), \psi_2(y), ..., \psi_n(y)) \]  

(2.49)
and the deviation of the above functions are represented as:

\[
\tilde{F}_L(y; B_i; + \delta_i) = 1 - S_p(\tilde{\psi}_1(y), \tilde{\psi}_2(y), \ldots, \tilde{\psi}_n(y)).
\]

(2.51)

where

\[
\tilde{\psi}_i(y) = T_p(\tau, 1 - D_i(y))
\]

(2.55)

and

\[
\tau = T_q(B_1(x_1^*), B_2(x_2^*), \ldots, B_n(x_n^*))
\]

(2.56)

\[
\tau + \delta = T_q(B_1(x_1^*), B_2(x_2^*), \ldots, B_n(x_n^*), \delta)
\]

(2.57)

where \(|\delta_i| \leq h_F\) and \(|\delta| \leq h_r\).

Now, we can state Theorem 4 that represents the extension of Theorem 3 to a multi-
input single-output system as follows:

**Theorem 4:**

\[
\gamma_c = \max_{B_i, \delta_i} e(B_i; \delta_i) \leq S_p(h_z, h_{z-1}, \ldots, h_{z-n})
\]

(2.58)

where \(h_z \leq S_{q}(h_F, h_{F-1}, \ldots, h_F)\)

(2.59)

**Proof:** First, we extend the result of Theorem 1 to the \(n\)-ary case using the associativity
and the monotonicity properties of the parametric \(t\)-conorm (Emami, 1996a):

\[
|S(f_1, f_2, \ldots, f_r) - S(f_1, f_2, \ldots, f_{r-1})| \leq S|S(f_1, f_2, \ldots, f_{r-1}) - S(f_1, f_2, \ldots, f_{r-1})| |f_r - f_r| (2.60)
\]

Likewise, we can write:
|S(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_{r-1}) - S(f_1, f_2, \ldots, f_{r-1})| = \\
|S[S(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_{r-2})] - S[S(f_1, f_2, \ldots, f_{r-2})]| \\ 
\leq S\left|\left|S(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_{r-2}) - S(f_1, f_2, \ldots, f_{r-2})\right|\right|_1 \leq |\bar{f}_{r-1} - f_{r-1}| \\
(2.61)

From 2.60 and 2.61 and the monotonicity property of the $t$-conorm, we have:

|S(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_r) - S(f_1, f_2, \ldots, f_r)| \\
\leq S\left|\left|S(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_{r-2}) - S(f_1, f_2, \ldots, f_{r-2})\right|\right|_1 \leq |\bar{f}_{r-1} - f_{r-1}| \leq |\bar{f}_r - f_r| \\
= S\left|\left|S(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_{r-2}) - S(f_1, f_2, \ldots, f_{r-2})\right|\right|_1 \leq |\bar{f}_{r-1} - f_{r-1}| \leq |\bar{f}_r - f_r| \\
(2.62)

Following the same procedure, we conclude:

\left|S_p(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_r) - S_p(f_1, f_2, \ldots, f_r)\right| \\
\leq S_p\left|\left|S_2(\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_r) - S(f_1, f_2, \ldots, f_r)\right|\right|_1 \leq |\bar{f}_{r-1} - f_{r-1}| \leq |\bar{f}_r - f_r| \\
(2.63)

From Theorem 2, we also have:

\left|\bar{\varphi}_i(y) - \varphi_i(y)\right| \leq h_i \quad i = 1, 2, \ldots, r \\
(2.64)

where $\bar{\varphi}_i$ and $\varphi_i$ are defined in equations 2.52 and 2.53.

Next, by using Theorem 3 and the associativity property, we have:

\left|S_p(\bar{\varphi}_1(y), \bar{\varphi}_2(y), \ldots, \bar{\varphi}_n(y)) - S_p(\varphi_1(y), \varphi_2(y), \ldots, \varphi_n(y))\right| \\
\leq S_p(h_1, h_2, \ldots, h_r) \\
(2.65)

and likewise:

\left|S_p(\bar{\psi}_1(y), \bar{\psi}_2(y), \ldots, \bar{\psi}_n(y)) - S_p(\psi_1(y), \psi_2(y), \ldots, \psi_n(y))\right| \\
\leq S_p(h_1, h_2, \ldots, h_r) \\
(2.66)

Substituting equations 2.19-2.22 into inequalities 2.65 and 2.66 results in:

\left|F_M(y; B_{i_1} + \delta_{i_1}) - F_M(y; B_{i_1})\right| \leq S_p(h_1, h_2, \ldots, h_r) \\
(2.67)

and

\left|F_L(y; B_{i_1} + \delta_{i_1}) - F_L(y; B_{i_1})\right| \leq S_p(h_1, h_2, \ldots, h_r) \\
(2.68)

Following the same procedure as Theorem 3, we conclude:

$\gamma_e = \max_{\delta_{i_1}, \delta_{i_2}}(\epsilon(B_{i_1}, \delta_{i_1})) \leq S_p(h_1, h_2, \ldots, h_r) \\
(2.69)

In order to relate $\epsilon$ to $h_F$, from equations 2.65 and 2.52 - 2.55, we have:

$h_i = \left|T_q(B_{i_1} + h_F, B_{i_2} + h_F, \ldots, B_{i_r} + h_F) - T_q(B_{i_1}, B_{i_2}, \ldots, B_{i_r})\right| = 1, 2, \ldots, r \\
(2.70)$
Then, using the dual form of the inequality 2.63 (having \( q \) as the norm operator), we have:

\[
|h_r| \leq S_q(h_F, h_F, \ldots, h_F)
\]  

(2.71)

Inequalities 2.69 and 2.71 indicate a relationship between the maximum deviation of the input membership grade, \( h_F \), and the modulus of continuity of the fuzzy output, \( \gamma \). This relationship contains the parametric \( t \)-conorms \( S_p \) and \( S_q \) that bring the parameters \( p \) and \( q \) into the picture. In fuzzy-logic modeling, our primary interest is to allow the maximum possible deviation of the input membership grades such that the maximum deviation of the fuzzy output over \( y \in \mathbb{R} \) does not exceed a certain amount \( \gamma \). Hence, we have to obtain the resultant input deviation through the following equation:

\[
\gamma_c = S_p \left( \underbrace{S_q(h_F, h_F, \ldots, h_F)}_{r \text{ times}}, \underbrace{S_q(h_F, h_F, \ldots, h_F)}_{n \text{ times}} \right)
\]  

(2.71b)

The question is what are the values of \( p \) and \( q \) that maximize \( h_F \) for a specific \( \gamma \). The equation 2.71b indicates that for a constant \( \gamma \), \( h_F \) is a monotonic function of \( p \) and \( q \). Thus, by increasing \( p \) and \( q \), the allowable input deviation increases. Therefore, the sensitivity of the model reduces. At the limits, when both \( p \) and \( q \) are at infinity, the maximum allowable deviation of the input membership grades will reach \( \gamma \). This will be illustrated in the example in section 2.7.

From the above analysis, we are able to prevent the so-called "hyper-sensitivity" of the inference mechanism to the input deviation by choosing \( p \) and \( q \) adequately high. In other words, for a desirable maximum fuzzy output deviation, we can specify the lower bound of \( p \) and \( q \) such that this output deviation will not occur for input deviations less than a certain value. This will give the lowest value of \( h_F \). This way, we adjust the sensitivity (robustness) of the inference mechanism to the deviation of the input membership grades. In summary, \( h_F \) has an upper bound equal to \( \gamma \) that occurs when \( p \) and \( q \) approach infinity. We set the minimum \( h_F \) and then obtain the lower bounds of \( p \) and \( q \) from equation 2.71b, given a specified modulus of continuity of the fuzzy output.
2.5 Marginal Continuity of the Crisp Output

Since we are practically interested in the crisp output, the concept of the marginal continuity is extended to the crisp output. The deffuzzified output for a range of \( y \in [y_u, y_f] \) is computed as follows (equation 2.7) (Yager, 1994):

\[
y^{*} = \frac{\int_{y_u}^{y_f} y[E(y)]^\alpha \, dy}{\int_{y_u}^{y_f} [E(y)]^\alpha \, dy}
\]

We assign the maximum deviation of the crisp output as (we use the notation \( \sim \) to denote for the deviation of the crisp output):

\[
\gamma_c \geq \max_{[y_u,y_f]} |y^{*} - y^{\sim}|.
\]

The domain of defuzzification parameter \( \alpha \) (equation 2.7) should be selected such that the crisp output deviation never exceeds the specified value \( \gamma_c \). We discuss the procedure in the next section.

2.6 Assignment of Parameters

The last part of the marginal continuity analysis is the selection of appropriate values of \( \gamma_c \), \( \gamma_e \) and the minimum deviation of the input grade from the training data set. The value of \( \gamma_c \) can be set as a small percentage of the entire output domain. In this work, we assign \( \gamma_c \) to be 5% of the entire range of the output. For identifying a suitable value of \( \gamma_c \), we first calculate the following "mutual ratio" for all data points:

\[
t^{ij} = \frac{\max_k \left| B_{x_i}^j (x_i^k) - B_{x_i}^j (x_i^l) \right|}{|y^{*} - y^{\sim}|} : k = 1,2,\ldots, n; l = 1,2,\ldots, r; i,j = 1,2,\ldots, N
\]

where \( N \) is the number of data in the training set, \( n \) is the number of rules, and \( r \) is the number of input variables. The ratio \( t^{ij} \) represents the mutual "jump" between the data, in a sense that a lower ratio illustrates a higher jump in the crisp output with less change in the input membership grades. Therefore, the minimum of \( t^{ij} \) can be a good index for calculating the maximum deviation of the input membership grades. As soon as we assign a value for the maximum deviation of the crisp output, i.e., \( \gamma_c \), the maximum
possible deviation of the input membership grades that could occur in the data is set to be:

\[ \gamma_c \times \min(t^u) \]  \hspace{1cm} (2.74)

From equation 2.71b, knowing that \( \gamma_c \) is always larger or equal to the maximum input deviation \( h_F \) as explained in page 33 (Paragraph 3), we can set:

\[ \gamma_c = \gamma_c \times \min(t^u) \]  \hspace{1cm} (2.75)

In order to eliminate the noise effect in the above calculation, we consider the values of \( t^u \) within the range of 25 % to 75 % of its entire domain. The minimum of \( h_F \) is a percentage of \( \gamma_c \), and depends on the data. In this research, we choose this value equal to 20-50% of \( \gamma_c \).

2.7 Case Study

Example 2.1

In (Sugeno, 1993), Sugeno and Yasukawa introduce the nonlinear static system shown in figure 2.4. The system has two inputs \( x_1 \) and \( x_2 \) and a single output \( y \) as follows:

\[ y = (1 + x_1^{-2} + x_2^{-1.5})^3; \quad 1 \leq x_1, x_2 \leq 5 \]  \hspace{1cm} (2.76)

Fig 2.4: Sugeno-Yasukawa deterministic function
50 input–output data are generated as the training set, and the output data were clustered by means of the FCM algorithm (Emami, 1996a). The optimum number of clusters is \( c = 8 \), for a weight exponent \( m = 3 \). After developing the output fuzzy clusters and performing the classification for the entire output space, the next step is to select and tune the set of parameters of the fuzzy model. These parameters are the inference parameters \( (p, q, \beta, \alpha) \) and the input and output membership parameters. Using the unified parameterized reasoning formulation introduced in section 1.3.1 (Equation 1.1), at the first step of parameter identification, the optimum inference parameters are identified. The optimum values are (Emami, 1996a):

\[
p = 19.9959 : q = 0.4536 : \beta = 0.0255 : \alpha = 5.3715.
\]  

(2.77)

Fig 2.5: Input-output rules of the original model (Emami, 1996a)
By optimizing the inference parameters, the fuzzy model performance index (equation 1.14) was obtained as $PI = 0.171$. The second step of parameter identification is to adjust the input-output membership parameters. After 10 iterations, the performance index becomes $PI = 0.004$. Figure 2.5 shows the set of input–output rules for the developed fuzzy model. Figure 2.6 shows a comparison between the output of the developed fuzzy model and the function output for the training set.

The methodology of achieving the appropriate marginal continuity is applied to the same training set. The idea is to investigate the behavior of the model subject to the conditions used to guarantee the marginal continuity. $\gamma_c$ is assigned as 5% of the entire output range. The following values were obtained from the data:

$$\gamma_c = 0.3750 : \min(\gamma^d) = 0.0897 : \gamma_c = 0.0336$$  \hspace{1cm} (2.78)

Equation 2.69, and equation 2.71 were used to determine the relation between the input membership function deviation, i.e., $h_F$, and both parameters $p$ and $q$. The variation of $h_F$ with the parameter $p$ and $q$ is shown in figure 2.7. The variation of the parameter $p$ with the parameter $q$ is shown in figure 2.8. The selected value for the minimum deviation of the input membership function is $h_F = 0.015$, for which the lower bounds of the reasoning parameters are chosen at point $x$ in figure 2.7. As shown in fig 2.8, we have infinite combinations for the values of $p$ and $q$ for which $h_F = 0.015$. We choose point $x$ such that both $p$ and $q$ are as far as possible from zero. Hence for point $x$, the parameters have the following values:

$$p_{\text{min}} = 2.9 : q_{\text{min}} = 3.7$$ \hspace{1cm} (2.79)

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig2_6.png}
\caption{A comparison between the function and the original model}
\end{figure}
CHAPTER 2
ROBUSTNESS OF THE FUZZY INFERENCE MECHANISM

Fig 2.7: Variation of the input deviation with both $p$ and $q$

Fig 2.8: Selection of the lower bounds of $p$ and $q$
Fig 2.9: Variation of the deviation in the crisp output with parameter $\alpha$

Fig 2.10: The tuning process of the robust model training set
Using the assigned value of $\gamma$ and the condition in equation 2.72, marginal continuity of the crisp output of the model is achieved when $\alpha$ is selected in the range $[0.89, \infty)$. Figure 2.9 shows the variation of crisp output deviation i.e., $|\hat{y} - \gamma|$ for different values of the parameter $\alpha$.

After assigning the lower bound of parameters $p$ and $q$ and the range of parameter $\alpha$, the optimum inference parameters are identified. The optimum values are:

$$p = 19.549 ; q = 13.42 ; \beta = 0.0001 ; \alpha = 1.0518.$$  \hspace{1cm} (2.80)

By optimizing the inference parameters, the fuzzy model performance index (equation 1.14) is $PI = 0.2608$. As mentioned earlier, the second step of parameter identification is to adjust the input output membership parameters. Figure 2.10 shows the tuning process of the training set. After 10 iterations, iteration $k = 4$ is chosen. The performance index of the training set is $PI = 0.0162$. Figure 2.11 shows a comparison between the developed fuzzy model for training set and Sugeno’s function (equation 2.76).

Fig 2.11: A comparison between the function and the robust model
Figure 2.12 shows the set of input-output rules for the developed fuzzy model. For five different estimation data sets, a comparison between the original model (Emami, 1996a) and the new model is introduced. Figures 2.13, 2.14, 2.15, 2.16, and 2.17 show the comparison between the performance of both models for the different estimation data sets considered. Table 2.1 shows the results of the comparison.

**Fig 2.12: Input-output rules of the robust model**
From the results obtained, we can see that the prediction error of the robust model is less than that of the original model for two testing data sets (data sets 2 and 4). However, the error of the robust model is higher for the rest of the testing sets. That is due to the marginal continuity property encountered in this analysis. In order to achieve the required robustness of the fuzzy inference mechanism, we have to assign lower bounds of $p$ and $q$ at one stage of the robustness analysis. Thus, the lower bounds of $p$ and $q$ are considered as constraints to avoid the "hyper-sensitivity" of the model. In the parameter identification process, the optimum reasoning parameters are free to vary between the bounds specified for each of them and not between zero and infinity (for $p$ and $q$). This explains the higher error we get for some of the testing sets. On the other hand, using the developed robust model, robustness of the fuzzy inference mechanism will be achieved.
Fig 2.13: A comparison between the performance of the original model (A) [Emami, 1996a] and the robust model (B) for the testing set I.
Fig 2.14: A comparison between the performance of the original model (A) and the robust model (B) for the testing set 2.
ROBUSTNESS OF THE FUZZY INFERENCE MECHANISM

Fig 2.15: A comparison between the performance of the original model (A) and the robust model (B) for the testing set 3
Fig 2.16: A comparison between the performance of the original model (A) and the robust model (B) for the testing set 4
Fig 2.17: A comparison between the performance of the original model (A) and the robust model (B) for the testing set 5.
CHAPTER 2
ROBUSTNESS OF THE FUZZY INFERENCE MECHANISM

2.8 Conclusions

In this chapter, the marginal continuity of the robust fuzzy model is investigated. A fuzzy model must be robust such that small changes of the input lead to small changes of the output. In order to investigate this property, the following tasks were performed in this chapter: (i) an index for the deviation of the input membership functions was introduced; (ii) the modulus of continuity from the approximation theory was used as a measure of the deviation of the fuzzy output; (iii) the relationship between the maximum deviation of the input membership function and the reasoning parameters $p$ and $q$ was studied; (iv) the concept of the marginal continuity was extended to the model crisp output; (v) a technique to assign $\gamma_F$ and $\gamma_c$ from the data was introduced; and (vi) four theorems to guarantee conditions for the marginal continuity were developed.

The developed methodology was applied to a well known example in the literature (Sugeno, 1993). For this example, a comparison between the original model (Emarni, 1996a) and the robust model was performed. The behavior of the robust model for different testing data is acceptable compared to the original model. However, for some testing sets, the robust model shows a higher error. That is due to the conditions applied to the reasoning parameters to guarantee the marginal continuity, i.e., lower bounds of $p$ and $q$, and the range of $\alpha$. On the other hand, using the developed robust model, no unwanted "jumps" in the model output (due to the deviation of the input membership functions) are expected. Finally, the extension of the fuzzy model from the training set to the new testing sets (which have not been used in the model construction) would still be a part of the fuzzy structure identification. This issue will be discussed in details in Chapter 4.
3.1 Introduction

An important step of fuzzy modeling is the fuzzy rule generation. The system data is partitioned into fuzzy clusters. We can define clustering as partitioning of a group of unlabeled data into a number of clusters such that similar data is assigned to one cluster and data that is less similar is assigned to different clusters (Emami, 1996b).

Two main approaches to clustering are typically used: hard clustering and fuzzy C-Means Clustering. Hard clustering assigns each data point to one and only one cluster with a membership grade equal to one, assuming fully defined boundaries between clusters. Practically, the boundaries between clusters can not be clearly defined. As an alternative, the fuzzy C-Means (FCM) clustering algorithms were suggested by Bezdek (Bezdek, 1981).

In this Chapter, the robustness of the fuzzy clustering algorithms is studied. A robust fuzzy model should identify and reject the noise from the training set and eliminate its effect during the system identification, parameter adjustment, and tuning. Although hard and fuzzy C-Means clustering algorithms are used in many applications, they are highly sensitive to the presence of noise and outliers. The FCM algorithm uses the sum of squared errors in its objective function. Thus, this clustering method may fail completely in the presence of noise. As an alternative, a modified version of the Possibilistic Clustering Algorithm (PCM) was introduced by Krishnapuram et al. (Krishnapuram, 1996) to handle the noisy data sets. The PCM clustering algorithm is more robust than the original FCM algorithm in the presence of noise because its objective function involves
unconstrained weights that decrease with the distance from the cluster centers. This will result in low weights for the noise points and therefore reduces their effect on the data set. However, this algorithm encounters the same problems as the original FCM algorithms in the sense that some parameters must be selected \textit{a priori}. Our approach to achieve robustness is based on combining the improved PCM clustering algorithm in (Krishnapuram, 1996) with the technique by (Emami et al., 1996b) to overcome the original FCM and the PCM algorithms drawbacks. In Section 3.2, we review the fuzzy C-Means clustering algorithm. We also study the proposed solutions to overcome the original FCM drawbacks. In Section 3.3, we present the noise rejection clustering algorithm (Krishnapuram, 1996). In Section 2.4, we introduce the robust clustering algorithm as a combination of the two above-mentioned clustering approaches. In Section 2.5, we apply the proposed methodology to four different examples. Finally, in Section 2.6, we discuss the conclusions.

\textbf{3.2 The Fuzzy C-Means Clustering Algorithm}

The FCM is the most commonly used algorithm for data partitioning. Due to the nature of our applications. i.e., MISO systems, a simpler and applicable FCM clustering is adopted (Sugeno, 1993). In this approach, the output data is clustered as a single-dimensional output space. The fuzzy partitions for the input space can then be specified by projecting the output clusters on each of the input variables, separately (Sugeno, 1993).

For a set of unlabeled data \( x = \{x_1, x_2, ..., x_N\} \subseteq \mathbb{R}^L \), where \( N \) is the number of data points and \( s \) is the dimension of each data feature, the fuzzy clustering is the assignment of \( c \) number of partition labels to the features in \( x \). \( c \)-partition of \( x \) are sets of \((c,N)\) membership values \( \{u_{ik}\} \) that can be arranged as a \((c \times N)\) matrix \( U = [u_{ik}] \). The major step in fuzzy clustering is to find the optimum membership matrix \( U \). The FCM uses the weighted within-groups sum of squared errors objective function \( J_m \) by defining the following optimization problem:

\[
\min_{U, V} \left\{ J_m(U, V; X) = \sum_{k=1}^{c} \sum_{i=1}^{N} (u_{ik})^m \| x_i - v_k \|_A^2 \right\}
\]

(3.1)
where $v = \{v_1, v_2, \ldots, v_c\}$ is the vector of the cluster centers. The matrix $A$ specifies the shape of the clusters. For spherical clusters, $A$ is chosen as an identity matrix. The membership matrix is calculated as:

$$u_{ikt} = \left[ \sum_{j=1}^{c} \left( \frac{\|x_k - v_{lt}\|_A}{\|x_k - v_{lt}\|_A} \right)^{\frac{2}{m-1}} \right]^{-1}$$

where $t$ is the iteration number in the iterative optimization, and $m$ is the weighting exponent.

The original FCM clustering algorithm has three major problems:

- In order to get the optimal partition, initial locations of the cluster centers should be assigned. The (FCM) algorithm always converges to a local extreme of $J_m$ (weighted within-groups sum of squared errors objective function). Different choices of initial cluster centers may lead to a different extrema.

- The scientific basis for the choice of the weight exponent is still not clear.

- The optimum number of clusters in the data is assigned a priori. There should be a criterion to assign the optimal number of clusters.

In the clustering algorithm introduced by Emami et al. (Emami, 1996b), a solution is proposed to each of these three problems:

- An approach for choosing the optimum number of clusters is to make the fuzzy clusters compact and far from each other (Fukuyama, 1989). The idea of scatter matrices was chosen and generalized for fuzzy clustering. As a result, a validity index is introduced for the choice of the optimum number of clusters: or in other words, we minimize:

$$S_{\alpha}(U, V; X) = \sum_{k=1}^{N} \sum_{i=1}^{c} (u_{ik})^m \left( \|x_k - v_i\|^2 - \|v_i - \bar{v}\|^2 \right)$$

where $v_i$ is defined in Equation 1.8, and $\bar{v}$ is the fuzzy total mean vector of the data set considering their belonging to each of the clusters. It can be defined as:

$$\bar{v} = \frac{1}{\sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m} \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m x_k$$
For the selection of the weight exponent, it is suggested to choose it far from its both extremes so as to ensure that the cluster validity index shows the optimum number of fuzzy clusters. In Emami et al. (Emami, 1996b), a fuzzy total scatter matrix is defined as:

\[ s_T = \sum_{k=1}^{N} \left( \sum_{i=1}^{N} u_{ik} \right)^m \left( x_k - \bar{v} \right) (x_k - \bar{v})^T \tag{3.5} \]

The trace of \( s_T \) decreases monotonically from a constant value \( z \) to zero as \( m \) varies from one to infinity. For data partitioning, a suitable value for \( m \) is that which gives a value for \( \text{trace}(s_T) \) equal to \( z/2 \). The constant value \( z \) is defined as:

\[ z = \text{trace} \left[ \sum_{k=1}^{N} \left( x_k - \frac{1}{N} \sum_{k=1}^{N} x_k \right) \left( x_k - \frac{1}{N} \sum_{k=1}^{N} x_k \right)^T \right] \tag{3.6} \]

For the choice of the initial cluster centers, an agglomerative hierarchical clustering algorithm (AHC) is suggested as the initial clustering tool. The AHC algorithm puts each of the \( n \) data vectors in an individual cluster. Then, by defining a matrix of dissimilarities \( D = [d_{ij}] \), the AHC merges two or more of these clusters, getting to a higher level of data partition. The process is repeated to form a sequence of nested clustering in which the number of clusters decreases gradually until the minimum required number of clusters \( c \) is reached (Ward, 1963).

### 3.3 A Noise-Rejection Clustering Algorithm

In this section, we study the new Possibilistic C-Means algorithm (NPCM) introduced by Krishnapuram et al. (Krishnapuram, 1996). The NPCM is more robust than the traditional FCM in the presence of noise and outliers. The NPCM robustifies the PCM by forcing finite rejection of noise and outliers in the data set. The Possibilistic C-Means clustering algorithms use the objective function:

\[ J(U, V; X) = \sum_{i=1}^{c} \sum_{j=1}^{N} (u_{ij})^m d^2(x_j, v_i) + \sum_{i=1}^{c} u_i \sum_{j=1}^{N} (1 - u_{ij})^m \tag{3.7} \]

where \( d^2(x_j, v_i) \) is the distance from a feature point \( x_j \) to the cluster center \( v_i \), and \( u_i \) is a resolution parameter and it depends on the number of data partitions. The membership matrix \( U=[u_{ij}] \) is a global minimum for \( J(U, V; X) \) when:
\[
\text{\textit{u}}_{\textit{ij}} = \frac{1}{1 + \left\{ \frac{d^2(x_j, v_i)}{\text{var}} \right\}^{\frac{1}{\text{m-1}}}}. \tag{3.8}
\]

The new Possibilistic clustering algorithm defines a cutoff distance for finite rejection of noise in the data. For an ideal Gaussian cluster, the cutoff distance is chosen such that 97.5\% of the data points are considered as inliers. For an ideal Gaussian cluster with variance \( \text{var} \), \( \left\{ \frac{d^2(x_j, v_i)}{\text{var}} \right\} \) has a chi-square distribution (\( \chi^2 \)) with degrees of freedom equal to the dimension of each data feature (Myers, 1991). The resolution parameter is defined as:

\[
\text{var}_i = \frac{\text{median}(d^2(x_j, v_i))}{\chi^2_{0.5}} \tag{3.9}
\]

Then, the cutoff distance can be calculated as follows:

\[
d_{\text{cut}}^2 = \text{var}_i \chi^2_{0.975} \tag{3.10}
\]

Finally, the membership values can be computed using equation (3.8). If \( d^2(x_j, v_i) \) is bigger than the cutoff distance, then the point is identified as noise and it takes a membership grade \( u_{ij} = 0 \).

The reason for using \( \chi^2 \) distribution is based on the assumption that the clusters could follow a Gaussian distribution. For other types of distributions, the corresponding indices should be used.

The \textit{Possibilistic} clustering algorithms suffer from the same problems of the traditional FCM algorithm discussed in section 3.2. In order to avoid these problems, we will combine the techniques developed in (Emami, 1996b) with the NPCM algorithm presented in this section. Furthermore, by using the NPCM algorithm, the cutoff distance is calculated assuming that 97.5\% of the data points are inliers. However, this criterion may not be applicable for most of the real application. In other words, if we assume that the data in each cluster follow a Gaussian distribution, the percentage of inliers may not necessarily be 97.5\%. Hence, there is no scientific basis for the choice of the exact percentage of inliers in the data set. Thus, based on the assumption that the data in each
cluster follows a Gaussian distribution. we introduce a noise rejection criterion that merely depends on the data to be partitioned.

### 3.4 A Combined Algorithm with a Noise-Rejection Capability

In this section, we define a new algorithm that has a noise rejection capability as well as a defined criterion to assign the cutoff distance from the data. The steps of the algorithm can be stated as follows:

1) Using the fuzzy C-Means algorithm (Emami, 1996b), find:
   a) The initial cluster centers, using the agglomerative hierarchical algorithm (AHC).
   b) The suitable weighting exponent \( m \).
   c) The optimum number of clusters \( c \).

2) Calculate the \((c \times N)\) matrix of dissimilarities \( D = [d_{ij}] \) as the following Euclidean-based distance:

\[
d_{ij} = d(X_i, X_j) = \sqrt{\frac{2n_i n_j}{n_i + n_j} ||v_{hi} - v_{hj}||}
\]  

(3.11)

Where, \( v_{hi} \) and \( v_{hj} \) are mean vectors of the hard clusters \( X_i \) and \( X_j \), respectively, and, \( n_i (n_j) \) is the number of data in the hard cluster \( X_i \) and \( X_j \).

3) Calculate the sum of all the columns of the matrix of dissimilarities as \( \Omega = \text{sum}(D) \) and then plot \( \Omega \).

4) Cutoff the points that have large values of \( \Omega \) (to be explained in section 3.5).

5) Count the cutoff points. Calculate \( \lambda = \frac{\text{cutoff points}}{\text{total number of data}} \).

6) Calculate \( \lambda = 1 - \hat{\lambda} \). Using the Chi-square tables (Appendix 1). Find the exact value of \( \chi^2_\lambda \). The percentage of inliers in the data set is \( \lambda \% \). Calculate the cutoff distance using equation (3.10).

Steps 1-6 represent the proposed combined algorithm that has a noise rejection capability. In the next step, the improved partitions can be obtained by using the NPCM algorithm as follows:
7) Knowing the new cutoff distance, the optimum number of clusters, the suitable weight exponent, and the initial location of the cluster centers, apply the NPCM algorithm to reject the noise points and calculate the membership matrix \( U = [u_{ij}] \).

The next step would be the construction of the input-output membership functions (Chapter 4).

### 3.5 Examples

#### 3.5.1 Example 1

Fig. 3.1a shows the example in Krishnapuram et al. (Krishnapuram. 1996). We have two Gaussian clusters generated with centers at (50,50) and (150,150) respectively, with 80 data points in each cluster. 91 uniformly distributed noise points are added to the data set in the range of \( 20 \leq x_1 \leq 200, 20 \leq x_2 \leq 200 \). Fig. 3.1b shows the clustering results using the fuzzy C-means clustering algorithm. The suitable weight exponent is \( m = 3.5 \).

Figure 3.2a shows the optimum number of clusters obtained using the fuzzy C-Means clustering algorithm. Figure 3.2b shows the cut-off distance selection criterion. Some points have large values for \( \Omega \). These points basically represent the noise in the data as they are far from all cluster centers, and they are not enough to form a separate cluster.

![Figure 3.1: (a) Krishnapuram example; (b) The fuzzy C-Means clustering algorithm for example 3.5.1](image)
If we choose the cutoff distance at \( \Omega = 200 \), 22 points on Fig 3.2b are rejected. then:

\[
\hat{\lambda} = \frac{22}{2(80) + 91} = 0.07 ; \quad \lambda = 1 - \hat{\lambda} = 0.93
\]  (3.12)

The value of the chi-square distribution is obtained for \( \lambda = 0.93 \) (Appendix 1). In other words, the cutoff distance is calculated assuming that 93\% of the data are inliers.

Figure 3.3 shows a comparison between the partitions obtained using the NPCM and the new partitions using the improved clustering algorithm. For this example, the noise points are known before performing the comparison between both clustering algorithms. Based on this fact, we are able to evaluate the partitions obtained by each clustering algorithm. In the NPCM partitions, the cluster centers are identified correctly. On the other hand, some of the inliers are identified as noise. That is due to the fact that these points are far enough from the centers of the clusters and so the algorithm rejects them. The improved algorithm correctly identifies the clusters and rejects the noise. The accuracy of the results obtained from the improved algorithm relies on the correct choice of the cutoff distance in figure 3.2b.

**Figure 3.2:** (a) The optimum number of clusters; (b) The noise-rejection criterion for example 3.51
3.5.2. Example 2

In this example, we generated a two-dimensional data set (data set3) as shown in figure 3.4a. The data was created manually in MATLAB (5.0) such that it forms three clusters and some random noise points. We have 100 data features including the noise. Figure 3.4b shows the trace of the scatter matrix obtained from the fuzzy C-Means clustering algorithm. The suitable weight exponent for this example is $m = 2.5$ (Emami, 1996b). Figure 3.5a shows the validity index for this example. The optimum number of clusters is $c = 3$. At this stage, all the points are considered as inliers because the fuzzy C-Means clustering algorithm (Emami, 1996b) does not have a noise rejection capability. The exact value of $\chi^2$ (chi-square distribution) to be used for this example can be assigned using figure 3.5b. If we choose the cutoff distance at $\Omega=6$, 12 points are rejected. The cutoff distance to be used in the NPCM is calculated using $\chi^2_{0.975}$, instead of the $\chi^2_{0.9975}$ suggested by Krishnapuram et al. (Krishnapuram, 1996).
Figure 3.4: (a) data set 3; (b) Trace of the total scatter matrix vs. m for data set 3

Figure 3.6 shows the comparison between the partitions obtained using the NPCM without the noise-rejection criteria and the partitions using the combined algorithm. The idea is to identify the clusters correctly as it is recognized by inspection where, as mentioned earlier, the noise points are known before applying the clustering algorithms. The combined algorithm correctly identifies the cluster centers, rejects the noise points, and assigns the exact partitions in the data. Using the NPCM without the noise rejection criterion would identify some of the points that belongs to the clusters as noise.

Figure 3.5: (a) The optimum number of clusters; (b) The noise-rejection criterion for data set 3
3.5.3 Example 3

In this example, we generated another two-dimensional data set (data set A) with 150 data features including the noise. This example is designed to be more realistic having overlaps between the clusters. The data was also created manually in MATLAB (5.0) such that it forms four clusters and some random noise points. This example shows the capability of the combined algorithm to identify the exact data partitions in the presence of noise and outliers. Figure 3.7a shows the data set studied. Figure 3.7b shows the trace of the total scatter matrix for different values of the weighting exponent. The suitable value for the weight exponent in this example is $m = 2.5$. Figure 3.8a shows the validity index for this example. The optimum number of clusters is $c = 4$. Figure 3.8b shows the
noise rejection curve. If the cutoff distance is chosen such that $\Omega = 10$, then 18 points on the curve are rejected, and therefore $\lambda = 0.88$. The cutoff distance in the combined algorithm will be calculated using $\chi^2_{0.88}$.

Fig. 3.9 shows a comparison between the NPCM partitions and the partitions obtained using the improved algorithm. The improved algorithm identified some of the inliers that were rejected in the NPCM algorithm. Considering the clusters centers, both algorithms identify the cluster centers correctly in the presence of noise.

Figure 3.7: (a) Data set A; (b) The choice of the weight exponent for data set A
3.5.4 Example 4

Sugeno and Yasukawa (Sugeno, 1993) introduce the nonlinear static system with two input variables and a single output variable $y$ studied in Chapter 2. The system is as follows:

$$y = (1 + x_1^2 + x_2^{4.5})^2; \ 1 \leq x_1, x_2 \leq 5 \quad (3.13)$$

50 input-output data are randomly obtained. The output data points in fig 3.10a are to be clustered. Fig 3.10b shows the trace of scatter matrix using the fuzzy C-Means clustering algorithm (Emami, 1996b). The suitable weight exponent for this example is $m = 3$. Fig.3.11a shows the validity index with $c = 8$. Figure 3.11b is the noise-rejection curve. For this specific example, it is difficult to assign a cutoff distance. That is because...
the data is obtained from a *deterministic* function and so it is clean, or in other words, there is no noise in the data set. In an attempt to find the exact partitions, we will choose the cutoff distance such that $\Omega = 14.8$. Hence, 3 points are rejected. The cutoff distance for the combined algorithm will be calculated using $\chi^2_{0.94}$.

![Figure 3.10: (a) Sugeno-Yasukawa static function; (b) The trace of the scatter matrix vs. $m$ for Sugeno-Yasukawa example](image1)

![Figure 3.11: (a) The optimum number of clusters; (b) The noise-rejection example](image2)
3.6 Conclusions

In this chapter, we reviewed the fuzzy C-Means clustering algorithm. The original FCM clustering algorithms has some weaknesses in identifying the initial cluster centers, selection of the weight exponent, and the assignment of the optimum number of clusters. The FCM problems can be solved using the techniques developed by Emami et al. (Emami, 1996b). On the other hand, the fuzzy C-Means clustering algorithms may fail completely in the presence of noise and outliers. Krishnapuram et al. (Krishnapuram, 1996) introduce an improved Possibilistic C-Means algorithm for noise rejection. The improved Possibilistic clustering algorithm (NPCM) efficiently identifies the cluster centers in the presence of noise and outliers. However, the NPCM suffers from the same drawbacks of the original FCM clustering algorithms and introduces another parameter that must be identified a priori, i.e., the percentage of inliers in the data set. A new improved algorithm is introduced in this chapter. It is based on combining the improved Possibilistic algorithm with the techniques developed in the improved fuzzy C-Means clustering algorithm (Emami, 1996b). Unlike the NPCM, the new algorithm is based on a defined criterion for the assignment of the cutoff distance from the data. The matrix of dissimilarities is calculated for the data set and the noise rejection curve is considered. This methodology is applicable for n-dimensional data set because the noise rejection curve is always two-dimensional regardless of the dimension of the data features, i.e., only the two-dimensional data are observable. That is why the partitions obtained using the combined algorithm are more effective than those obtained by the NPCM.
CHAPTER

GENERALIZATION CAPABILITY OF FUZZY-LOGIC MODELS

4.1 Introduction

In this chapter, we investigate how the fuzzy-logic model can be extended from the training data set to the new testing data set that has not been used in the model construction. Further, we investigate how to maximize the model accuracy and at the same time reduce its complexity. The following issues are addressed: (i) in order to have a training data set that fully expresses the system behaviour, some conditions must be applied to the available data; (ii) to improve the generalization capability of the fuzzy-logic model, some conditions must be applied to the input-output membership functions; (iii) in order to choose the optimum number of tuning iterations, a suitable tuning technique should be defined for the training set; and (iv) some criteria should be available to evaluate the identified fuzzy model in terms of how accurately it performs for the new testing data sets.

In Section 4.2, some fundamentals of the data observation method are discussed, and conditions on the input-output data are introduced. In Section 4.3, the conditions on the input-output membership functions are presented, and a new algorithm is applied to the input-output membership functions during the tuning process of the training set of the model. As a result, we modify the input-output membership functions when the performance index of the training set drops below a specific value. Using the new algorithm, the model behaviour is improved when used for evaluating the outer limit testing data sets. In Section 4.4, different model evaluation methods are studied. A definition of the principle of parsimony is introduced. For the evaluation of the identified fuzzy model, the Akaike's information criterion and Occam's Razor are discussed. A comparison between the two criteria is presented based on the applicability of each of
them for evaluating the identified fuzzy model. In Section 4.5, the techniques developed in Section 4.3 are applied to the Sugeno-Yasukawa static function (Sugeno, 1993). Conclusions are presented in Section 4.6.

4.2 Conditions on the Data

A prerequisite step for any modeling and system identification approach is to perform a close and careful observation of the data sampled from the system to be modeled. We use sample data in developing and assessing the model not because of any special interest in the sample, but because we believe that ordinary samples provide valuable information about the system. In order to justify this hypothesis, certain precautions must be taken in collecting the data, and certain conditions must be satisfied by the sampled observations. In this section, we present the following questions and their answers, which should be fully addressed prior to the modeling process:

1) Is there any sign of dependence or contingency between the output and the input variables? A positive answer to this question helps assuring the validity of the model that is constructed based on the existing data, for the future data. Although our concern of system modeling is about the general nonlinear input-output relationship, the linear relation or trend should be first recognized. We should perform a trend analysis to know the type of the output dependence on each of the input variables (Emami, 1997b). The trend analysis reveals the type of the relationship between the output and each of the input variables. Linear or at least "monotonic" behaviour has a significant role in the interpretation of input-output relationship. In order to obtain more insight through the characteristic of the system, the crosscorrelation analysis would be of great help (Emami, 1997b). The correlation function shows the monotonic dependency of the output data point to the corresponding and pre and post input data points. A monotonic relationship is highly expected in real systems such as physiological systems (Emami, 1997b).

2) Do the input variables vary independently in the data set? The data set should illustrate no interrelationship between those variables that are intended to be inputs to the model. Otherwise, the constructed model may be limited to the correlated variation of the input variables, which is not desirable for a comprehensive model. A
simple test of the null hypothesis of the input dependence is the crosscorrelation analysis (Emami, 1997b).

3) In order to reduce the effect of high bandwidth noise on the signal that is practically unavoidable, what is the best cut-off frequency for each signal? A smoother signal is always preferred as it prevents "hyper-sensitivity" of the model, as long as no information is lost as a result of the filtering process.

4) An important question with regard to reducing the amount of data to a reasonable and applicable level is how much we can reduce the number of data in the signal such that the major information remains intact. For a "time series", the question is critical and directly depends on the filtering strategy. Reducing data has a crucial effect on the efficiency of the modeling procedure in terms of the computational load. The compromise is to provide the least possible data that contains the most information.

### 4.3 Construction of the Input-Output Membership Functions

After assigning the appropriate partitions of the output space, the next step is to form the membership functions of the entire output space. One method is to directly estimate the membership grades $u_{ij}$ derived from the fuzzy clustering algorithm by suitable trapezoids (Sugeno, 1993), and to use the approximate functions as a classification for the entire output space. After partitioning the entire space, in order to obtain simple membership functions, we can approximate the classified data by trapezoidal functions such that for each fuzzy cluster, convex points are picked up and a trapezoid is fitted to them (Nakanishi et al., 1993).

In order to identify the significant input variables among a finite number of candidates, we first project the output clusters onto the space of each of the input candidates. As a result, for each input candidate $x_i$, the membership functions $\hat{B}_{ij}(i = 1, 2, \ldots n)$ are formed. Then, we can define the following index:

$$\pi_j = \prod_{i=1}^{n} \frac{\Gamma_{ij}}{\Gamma_j}, \quad j = 1, 2, \ldots \tilde{r}$$

where, $\Gamma_{ij}$ is the range in which the membership function $\hat{B}_{ij}$ is equal to one, $\Gamma_j$ is the entire range of $x_j$, $n$ is the number of rules, and $\tilde{r}$ is the number of input candidates. Less
illustrates more dominant variable $x_j$, and hence, significant variables are selected among those that produce less $\pi$.

The convex membership functions $B_q$ for significant inputs $x_j (j=1,2,...,r)$ are then formed by using the range $\Gamma_q$ and performing "fuzzy line clustering" (Emami, 1996b).

After the construction of the input-output membership functions, one question needs to be answered: what constraints can we put on the membership functions in order to obtain a better performance from the model specifically when used for the outer limit testing data, i.e., data which is outside the range of the training set for both inputs and outputs.

Two situations may arise when the fuzzy-logic model is applied to a testing data set: In the first case, for a testing set, one or more of the inputs are beyond the range of all the input membership functions. In this case, the model will never be able to give a correct estimate of the output. This is due to the fact that a parametric $t$-norm is used for the antecedents aggregation as well as the implication relation in each rule (Emami, 1996a). If one of the arguments of the $t$-norm used for the antecedents aggregation is zero, the $t$-norm will be zero. Also, this $t$-norm is by itself an argument of another $t$-norm used for the implication relation in each rule, and then the final product of this relation will be zero. Hence, if one of the inputs does not fire any of the fuzzy rules, the output of the fuzzy model will be zero. Thus, the fuzzy model will never be able to give a correct estimate for the fuzzy and the crisp output.

In the second situation, for some testing sets, all the inputs may be inside the range of the input membership functions but the desired output is out of the range of the developed model. In this case, the model can not and is not expected to match the true output. That is because the initial training information is incomplete. The training set does not fully express the system behavior. A high sampling frequency should be chosen to catch up the higher bandwidth system behavior. In other words, we must include more data points in the training set such that it covers the entire range of the output variation.

To improve the generalization capability of the fuzzy model, the first possibility must be taken into consideration and the question is what kind of constraints we should apply to the membership functions to provide the potential generalization behavior.
In the fuzzy rule shown in Fig. 4.1a, the input \( x_i \) is more significant beyond \( x = c_i \). For \( c_i \leq x \leq c_j \), this input has no significant effect on the output due to the boundary condition (Smets, 1982) of the parametric \( t \)-norm used for antecedents aggregation (equation 1.5). For inputs beyond the range of the input membership function of the training set, i.e., \( x \leq c_i \), the output of the model will be zero because the result of the \( t \)-norm for antecedents aggregation is zero. This is because one or more of the inputs did not fire any of the rules (this input(s) has zero membership grades among all rules) as shown by the dashed line in Figure 4.1a. To avoid this problem, the input membership functions should be opened at the limits if \( c_i'' = c_i'' \) and/or \( c_i'' = c_i'' \) or if the distance between these adjacent membership values is smaller than the index in equation 4.2. The criterion is applied to each input membership function through the algorithm shown in Figure 4.2.

In the tuning process, we should first let the model be tuned for the training set. Hence, at the beginning, accuracy is more important than robustness. After some iterations when the model adequately reaches the training data, the constraints for robustness must be applied. The issue is when in the tuning process we should apply this algorithm. We will pick point \( a \) in Fig.4.3 for which the error in equation 1.14 drops to 20% of the initial model error to apply the new algorithm for the construction of the input membership functions. For this point, the error of the training set is not too small, and hence the change in the input membership functions using Algorithm1 would not cause a drastic increase in the error. However, by changing the input membership functions, the final error after the tuning is expected to be higher than the error of the tuned membership functions obtained through the robust inference mechanism introduced in Chapter 2. On the other hand, by using the new algorithm, the generalization capability of the fuzzy model will be improved. This will be illustrated through an example in Section 4.6.

The restriction after applying the new tuning method is that the points \( c_i'' \) equivalent to \( \bar{a}'' \) and \( \bar{a}'' \) should move together during the rest of the tuning process (after point (a)). However, the output membership function calculation should be different than the input. In other words, a point outside the output membership functions range should have a zero membership grade as discussed above.
**Algorithm 1**

(i) Set index $\omega^j$ for each input $x_j$ as follows:

$$\omega^j = 0.05\left(\max(c^uj_i) - \min(c^uj_i)\right) \quad (4.2)$$

(ii) Calculate $\omega^u_{ji} = c^uj_i - c^u_i$ for each input variable $x_j$ in each rule $i$ ($j = 1, 2, \ldots, r$, $i = 1, 2, \ldots, n$), where $n$ is the number of the rules, $r$ is the number of input variables, and $c_1$ and $c_2$ are shown in Figure 4.1.

(iii) For each input variable $x_j$, find the rule $i^*_j$ in which:

$$\tilde{\omega}^i_{2j} = \min_i(\omega^u_{2j}) \quad (4.3)$$

(iv) If $\tilde{\omega}^i_{2j} \leq \omega^j$, put $\tilde{c}^i_j = \tilde{c}^i_j$.

(v) Calculate $\omega^u_{4j} = c^u_{4j} - c^l_{4j}$ for each input variable $x_j$ in each rule $i$ where $c_{4j}$ and $c_{4j}$ are shown in Figure 4.1.

(vi) For each input variable $x_j$, find the rule $i^*_j$ in which:

$$\tilde{\omega}^i_{4j} = \min_i(\omega^u_{4j})$$

(vii) If $\tilde{\omega}^i_{4j} \leq \omega^j$, put $\tilde{c}^i_j = \tilde{c}^i_j$.

(viii) For each input membership function $B_{ij}$:

- if $c^l_{4j} = c^u_{4j}$, then $B_{ij}(x_j < c^l_{4j}) = 1$
- if $c^l_{4j} = c^u_{4j}$, then $B_{ij}(x_j > c^l_{4j}) = 1$

Fig 4.1: The proposed method for the input membership functions construction

Fig 4.2: An algorithm for the constraints on the input membership functions
4.4 The Tuning Process of the Model Training Set

In this section, we propose a new tuning process that contains the evaluation of different testing data sets during the tuning process of the model training set. These testing sets if available are not used in the model construction. Hence, by using the proposed tuning process, the optimum number of iterations can be chosen where the different testing data show minimum error. The procedure is that while the tuning is performed only with the training data set, after each iteration, the model error is calculated for different testing data sets. The tuned membership functions are selected at the iteration where a lower error is obtained from both the training and the testing data sets. In this way, we can avoid over-tuning the model parameters and hence avoid the risk of over-fitting. This will be illustrated through the example in Section 4.6.

Furthermore, in the systematic methodology of fuzzy-logic modeling, the optimum values of the inference parameters \( (p, q, \beta, \text{ and } \alpha) \) are identified through a nonlinear constrained optimization problem, by minimizing:

\[
PL (p, q, \beta, \alpha) = \sum_{k=1}^{N} (y_k - \hat{y}_k)^2 / N, \tag{4.4}
\]
subject to the following constraints:

\[ 0 < p, q < \infty \quad \& \quad 0 < a < \infty \quad \& \quad 0 < \beta < 1 \]  \quad (4.5)

where \( y_k \) is the actual output, \( \hat{y}_k \) is the model output, and \( N \) is the number of data.

If Algorithm 1 is applied to the input membership functions, the optimum values of the reasoning parameters identified by equation 4.4 will change. Thus, we recommend tuning the reasoning parameters during the tuning process of the model with the training set in order to identify the optimum reasoning parameters after applying the constraints to the input membership functions. During the tuning process, the reasoning parameters are free to vary between the lower and the upper bounds specified for each of them in Chapter 2 (Equations 2.69, and 2.71). Hence, we can identify the optimum reasoning parameters and guarantee the marginal continuity property at the same time.

### 4.5 Model Validation Criteria

In this section, we need to investigate the conditions required for extending the fuzzy-logic model from the training data set to the new testing data sets, which have not been used for the model construction. A robust fuzzy model should show an acceptable performance for different testing data sets as well as the training data set. When studying the generalization capability of fuzzy-logic models, one question arises: what is the most suitable method to evaluate an identified model? We adopt the Parsimony principle as a ground for evaluating the identified models. This principle indicates, roughly, that of two identifiable model structures that fit certain data, the simpler one (that is, the structure containing the smaller number of parameters) will on average give better accuracy (Soderstrom, 1989). The Parsimony principle has two major indices: The Akaike’s Information Criterion (Akaike, 1974) and the Occam’s Razor (Blumer, 1987). Both indices can be used as tools for evaluating an identified model or assigning the most generalized model among a group of competing models in terms of how satisfactorily they perform for the different testing data sets. One other question then arises: regarding the generalization capability of the fuzzy logic models, which of the two criteria is more applicable: Occam’s Razor or AIC? A detailed investigation is presented in this section to find the most applicable criterion for evaluating an identified fuzzy model as well as choosing the most generalized model among a group of competing models.
4.5.1 Description of the Prediction Error

The model obtained by identification can be used in many ways, depending on the purpose of modeling. In many applications the model is used for prediction. For each fuzzy model, we can define \( \theta \) as the vector of the model parameters. The model parameters include the reasoning parameters, and the membership function parameters. We define \( y_k \) as the actual output predicted by the model (with parameter vector \( \theta \)). For this case, the model output is \( \hat{y}_k(\theta) \), where \( k = 1, 2, \ldots, N \), and \( N \) is the number of data points. The prediction error in this case can be defined as the performance index in equation 1.14. Thus, we can write the prediction error as follows:

\[
\varepsilon(\theta) = \frac{1}{N} \sum_{k=1}^{N} (y_k - \hat{y}_k(\theta))^2
\]  

(4.6)

where \( \theta \) is the vector of the model parameters, i.e., reasoning parameters and the membership function parameters, and \( N \) is the number of data points.

For a group of competing fuzzy models, the prediction error in equation 4.6 is to be minimized with respect to \( \theta \) to choose the optimum predictor (model) in the considered class. In other words, under the given model assumptions, the prediction error should have as small variance as possible.

4.5.2 The Parsimony Principle

The Parsimony principle is a useful rule when determining an appropriate model order. This principle says that out of two or more competing models, which all explain the data well, the model with the smallest number of independent parameters should be chosen (Soderstrom, 1989).

Let us consider a group of competing models and assume that we know the model that has the lowest prediction error, i.e., the original model. For this model, there exists a true parameter vector \( \theta_0 \) that belongs to the original model structure. If we consider one other model among the group of the available models, \( \hat{\theta}_N \) will represent the parameter vector for this model. A scalar measure will be used to test the goodness of the model associated with \( \hat{\theta}_N \). Such a measure should be a function of \( \hat{\theta}_N \) and will be denoted by
$W(\hat{\theta}_N)$ where the dependence on the number of data is emphasized (Soderstrom. 1989). The measure $W(\theta)$ must be a smooth function of $\theta$ and be minimized by the true parameter vector $\theta_0$. Where, for all values of $\theta$:

$$W(\theta) \geq W(\theta_0). \tag{4.7}$$

When the estimate $\hat{\theta}_N$ deviates a little from $\theta_0$, the criterion will increase somewhat above its minimum value $W(\theta_0)$. This increase $W(\hat{\theta}_N) - W(\theta_0)$ will be taken as a scalar measure of the performance of the model.

Let $w_N$ be the prediction error variance when the model corresponding to $\hat{\theta}_N$ is used to predict the testing data, which has not been used in the model construction. This means that:

$$W_N = E(e(\hat{\theta}_N)) \tag{4.8}$$

where $E$ is the expectation operator and $e$ is the prediction error given in equation 4.6. The expectation is conditional with respect to the testing data. If the estimate $\hat{\theta}_N$ were exact, i.e., $\hat{\theta}_N = \theta_0$, then the prediction error would be white noise (Soderstrom. 1989) and have a minimum variance $\Lambda$. It is proved that (Box, 1970):

$$E(W_N) = \Lambda \left(1 + \frac{\nabla}{N}\right) \tag{4.9}$$

where $N$ is the number of data points, $\nabla = \dim(\theta)$, and $E(W_N)$ represents the expectation of the prediction error variance.

This expression is remarkable in its simplicity. It says that the expected prediction error variance increases with a relative amount of $\nabla/N$. Thus, as the number of parameters of the model increases, $\nabla$ will increase and the prediction error variance in equation 4.9 will also increase. Hence, there is a penalty in using models with unnecessarily many parameters. In other words, the model with the smallest number of parameters will have the minimum variance. This can be seen as a formal statement of the Parsimony principle.
4.5.3 Akaike’s Information Criterion (AIC)

The Akaike’s information criterion is one of the indices of the parsimony principle. This criterion is used for the assessment of the model structure under study. Such a criterion for example may be obtained by penalizing in some way the decrease of the loss function \( V_N(\hat{\theta}_N) \) with increasing the model parameters. The model structure giving the smallest value of this criterion is selected. A general form of this criterion is as follows (Soderstrom, 1989):

\[
W_N = V_N(\hat{\theta}_N)\{1 + \partial(N, \nabla)\} \tag{4.10}
\]

The parameter estimate \( \hat{\theta}_N \) is obtained as the minimizing element of the loss function \( V_N(\theta) \). If we choose:

\[
\partial(N, \nabla) = \frac{2\nabla}{N - \nabla}, \tag{4.10a}
\]

we get the so-called Final Prediction Error (FPE) criterion (Akaike, 1974). The term \( \partial(N, \nabla) \) is a function of the number of data \( N \) and number of parameters of the model and should increase with \( \nabla \) in order to penalize "too complex" model structures in view of the parsimony principle. The function \( \partial(N, \nabla) \) must tend to zero as \( N \to \infty \) to guarantee that the penalizing term will not obscure the decrease of the loss function \( V_N(\hat{\theta}_N) \) with increasing under parameterized model structure. An alternative form of equation 4.10 is:

\[
W_N = N \log V_N(\hat{\theta}_N) + \Phi(N, \nabla) \tag{4.11}
\]

where \( \Phi(N, \nabla) \) should penalize high-order models. The choice \( \Phi(N, \nabla) = \frac{2\nabla N}{N - \nabla} \) with the assumption that \( N \) is large leads to the widely used Akaike’s Information Criterion (AIC), which represents a special case of the Final Prediction Error criterion. The loss function in equation 4.11 can be chosen as follows:

\[
V_N(\hat{\theta}_N) = \sum_{k=1}^{N} \left( y_k - \hat{y}_k(\hat{\theta}_N) \right)^2 \tag{4.12}
\]

where \( \hat{\theta}_N \) is the vector of the model parameters, i.e., reasoning parameters and the membership function parameters, and \( N \) is the number of data points.
The formulation in equation 4.11 has two terms. The first term measures the prediction error and the second term accounts for the number of the parameters of the model. Let us assume that we have two different models: the first model has a large number of parameters and fits the data “very” well. The second model has a much smaller number of parameters and fits the data “fairly” well. The AIC may fail to give a consistent estimate of the most generalized model in the two above models. The second model will give a small value for the second term in equation 4.11. However, for the first term in equation 4.11, the second model will give a higher value than the first model. Hence, the AIC may choose the first model as the most generalized model as it minimizes its index. This can be viewed as a drawback of the AIC, where the criterion fails when overfitting occurs.

4.5.4 Occam’s Razor

Occam’s Razor criterion is another index of the parsimony principle (Hidetomo, 1993). The criterion simply says that: given two models for the data, all other things are equal, i.e., the same order, the same number of parameters, etc., the simpler model is preferable. In the following section, we introduce the Bayesian probability theory (MacKay, 1995) that embodies the Occam’s Razor criterion. A Bayesian model comparison is performed to give a better understanding of the Occam’s Razor.

4.5.4.1 Bayesian Model Comparison and Occam’s Razor

Using the Bayesian theorem (MacKay, 1995), we evaluate two alternative theories \( \Psi_1 \) and \( \Psi_2 \) in the light of the data \( N \) as follows: we relate the plausibility of model \( \Psi_1 \) given the data, \( P(\Psi_1|N) \) (a measure of the plausibility of proposition \( \Psi_1 \), assuming that the information in data \( N \) is true), to the predictions made by the model about the data, \( P(N|\Psi_1) \), and the prior plausibility of \( \Psi_1 \), \( P(\Psi_1) \). This gives the following probability ratio between theory \( \Psi_1 \) and theory \( \Psi_2 \):

\[
\frac{P(\Psi_1|N)}{P(\Psi_2|N)} = \frac{P(\Psi_1)P(N|\Psi_1)}{P(\Psi_2)P(N|\Psi_2)} \quad (4.13)
\]
The first ratio \( \frac{P(\Psi_1)}{P(\Psi_2)} \) on the right-hand side measures how much our initial beliefs favored \( \Psi_1 \) over \( \Psi_2 \). The second ratio expresses how well the observed data were predicted by \( \Psi_1 \) compared to \( \Psi_2 \).

![Evidence](image)

**Figure 4.4: Occam Razor and models comparison**

Figure 4.4 gives the basic intuition for why complex models can turn out to be less probable. The horizontal axis represents the space of the possible data sets \( N \). Bayesian theorem rewards models in proportion to how much they *predicted* the data that occurred. These predictions are quantified by a normalized probability distribution on \( N \). In this work, this probability of the data given model \( \Psi_i \), \( P(N|\Psi_i) \), is called the evidence of \( \Psi_i \).

A simple model \( \Psi_1 \) makes only a limited range of predictions, shown by \( P(N|\Psi_1) \); a more powerful model \( \Psi_2 \) that has, for example, more free parameters than \( \Psi_1 \) is able to predict a greater variety of data sets. This means, however, that \( \Psi_2 \) do not predict the data sets in region \( c_1 \) as strongly as \( \Psi_1 \). Suppose that equal prior probabilities have been assigned to the two models. Then, if the data set falls in region \( c_1 \), the less *powerful* model \( \Psi_1 \) will be the more *probable* model.

Model comparison is a difficult task because it is not possible simply to choose the model that fits the data best: more complex models can always fit the data better, so the maximum likelihood model choice would lead us inevitably to implausible, over-parameterized models which generalize poorly. Occam's Razor is needed. Each model \( \Psi_i \) is assumed to have a vector parameter \( \theta \). A model is defined by a collection of probability distributions: a "prior" distribution \( P(\theta|\Psi_i) \) which states what values the model
parameters might be expected to take; and a set of conditional distributions, one for each value of $\theta$, defining the predictions $P(N|\theta, \Psi_i)$ that the model makes about the data $N$. The posterior probability of each model is:

$$P(\Psi_i|N) = P(N|\Psi_i)P(\Psi_i)$$

(4.14)

where $P(N|\Psi_i)$ is the evidence for $\Psi_i$. The second term, $P(\Psi_i)$ is the subjective prior over our hypothesis space which expresses how plausible we thought the alternative models were before the data arrived. Assuming that we choose to assign equal priors $P(\Psi_i)$ to the alternative models, models $\Psi_i$ are ranked by evaluating the evidence.

To reiterate the key concept, to rank alternative models $\Psi_i$, a Bayesian evaluates the evidence $P(N|\Psi_i)$. This concept is very general: the evidence can be evaluated for parametric and non-parametric models alike; whatever our data modeling task, a regression problem, a classification problem, or a density estimation problem, the evidence is a transportable quantity for comparing alternative models. In all these cases, the evidence naturally embodies Occam’s Razor.

Finally, a major disadvantage of Occam’s Razor method is that measuring $P(N|\Psi_i)$ is very difficult if the distribution of the model parameters is not known.

### 4.5.5 The Relation between AIC and Occam’s Razor and their applicability to Fuzzy-Logic Modeling

In this section, a famous method of generalization called “leave one out” cross validation (Hidetomo, 1993), is adopted to give a better understanding of how the AIC can be used to evaluate the identified fuzzy model structures. For $N$ data, a fuzzy model is generated using $N-1$ points, leaving only one point out to test the developed model. The procedure is repeated $N$ times, each time leaving a different point out for testing. If the number of data points is large, this method may be computationally expensive. The $K$-fold cross validation methodology suggests that the data be divided into $K$ mutually exclusive data sets of equal size. $K$ fuzzy models will be developed; each time leaving one set out. The AIC is used to choose the best model among the total number of models developed. The model with the minimum AIC generalizes better. For different competing fuzzy models, if the prediction errors of all models are comparable, the model with the
minimum number of clusters is the model with the smallest number of parameters. Hence, the fuzzy model with the minimum number of rules (i.e., Clusters) is the most general model according to the Akaike’s Information Criterion (AIC).

The AIC criterion is an applicable index for evaluating fuzzy models. However, the risk of over-fitting the models should be avoided due to the drawback of the AIC under this specific condition. On the other hand, it is not possible to simply apply Occam’s Razor for fuzzy models since we do not know the distribution of the model parameters and no assumptions are valid.

4.6 Example

In this section, we apply the techniques developed in this chapter (reconstruction of the input membership functions and the generalization analysis) to the Sugeno-Yasukawa example introduced in Chapter 2. In order to guarantee the marginal continuity and improve the generalization capability of the model at the same time, we combine the methodology developed in Chapter 2 with the techniques introduced in this Chapter. As mentioned in Section 2.7, in order to guarantee the marginal continuity, the model parameters are chosen as:

$$\gamma_c = 0.3750 : r^l = 0.0897 : \gamma_F = 0.0336$$  \hspace{1cm} (4.15)

where $\gamma_c$ is the modulus of continuity of the crisp output, $r^l$ is the mutual ratio of the data, and $\gamma_F$ is the modulus of continuity of the fuzzy output.

The chosen value for input membership function deviation is $\delta_F = 0.015$, for which the lower bounds of the reasoning parameters are as follows:

$$p_{min} = 2.9 : q_{min} = 3.7$$  \hspace{1cm} (4.16)

Using the assigned $\gamma_c$ and the condition in equation 2.73, the marginal continuity of the model crisp output is achieved when the range of the parameter $\alpha$ is $[0.89 \rightarrow \infty)$. After assigning the lower bound of parameters $p$ and $q$ and the range of parameter $\alpha$, the optimum inference parameters are identified. The optimum values are:

$$p = 19.549 : q = 13.42 : \beta = 0.0001 : \alpha = 1.0518.$$  \hspace{1cm} (4.17)

By optimizing the inference parameters, the fuzzy model performance index (equation 1.14) is $PI = 0.2608$. The second step of parameter identification is to adjust the
input-output membership parameters. Several testing sets are generated using the same function in equation 2.78. Some of these sets have outer limit characteristics, i.e., some of the input points are outside the range of the input for the training set. The performance index of these sets is evaluated during the tuning process of the model training set. Figure 4.5 shows the tuning process of the model training set. The idea is to choose the most optimum iteration for which several testing data show minimum error.

The new algorithm introduced in this section is applied to the input membership functions (during the tuning process) when the performance index of the training set drops to 20% of its initial value. After 15 iterations, iteration \( k = 12 \) is chosen and the performance index of the training set is \( \Pi = 0.0137 \). Furthermore, the reasoning parameters are tuned as well to minimize the model performance index. Figure 4.6 shows the variation of the reasoning parameters during the tuning process of the model training set. The final values of the tuned reasoning parameters are as follows:

\[
\begin{align*}
\rho_{\text{tuned}} &= 24.4360 \quad q_{\text{tuned}} = 14.0924 \quad \beta_{\text{tuned}} = 0 \quad \alpha_{\text{tuned}} = 1.1175 \\
\end{align*}
\]  \tag{4.18}

Figure 4.7 shows a comparison between the developed fuzzy model for the training set and the function. Figure 4.8 shows the input-output fuzzy rules of the model.

We perform a comparison study between the model developed in this section and the original model developed by (Emami et al., 1996a) for the different testing data sets available. Table 4.1 shows a table of the final results of the comparison introduced. The performance index of both models for the training set as well as the model testing sets is included. Figures 4.9 and 4.10 shows a comparison between the two models for two outer limit testing data sets. Figures 4.11, 4.12, and 4.13 shows the same comparison for some of the testing sets used in Chapter 2.
CHAPTER 4
GENERALIZATION CAPABILITY OF FUZZY-LOGIC MODELS

<table>
<thead>
<tr>
<th>Data set (Training set &amp; Testing sets)</th>
<th>Robust Model Performance Index</th>
<th>Original Model Performance Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Training set</td>
<td>0.0137</td>
<td>0.004</td>
</tr>
<tr>
<td>Testing Set 1 (Outer limit characteristics)</td>
<td>0.0455</td>
<td>1.0129</td>
</tr>
<tr>
<td>Testing Set 2 (Outer limit characteristics)</td>
<td>0.0312</td>
<td>1.2595</td>
</tr>
<tr>
<td>Testing Set 3 (Inner limit characteristics)</td>
<td>0.0141</td>
<td>0.0393</td>
</tr>
<tr>
<td>Testing Set 4 (Inner limit characteristics)</td>
<td>0.0150</td>
<td>0.0155</td>
</tr>
<tr>
<td>Testing Set 5 (Inner limit characteristics)</td>
<td>0.0157</td>
<td>0.0309</td>
</tr>
</tbody>
</table>

Table 4.1: The comparison between the performance index of the robust model and the original model

From the results in Table 4.1, we can see that for the testing set 1 (outer limit characteristics), the robust model outperformed the original model (Emami, 1996a) showing 183% improvement in the performance. For testing set 2 that has the same nature as testing set 1, the robust model also shows a much lower error with 185% improvement in the performance. The testing sets used have 40% of the data outside the input range of the training data set. Thus, for the original model, none of these points fire any of the fuzzy rules in Figure 2.4. Hence, the model will never be able to give a correct estimate of the output as shown in Figures 4.9 and 4.10. On the other hand, using the Algorithm 1 introduced in this Chapter, the robust model is able to give a much better estimate of these testing data that has an outer limit characteristics. For the testing sets 3, 4, and 5, the robust model also shows an error that is less than that of the original model with a 55% improvement in average. For these testing sets, we do not have any outer limit data. That is why the performance index of both models is close. However, the robust model shows a better error as mentioned earlier, due to the proposed techniques of tuning the training set without over fitting. As a conclusion, the results obtained in this
Chapter 4
GENERALIZATION CAPABILITY OF FUZZY-LOGIC MODELS

Chapter show the significance of the proposed techniques to improve the generalization capability of the fuzzy-logic models.

Figure 4.5: The tuning process of the model training set

Figure 4.6: (a) Variation of the parameter p: (b) Variation of the parameter q
CHAPTER 4
GENERALIZATION CAPABILITY OF FUZZY-LOGIC MODELS

Figure 4.6 (contd.): (c) Variation of the parameter beta; (d) Variation of the parameter alpha

Figure 4.7: A comparison between the function and the developed robust model for the training set
Fig 4.8: The input-output rules for the developed robust model
Fig 4.9: A comparison between the robust model and the original model for the testing set (outer limit characteristic data)
Fig 4.10: A comparison between the robust model and the original model for the testing set 2 (outer limit characteristic data)
Fig 4.11: A comparison between the robust model and the original model for the testing set 3.
Fig 4.12: A comparison between the robust model and the original model for the testing set 4
Fig 4.13: A comparison between the robust model and the original model for the testing set 5.
4.7 Conclusions

In this Chapter, the generalization capability of the fuzzy-logic model was investigated. In order to improve this capability, several recommendations and conditions on the sampled data are introduced. Data observation methods are studied and some conditions on the input-output membership functions are presented. A new tuning process was introduced. The proposed tuning process is based on evaluating several testing data sets during the tuning process of the model training set. Finally, two different evaluation criteria were introduced to evaluate the developed fuzzy model. The techniques developed were illustrated through the example introduced in Chapter 2 (Sugeno, 1993).

A comparison between the original model (Emami, 1996a) and the developed robust model was performed. For testing sets with outer limit characteristics, the robust model outperformed the original model showing a 185% improvement in the performance. Furthermore, for the testing sets with inner limit characteristics, using the robust model, the average percentage improvement is 55%. Hence, we conclude from these results that the proposed techniques in this Chapter show a significant improvement in the generalization capability of the fuzzy model.

In Chapter 5, the proposed techniques to improve the generalization capability of the fuzzy model are applied to a 4 degree-of-freedom reconfigurable robot (Hui et al., 1993). The significance of the developed techniques is also illustrated through the gas furnace plant example addressed in Box et al. (Box, 1970).
5.1 Introduction

In this Chapter, we introduce two real applications. The first application is a gas furnace plant introduced in Box and Jenkins (Box, 1970). The second application is the IRIS robot arm (Hui et al., 1993). We build a robust model for each of the two applications and we perform a comparison study between the developed model and the original model in (Emami, 1996a).

5.2 A Gas Furnace Plant Application

In this Section, we apply the techniques developed in this work to a famous example (Box, 1970). The process is a gas furnace plant with a single input $u(t)$ (gas flow rate) and a single output $y(t)$ (CO$_2$ concentration). For this dynamic system, 10 input candidates $y(t-1), ..., y(t-4), u(t-1), ..., u(t-6)$ are considered. We use 100 data points as the training set. Figure 5.1 and 5.2 show the indices $tr(S_T)$ and $s_{cs}$ as functions of $m$ and $c$ respectively. For this application, $m=2.5$ and $c = 6$. The significant input variables are determined as $y(t-1), u(t-2), \text{and } u(t-3)$. Using the original modeling methodology (Emami, 1996a), the optimum inference parameters for the gas furnace system are derived as:

$$p_{opt} = 7.1463; \quad q_{opt} = 24.67; \quad \beta_{opt} = 0.3724; \quad \alpha_{opt} = 20.2 \quad (5.1)$$

After 10 iterations, the performance index reduces to a value of $PI = 0.0630$. For this application, two testing sets were used to evaluate the model behaviour for inputs that have not been used in the system identification problem.
For the same training set, we apply the techniques developed in this work. Figure 5.3 shows the criterion for the selection of the lower bounds of the parameters $p$ and $q$. Figure 5.4 shows the bounds of the parameter $\alpha$. The model parameters are as follows:

\[ t^i = 0.0148 \quad \chi = 24.67 \quad \zeta = 0.3724 \quad h_F = 20.2 \quad (5.2) \]

The required bounds of the reasoning parameters to achieve the marginal continuity are as follows (we use the subscript $\text{min}$ to denote the lower bounds of $p$ and $q$):

\[ p_{\text{min}} = 2.5 \quad q_{\text{min}} = 3.2 \quad \beta = [0 \rightarrow 1] \quad \alpha = [0.5 \rightarrow \infty] \quad (5.3) \]

where the optimum inference parameters for the gas furnace system are identified as:

\[ p_{\text{opt}} = 7.1463 \quad q_{\text{opt}} = 24.67 \quad \beta_{\text{opt}} = 0.3724 \quad \alpha_{\text{opt}} = 20.2 \quad (5.4) \]

Figure 5.5 shows the tuning process of the model training set. The optimum iteration is identified as $k = 6$ for which the performance index reduces to $PI = 0.0631$. It may be obvious from the figure that iteration $k = 4$ has a smaller error for both testing sets. However, the Algorithm 1 introduced in Chapter 4 is applied after iteration 4 to reconstruct the input membership functions. This can be concluded due to the increase of the error of the training set after iteration 4. Moreover, the testing sets evaluated have inner limit characteristics. Hence, to guarantee a better generalization capability from the model for other testing sets (outer limit characteristics), we choose iteration $k = 6$ instead of iteration $k = 4$. Figure 5.6 shows the variation of the reasoning parameters during the tuning process of the model training set. As we can see, the parameter $p$ is decreasing in the early iterations and then becomes almost constant after 7 iterations. On the other hand, the parameter $q$ keeps on increasing from one iteration to the other. However, no significant change in the values of both parameters is experienced. For the parameter $\beta$, a non-significant increase is experienced after iteration 4 and then it gets almost constant after applying Algorithm 1 at iteration 6. Finally, for the parameter $\alpha$, a non-significant increase can be seen in the first four iterations. Again, no significant change in the values of these parameters is experienced during the iterations. The tuned values of the reasoning parameters for the chosen iteration are as follows (we use the subscript $\text{tuned}$ to denote for the tuned reasoning parameters):

\[ p_{\text{tuned}} = 6.1637 \quad q_{\text{tuned}} = 30.837 \quad \beta_{\text{tuned}} = 0.3812 \quad \alpha_{\text{tuned}} = 21.20 (5.5) \]

Figure 5.7 shows the fuzzy rules of the developed robust fuzzy model. Finally, a comparison between the robust model and the original model is presented in Figures 5.8.
5.9. and 5.10. For the training set, both models show a perfect match to the system. For the testing set 1, the robust model shows a smaller error with a 10% improvement in the performance. For testing set 2, both models show a higher error. The original model failed to predict few points in this testing set. On the other hand, the robust model shows a better estimation for some of these points and hence gives a lower error. Table 5.1 shows the performance index of both models for the training set and the testing sets.

<table>
<thead>
<tr>
<th>Data set (Training set &amp; Testing sets)</th>
<th>Robust Model Performance Index</th>
<th>Original Model Performance Index</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Training set</td>
<td>0.0631</td>
<td>0.0630</td>
<td>-1.58%</td>
</tr>
<tr>
<td>Testing Set 1</td>
<td>0.5931</td>
<td>0.6595</td>
<td>10.6%</td>
</tr>
<tr>
<td>Testing Set 2</td>
<td>1.5585</td>
<td>1.6286</td>
<td>4.39%</td>
</tr>
</tbody>
</table>

Table 5.1: Results of the comparison between the original model and the robust model

From the results in Table 5.1, we conclude that the robust model has a smaller error for both of the testing sets used, while the error for the training set remains almost the same for both approaches. The robust model shows a performance improvement of 7% in average. The original model is also rather robust. Note also that no outer limit data is used in the testing sets.
Fig 5.1: Selection of the weight exponent for the gas furnace example

Fig 5.2: Identification of the optimum number of clusters for the gas furnace example
Fig 5.3: Selection of the lower bounds for the parameters $p$ and $q$

Fig 5.4: The range of the parameter $\alpha$ for the gas furnace example
CHAPTER 5
APPLICATIONS

Fig 5.5: The tuning process of the training set of the gas furnace model

Fig 5.6: Variation of the reasoning parameters during the tuning process of the model training set
Fig 5.7: The fuzzy input–output rules of the robust fuzzy model for the gas furnace example
Fig 5.8: Comparison between the original model and the robust model for the training set
Fig 5.9: Comparison between the original model and the robust model for the testing set 1

Fig 5.10: Comparison between the original model and the robust model for the testing set 2
5.3 Application to Robot Manipulators

This application was introduced in Emami (Emami, 1996a). The application is the IRIS robot arm (RoboTwin) developed in the Robotics and Automation Laboratory at the University of Toronto (Hui et al., 1993). This facility is a reconfigurable and versatile test-bed composed of two 4 d.o.f robot manipulators that can easily be disassembled and reassembled to provide a large variety of configurations. Data collection for this application was performed. Several trajectories for each joint are used to collect data sets used for modeling. The significant input candidates were identified for each joint. The 4 d.o.f IRIS arm has 12 input candidates, i.e., joints displacements ($q_1$, $q_2$, $q_3$, and $q_4$), velocities ($q_{d1}$, $q_{d2}$, $q_{d3}$, and $q_{d4}$), and accelerations ($q_{dd1}$, $q_{dd2}$, $q_{dd3}$, and $q_{dd4}$), and 4 output functions as the joint torques ($tau_1$, $tau_2$, $tau_3$, and $tau_4$). In this work, we choose some data from the random trajectory for each joint and the fuzzy logic model of each joint is built up separately by considering the effect of other joints dynamics. The inputs and the output for each joint are shown in Table 5.2. For joint 1, 200 data points from the random trajectory used (Emami, 1996a) are chosen as the training set. Two testing sets are chosen with 33 and 34 points respectively. For joint 2, the training set has 216 random points and two testing sets are chosen with 41 and 47 points respectively. For joint 3, the training set has 170 points and the two testing sets have 40 and 57 points respectively. Finally, for joint 4, the training set chosen from the random trajectory has 180 points and the two testing sets have 40 and 47 points respectively. After data preparation, the output of each joint, i.e., joint torque, is clustered. Figure 5.11 shows the clustering results for the output data of each joint. The index $s_{vs}$ (equation 3.3) for selecting the optimum number of clusters for the selected weight exponent $m$ is also shown in figure 5.11. For joint 1, 7 clusters are obtained, which is different from the number of partitions obtained in (Emami, 1996a), i.e., $c=4$. For joint 4, the optimum number of partitions is identified as $c=2$, which is also different from the number of partitions obtained in Emami, 1997), i.e., $c=6$. In this application, only a part of the random trajectory of each joint is used and that is why we get different partitions for some joints. The optimum values of the inference parameters are obtained by minimizing the performance index in equation 1.14. Table 5.3 shows the optimum weight exponent $m$, the optimum number of clusters $c$ identified for each joint, and the optimum values of
the reasoning parameters. After partitioning the output data for each joint, we develop the original fuzzy model (Emami, 1996a) and the robust model for each joint separately. For the robust model, we use the techniques developed in this work to guarantee the marginal continuity of the fuzzy output and to improve the generalization capability of the developed fuzzy model. Figure 5.12 shows the criterion for the selection of the lower bounds of the parameters \( p \) and \( q \). As a reminder, we choose the lower bounds of the parameters \( p \) and \( q \) such that the product \( pxq \) is minimum. In other words, we need to give a wider range of variation for both parameters during the tuning process. Figure 5.13 shows the criteria for the selection of the range of the parameter \( \alpha \) for each joint. Table 5.4 lists the bounds of the reasoning parameters and the modulus of continuity of the fuzzy output and the crisp output for each joint (we use the subscript \textit{min} to denote for the lower bounds of \( p \) and \( q \) and \textit{range} to denote for the range of \( \alpha \)). Figure 5.14 shows the tuning process of the training data set for each joint separately. The iteration stop point in each case is chosen where the training data as well as the testing data show satisfactory error. For each joint, the variations of the reasoning parameters during the tuning process of the model training set are shown in Figure 5.15. Table 5.5 shows the tuned reasoning parameters identified during the tuning process for each training set (we use the subscript \textit{tuned} to denote for the tuned parameters). Figures 5.16 – 5.19 show the final fuzzy rules for each joint. Table 5.6 shows the final results of the comparison between the original model and the robust model for each of the IRIS robot joints. Results are shown in Figures 5.20-5.31 and the final conclusions are presented below Table 5.6.

<table>
<thead>
<tr>
<th>Input Variables</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOINT # 1 qd1,qdd1,qdd2,qdd3,qdd4</td>
<td>tau1</td>
</tr>
<tr>
<td>JOINT # 2 qd2,qdd2,qdd1,qdd3,qdd4</td>
<td>tau2</td>
</tr>
<tr>
<td>JOINT # 3 q1,q3,qd3,qdd3,qdd1,qdd2,qdd4</td>
<td>tau3</td>
</tr>
<tr>
<td>JOINT # 4 qd4,qdd2,qdd4</td>
<td>tau4</td>
</tr>
</tbody>
</table>

Table 5.2: The inputs and the outputs for each joint
For each joint, the lower bounds of the parameters $p$ and $q$ are identified as shown in Fig 5.12. The lower bounds are chosen in a way to give a wider range of variation for these parameters during the tuning process of the robust model. In other words, the lower bounds are chosen such that both $p$ and $q$ are as small as possible. For joint 1, the maximum allowed deviation of the input membership grades is chosen as 0.004, which is about 40% of the modulus of continuity of the fuzzy output. The range of $\alpha$ is identified such that the maximum deviation of the crisp output never exceeds $\gamma_c$. The same criterion for the selection of the lower bounds of $p$ and $q$ and the range of $\alpha$ is applied to the other joints. For joint 2, the maximum allowable deviation of the input membership grades is chosen as 0.005, which is 25% of the modulus of continuity of the fuzzy output for this joint. For joint 3, the maximum deviation is 0.015, which is about 40% of the modulus of continuity. Finally, for joint 4, the maximum deviation of the input membership grades is
0.01 which is also 40% of the value of the corresponding modulus of continuity for this joint.

<table>
<thead>
<tr>
<th>JOINT # 1</th>
<th>$p_{tuned}$</th>
<th>$q_{tuned}$</th>
<th>$\beta_{tuned}$</th>
<th>$\alpha_{tuned}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOINT # 1</td>
<td>4</td>
<td>4.56</td>
<td>0.0001375</td>
<td>2.6936</td>
</tr>
<tr>
<td>JOINT # 2</td>
<td>35</td>
<td>2</td>
<td>0.00035</td>
<td>7.1479</td>
</tr>
<tr>
<td>JOINT # 3</td>
<td>148.5</td>
<td>4.53</td>
<td>0.098</td>
<td>4.7342</td>
</tr>
<tr>
<td>JOINT # 4</td>
<td>2.3</td>
<td>2.9750</td>
<td>0.00025</td>
<td>8.0893</td>
</tr>
</tbody>
</table>

Table 5.5: Identifying the reasoning parameters during the tuning process

<table>
<thead>
<tr>
<th>JOINT # 1</th>
<th>Original Model</th>
<th>Robust Model</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Set</td>
<td>0.4881</td>
<td>0.6583</td>
<td>-30%</td>
</tr>
<tr>
<td>Testing Set 1</td>
<td>0.8513</td>
<td>0.6624</td>
<td>25%</td>
</tr>
<tr>
<td>Testing Set 2</td>
<td>1.4517</td>
<td>1.1739</td>
<td>21%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JOINT # 2</th>
<th>Original Model</th>
<th>Robust Model</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Set</td>
<td>0.1501</td>
<td>0.1755</td>
<td>-15.6%</td>
</tr>
<tr>
<td>Testing Set 3</td>
<td>0.4200</td>
<td>0.4230</td>
<td>-0.71%</td>
</tr>
<tr>
<td>Testing Set 4</td>
<td>1.0261</td>
<td>0.2891</td>
<td>112%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JOINT # 3</th>
<th>Original Model</th>
<th>Robust Model</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Set</td>
<td>0.0652</td>
<td>0.0729</td>
<td>-11.15%</td>
</tr>
<tr>
<td>Testing Set 5</td>
<td>0.6624</td>
<td>0.2174</td>
<td>100%</td>
</tr>
<tr>
<td>Testing Set 6</td>
<td>0.4140</td>
<td>0.3650</td>
<td>12.5%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JOINT # 4</th>
<th>Original Model</th>
<th>Robust Model</th>
<th>Percentage Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Set</td>
<td>0.0349</td>
<td>0.0327</td>
<td>6.5%</td>
</tr>
<tr>
<td>Testing Set 7</td>
<td>0.0295</td>
<td>0.0213</td>
<td>33%</td>
</tr>
<tr>
<td>Testing Set 8</td>
<td>0.0424</td>
<td>0.0420</td>
<td>1%</td>
</tr>
</tbody>
</table>

Table 5.6: Results of the comparison between the original model and the robust model
In Table 5.6, 2 testing data sets were chosen from the random trajectory of each joint. For joint 1, both the original model and the robust model are well tuned. For the testing set 1, the percentage improvement in the performance using the robust model is 25%. For the testing set 2, the robust model again shows a lower error than the original model with a percentage improvement of 21%. That is due to the fact that some of the points in the testing sets are outer limit data, i.e., points that are not expressed by the model training set. Thus, the original model fails to give a correct estimate of the output for these points. On the other hand, the robust model gives a better estimate of these points and shows a lower error. For joint 2, testing sets 3 and 4 are presented. For the testing set 3, the error of both models is almost the same. For this testing set, all the points have inner limit characteristics and thus, both models predict the data equally well. For the testing set 4, the robust model outperformed the original model showing a much lower error. The percentage improvement in the performance is 112%. For the testing set 4, the original model fails to give the correct estimate of two points in the whole data set and that is why it shows a relatively high error. For joint 3, testing sets 5 and 6 are presented. For both testing sets 5, and 6, the robust model shows a lower error than the original model. The percentage improvement in the performance is 100% for the testing set 5 and 12% for the testing set 6. For joint 4, testing sets 7 and 8 are presented. Again, the robust model shows a lower error for both sets. The percentage improvement in the performance is 33% for testing set 7 and 1% for testing set 8. Using the robust model, the percentage improvement in the accuracy differs from one testing set to another. That is because the testing sets used in this example have different natures. In other words, for some testing sets, all the points have inner limit characteristics (i.e., testing set 3). For other testing sets, the percentage of points with inner limit characteristics is not fixed. Furthermore, using the robust model, a significant improvement in the accuracy is observed for joint 2 and joint 3. For these joints, the original model is not able to give an accurate estimate of the corresponding testing sets, where the effect of friction and dynamics interaction is included. On the other hand, the robust model gives much better estimate for these testing sets and hence shows a lower error. Thus, we conclude that the robust model has a significant generalization capability compared to the original model. Further, by using the
robust model, the marginal continuity property is guaranteed and no unwanted "jumps" in the model output are expected.

Figure 5.11: The fuzziness and cluster validity indices for IRIS joints
Figure 5.11 (contd.): The fuzziness and cluster validity indices for IRS joins.
Figure 5.12: The criterion for the selection of the lower bounds of \( p \) and \( q \) for each joint.
Figure 5.12 (contd.): The criterion for the selection of the lower bounds of $p$ and $q$ for each joint.
Fig. 5.13: Selection of the range for the parameter $\alpha$
Fig 5.14: The tuning process of the training set of each joint
Figure 5.15: Identification of the tuned reasoning parameters from the tuning process
Figure 5.15(contd.): Identification of the tuned reasoning parameters from the tuning process
Fig. 5.16: The fuzzy input-output rules of the robust model for joint 1

Fig. 5.17: The fuzzy input-output rules of the robust model for joint 2
Fig. 5.18: The fuzzy input-output rules of the robust model for joint 3

Figure 5.19: The fuzzy input-output rules of the robust model for joint 4
Figure 5.20: A comparison between the original model and the robust model for the training set of joint 1.
Figure 5.21: A comparison between the original model and the robust model for the testing set 1

Figure 5.22: A comparison between the original model and the robust model for the testing set 2
Figure 5.23: A comparison between the original model and the robust model for the training set of joint 2
Figure 5.24: A comparison between the original model and the robust model for the testing set 3

Figure 5.25: A comparison between the original model and the robust model for the testing set 4
Figure 5.26: A comparison between the original model and the robust model for the training set of joint 3.
**Figure 5.27:** A comparison between the original model and the robust model for the testing set5

**Figure 5.28:** A comparison between the original model and the robust model for the testing set6
Figure 5.29: A comparison between the original model and the robust model for the training set of joint 4
Figure 5.30: A comparison between the original model and the robust model for the testing set7

Figure 5.31: A comparison between the original model and the robust model for the testing set8
5.4 Conclusions

In this Chapter, two main applications were tested. The first application was the famous gas furnace plant introduced by Box and Jenkins (Box, 1970). The second application was the IRIS 4 d.o.f robot manipulator introduced in (Emami, 1996a). For both applications, the techniques developed in this work were used to build a robust fuzzy model that has a continuous behaviour as well as a satisfactory generalization capability. Two testing sets were introduced for the gas furnace application. A comparison between the developed robust model and the original model (Emami, 1996a) was performed. Based on the results, we can see that the robust model shows a lower error for both the testing sets. The comparison between the original model and the robust model was based on minimizing the performance index in equation 1.14. Using the robust model, the average percentage improvement is 7.5%. Since both testing sets used for this application have inner limit characteristics, both the original and the robust model showed a robust performance. In order to illustrate the significance of the techniques developed in this work, we used the IRIS arm, which is a more complicated and realistic dynamic system. For this application, the same comparison between the robust model and the original model was performed. Again, the comparison between the original model and the robust model was based on minimizing the performance index in equation 1.14. The robust model outperformed the original model showing a smaller error for almost all the testing sets introduced. The average percentage improvement in the performance is 41%. As a final conclusion, through the applications illustrated in this thesis, we conclude that the proposed techniques have a significant effect on improving the robustness and the generalization capability of the fuzzy logic models.
6.1 Conclusions

In this work, problems related to the development of a fuzzy model were investigated. First, there is no guarantee that the developed model will show a satisfactory continuous behaviour. In other words, we may experience unwanted jumps in the model output for small deviation in the input. Second, the model may fail in identifying the appropriate partitions in the data if the training input-output data set is contaminated by noise. Finally, there is no guarantee that the developed model will be satisfactory if extended to the new testing data that has not been used in the model construction. These problems are significant in the black-box modeling approach assumed herein due to the fact that there is no physical insight available or used.

In order to perform this investigation, we introduced a definition of the model robustness. A fuzzy model is robust when it has: (i) an appropriate continuous behaviour. (ii) a noise rejection capability, and (iii) an appropriate generalization capability when extended to the new testing data sets.

The systematic methodology of fuzzy-logic modeling (Emami, 1996a) was reviewed. This methodology provides a firm analytical ground for the robustness study in fuzzy models. The goal was to improve this modeling methodology with regard to the above-mentioned problems. First, the continuity behaviour of the inference mechanism was investigated. Through this investigation, the definition of the modulus of continuity in the approximation theory was extended to the fuzzy output of the model, and a relation between the maximum deviation of the input membership function and the reasoning parameters \( p \) and \( q \) was developed. Through this relation, a lower bound for the reasoning parameters was specified to guarantee the marginal continuity property. The concept of the marginal continuity was also extended to the model crisp output. Four
theorems were developed to guarantee the marginal continuity property and to improve the robustness of the fuzzy inference mechanism.

Second, an improved noise rejection clustering algorithm was introduced, based on combining the PCM algorithm (Krishnapuram, 1994) with the techniques developed in the improved fuzzy C-Means clustering algorithm (Emami, 1996b). The new algorithm is based on a defined criterion for the assignment of the cutoff distance from the data. The partitions obtained using the combined algorithm were more efficient than those obtained by the NPCM (Krishnapuram, 1996). After rejecting the noise, a new membership matrix was constructed, where the noise points have low or zero weights among all the clusters.

Third, the generalization capability of the fuzzy logic model was investigated. Some conditions on the sampled data were introduced. Data observation methods were also studied and some conditions on the input-output membership functions were presented. A new tuning process was introduced that is based on evaluating several testing data sets during the tuning process of the model training set. Finally, two different evaluation criteria were introduced to validate the developed fuzzy-logic models.

The above results were illustrated through a simple and well known example of two-inputs single-output nonlinear function (Sugeno, 1993). For this example, a comparison study between the original model (Emami, 1996a) and the developed robust model was performed. For the testing sets with outer limit characteristics (contains data points out of the input membership range), the robust model showed a much lower error with an improvement of 95% in the performance over the original model. For the testing sets without outer limit characteristics, the robust model showed a percentage improvement of 45% in the performance compared to the original model. Therefore, a significant improvement in the generalization capability of the fuzzy model was achieved.

Finally, the results were applied to two real applications. The first application was the famous gas furnace plant introduced by Box and Jenkins (Box, 1970). The second application was the IRIS 4 d.o.f robot manipulator introduced in (Emami, 1996a). For the gas furnace plant application, a comparison between the developed robust model and the original model (Emami, 1996a) was performed. The robust model showed a better performance for both testing sets defined for this application. Using the robust model, the percentage improvement in the model performance was 7.5%. For the IRIS robot arm, the
same comparison between the robust model and the original model was performed. For this application, the robust model significantly outperformed the original model showing a smaller error for almost all the testing sets introduced. The average percentage improvement in the model performance was 41%.

From all the three applications presented in this work (Sugeno example, gas furnace plant, and robot manipulator), we conclude that the proposed techniques have a significant effect on improving the robustness and the generalization capability of the fuzzy logic models.

6.2 Future Research

At the completion of this work, it is concluded that there are some issues that need further investigation. Some of the future research directions are as follows:

- In order to improve the generalization capability of the fuzzy logic models, some conditions are applied to the input membership functions. Further investigation about the construction of the input–output membership functions is needed to provide a better model performance subject to any testing data set.

- In the noise rejection clustering algorithm introduced in Chapter 3, we introduced a criterion to assign the cut-off distance from the data. Further research is needed to eliminate any heuristics related to choosing this criterion.

- In this work, physical systems with "crisp" inputs are considered. We suggest to extend the techniques developed in this research to the case of "fuzzy" inputs. This extension will be useful for economics and environmental applications.

- Different model evaluation criteria were introduced in this work. It would be useful to perform a comparison study between these criteria.

- The ideas presented in this research can also be extended to the fuzzy-logic control.
REFERENCES

- Adnan M. Awad; “Properties of the akaike information criterion”: Microelectronics and Reliability; Vol. 36; No. 4; pp. 457-464; 1996
- Domingo C., Tsukiji T., Watanabe O.; “Partial occam’s razor and its applications”: Information Processing Letters 64; pp.179-185; 1997


• Lorenz G. G.; “Approximation of Functions”: *Holt*: NY; 1966

• Lunneborg C.E.; “Modeling Experimental and Observational Data”: *Duxbury Press*: pp.27-37; 1994

• MacKay D. J.; “Bayesian interpolations”: *Neural Computations* 4; No. 3; pp. 415-447; 1992


• Schweizer B., Sklar A.; “Associative functions and statistical triangular inequalities”: *Publicationes Mathematicae Debrecen*: No. 8; pp. 169-186; 1961

• Schweizer B., Sklar A.; “Probabilistic Metric Spaces”: *North-Holland: Amsterdam*; 1983


• Sugeno M., Yasukawa T.; “A fuzzy-logic-based approach to qualitative modeling”: *IEEE Transactions on Fuzzy Systems*; Vol.1; No.1; 1993

• Van den Bosch P. P. J., Van der Klauw A. C.; “Modeling, Identification and Simulations of Dynamical Systems”; *CRC Press*; pp.51-67; 1994

• Ward J. H.; “Hierarchical grouping to optimize an objective function”: *Journal of American Statistics Association*; No. 58; pp.236-244; 1963


• Zadeh L.; “Outline of a new approach to the analysis of complex systems and decision processes”: *IEEE Transactions on Systems, Man, and Cybernetics, SMC-3*; pp. 28-44; 1973

• Zadeh L.; “Fuzzy sets”: *Information and Control*; Vol. 8; pp. 338-353; 1965
### APPENDIX 1

**CHI-SQUARE DISTRIBUTION TABLES**

<table>
<thead>
<tr>
<th>df</th>
<th>0.995</th>
<th>0.990</th>
<th>0.975</th>
<th>0.950</th>
<th>0.900</th>
<th>0.750</th>
<th>0.500</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>392704.10^{-6}</td>
<td>1.57088.10^{-6}</td>
<td>9.82069.10^{-6}</td>
<td>9.93214.10^{-6}</td>
<td>0.0157908</td>
<td>0.1015308</td>
<td>0.454937</td>
</tr>
<tr>
<td>2</td>
<td>0.0100251</td>
<td>0.0201007</td>
<td>0.0506356</td>
<td>0.102587</td>
<td>0.210720</td>
<td>0.575364</td>
<td>1.38629</td>
</tr>
<tr>
<td>3</td>
<td>0.0717212</td>
<td>0.114832</td>
<td>0.215795</td>
<td>0.351846</td>
<td>0.584375</td>
<td>1.212534</td>
<td>2.36597</td>
</tr>
<tr>
<td>4</td>
<td>0.206990</td>
<td>0.297110</td>
<td>0.484419</td>
<td>0.710721</td>
<td>1.063623</td>
<td>1.92255</td>
<td>3.35670</td>
</tr>
<tr>
<td>5</td>
<td>0.411740</td>
<td>0.554300</td>
<td>0.831211</td>
<td>1.154576</td>
<td>1.61031</td>
<td>2.67460</td>
<td>4.35146</td>
</tr>
<tr>
<td>6</td>
<td>0.675727</td>
<td>0.872085</td>
<td>1.237347</td>
<td>1.63539</td>
<td>2.20413</td>
<td>3.45640</td>
<td>5.34812</td>
</tr>
<tr>
<td>7</td>
<td>0.989265</td>
<td>1.239043</td>
<td>1.68987</td>
<td>2.16735</td>
<td>2.83311</td>
<td>4.25485</td>
<td>6.34581</td>
</tr>
<tr>
<td>8</td>
<td>1.344419</td>
<td>1.646482</td>
<td>2.17973</td>
<td>2.73264</td>
<td>3.48954</td>
<td>5.07064</td>
<td>7.34412</td>
</tr>
<tr>
<td>9</td>
<td>1.734926</td>
<td>2.087912</td>
<td>2.70039</td>
<td>3.32511</td>
<td>4.16186</td>
<td>5.89883</td>
<td>8.34283</td>
</tr>
<tr>
<td>11</td>
<td>2.60321</td>
<td>3.05347</td>
<td>3.81575</td>
<td>4.57481</td>
<td>5.57779</td>
<td>7.58412</td>
<td>10.3410</td>
</tr>
<tr>
<td>12</td>
<td>3.07382</td>
<td>3.57056</td>
<td>4.40379</td>
<td>5.22603</td>
<td>6.30380</td>
<td>8.43842</td>
<td>11.3403</td>
</tr>
<tr>
<td>17</td>
<td>5.69724</td>
<td>6.40776</td>
<td>7.56418</td>
<td>8.67176</td>
<td>10.0852</td>
<td>12.7919</td>
<td>16.3381</td>
</tr>
<tr>
<td>50</td>
<td>27.9907</td>
<td>29.7067</td>
<td>32.3574</td>
<td>34.7642</td>
<td>37.6886</td>
<td>42.9421</td>
<td>49.3349</td>
</tr>
<tr>
<td>60</td>
<td>35.5346</td>
<td>37.4848</td>
<td>40.4817</td>
<td>43.1879</td>
<td>46.4589</td>
<td>52.2938</td>
<td>59.3347</td>
</tr>
<tr>
<td>70</td>
<td>43.2752</td>
<td>45.4418</td>
<td>48.7576</td>
<td>51.7393</td>
<td>55.3290</td>
<td>61.6983</td>
<td>69.3344</td>
</tr>
<tr>
<td>80</td>
<td>51.1720</td>
<td>53.5040</td>
<td>57.1532</td>
<td>60.3915</td>
<td>64.2778</td>
<td>71.1445</td>
<td>79.3343</td>
</tr>
<tr>
<td>90</td>
<td>59.1963</td>
<td>61.7541</td>
<td>65.6466</td>
<td>69.1260</td>
<td>73.2912</td>
<td>80.6247</td>
<td>89.3342</td>
</tr>
<tr>
<td>100</td>
<td>67.3276</td>
<td>70.0648</td>
<td>74.2219</td>
<td>77.9295</td>
<td>82.3581</td>
<td>90.1332</td>
<td>99.3341</td>
</tr>
</tbody>
</table>
## Chi-Square Distribution Tables

<table>
<thead>
<tr>
<th>df</th>
<th>0.250</th>
<th>0.100</th>
<th>0.050</th>
<th>0.025</th>
<th>0.010</th>
<th>0.005</th>
<th>0.001</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.32330</td>
<td>2.70554</td>
<td>3.84146</td>
<td>5.02389</td>
<td>6.63490</td>
<td>7.87944</td>
<td>10.828</td>
</tr>
<tr>
<td>2</td>
<td>2.77259</td>
<td>4.60517</td>
<td>5.99147</td>
<td>7.37762</td>
<td>9.21034</td>
<td>10.587</td>
<td>13.816</td>
</tr>
<tr>
<td>13</td>
<td>15.9839</td>
<td>19.8119</td>
<td>22.3621</td>
<td>24.7356</td>
<td>27.6883</td>
<td>29.8194</td>
<td>34.528</td>
</tr>
<tr>
<td>15</td>
<td>18.2451</td>
<td>22.3072</td>
<td>24.9958</td>
<td>27.4884</td>
<td>30.5779</td>
<td>32.8013</td>
<td>37.697</td>
</tr>
<tr>
<td>17</td>
<td>20.4887</td>
<td>24.7690</td>
<td>27.5871</td>
<td>30.1910</td>
<td>33.4087</td>
<td>35.7185</td>
<td>40.790</td>
</tr>
<tr>
<td>18</td>
<td>21.6049</td>
<td>25.9894</td>
<td>28.8693</td>
<td>31.5264</td>
<td>34.8053</td>
<td>37.1564</td>
<td>42.312</td>
</tr>
<tr>
<td>19</td>
<td>22.7178</td>
<td>27.2036</td>
<td>30.1435</td>
<td>32.8523</td>
<td>36.1908</td>
<td>38.5822</td>
<td>43.820</td>
</tr>
<tr>
<td>20</td>
<td>23.8277</td>
<td>28.4120</td>
<td>31.4104</td>
<td>34.1696</td>
<td>37.5662</td>
<td>39.9968</td>
<td>45.315</td>
</tr>
<tr>
<td>21</td>
<td>24.9348</td>
<td>29.6151</td>
<td>32.6705</td>
<td>35.4789</td>
<td>38.9321</td>
<td>41.4010</td>
<td>46.797</td>
</tr>
<tr>
<td>22</td>
<td>26.0393</td>
<td>30.8133</td>
<td>33.9244</td>
<td>36.7807</td>
<td>40.2894</td>
<td>42.7956</td>
<td>48.268</td>
</tr>
<tr>
<td>23</td>
<td>27.1413</td>
<td>32.0069</td>
<td>35.1725</td>
<td>38.0757</td>
<td>41.6384</td>
<td>44.1813</td>
<td>49.728</td>
</tr>
<tr>
<td>24</td>
<td>28.2412</td>
<td>33.1963</td>
<td>36.4151</td>
<td>39.3641</td>
<td>42.9798</td>
<td>45.5585</td>
<td>51.179</td>
</tr>
<tr>
<td>25</td>
<td>29.3389</td>
<td>34.3816</td>
<td>37.6525</td>
<td>40.6665</td>
<td>44.3141</td>
<td>46.9278</td>
<td>52.620</td>
</tr>
<tr>
<td>26</td>
<td>30.4345</td>
<td>35.5631</td>
<td>38.8832</td>
<td>42.0232</td>
<td>45.6417</td>
<td>48.2899</td>
<td>54.052</td>
</tr>
<tr>
<td>27</td>
<td>31.5284</td>
<td>36.7412</td>
<td>40.1133</td>
<td>43.3944</td>
<td>46.9630</td>
<td>49.6449</td>
<td>55.476</td>
</tr>
<tr>
<td>28</td>
<td>32.6205</td>
<td>37.9159</td>
<td>41.3372</td>
<td>44.6607</td>
<td>48.2782</td>
<td>50.9933</td>
<td>56.892</td>
</tr>
<tr>
<td>29</td>
<td>33.7109</td>
<td>39.0875</td>
<td>42.5569</td>
<td>45.9222</td>
<td>49.5879</td>
<td>52.3356</td>
<td>58.302</td>
</tr>
<tr>
<td>30</td>
<td>34.7998</td>
<td>40.2560</td>
<td>43.7729</td>
<td>46.9792</td>
<td>50.8922</td>
<td>53.6720</td>
<td>59.703</td>
</tr>
<tr>
<td>40</td>
<td>45.6160</td>
<td>51.8050</td>
<td>55.7585</td>
<td>59.3417</td>
<td>63.6907</td>
<td>66.7659</td>
<td>73.402</td>
</tr>
<tr>
<td>50</td>
<td>56.3336</td>
<td>63.1671</td>
<td>67.5048</td>
<td>71.4202</td>
<td>76.1539</td>
<td>79.4900</td>
<td>86.661</td>
</tr>
<tr>
<td>60</td>
<td>66.9814</td>
<td>74.3970</td>
<td>79.0819</td>
<td>83.2976</td>
<td>88.3794</td>
<td>91.9517</td>
<td>99.607</td>
</tr>
<tr>
<td>70</td>
<td>77.5766</td>
<td>85.5271</td>
<td>90.5312</td>
<td>95.0231</td>
<td>100.425</td>
<td>104.215</td>
<td>112.317</td>
</tr>
<tr>
<td>80</td>
<td>88.1303</td>
<td>96.5782</td>
<td>101.879</td>
<td>106.629</td>
<td>112.329</td>
<td>116.321</td>
<td>124.839</td>
</tr>
<tr>
<td>90</td>
<td>98.6499</td>
<td>107.565</td>
<td>113.145</td>
<td>118.136</td>
<td>124.116</td>
<td>128.299</td>
<td>137.208</td>
</tr>
<tr>
<td>100</td>
<td>109.141</td>
<td>118.498</td>
<td>124.342</td>
<td>129.561</td>
<td>135.807</td>
<td>140.169</td>
<td>149.449</td>
</tr>
</tbody>
</table>